Embedding Motion in Model-Based Stochastic Tracking

Jean-Marc Odobez, *Member, IEEE,* Daniel Gatica-Perez, *Member, IEEE,* and Sileye Ba

***Abstract*— Particle filtering is now established as one of the most popular methods for visual tracking. Within this framework, there are two important considerations. The first one refers to the generic assumption that the observations are temporally independent given the sequence of object states. The second consideration, often made in the literature, uses the transition prior as proposal distribution. Thus, the current observations are not taken into account, requesting the noise process of this prior to be large enough to handle abrupt trajectory changes. As a result, many particles are either wasted in low likelihood regions of the state space, resulting in low sampling efficiency, or more importantly, propagated to distractor regions of the image, resulting in tracking failures. In this paper, we propose to handle both considerations using motion. We first argue that in general observations are conditionally correlated, and propose a new model to account for this correlation allowing for the natural introduction of implicit and/or explicit motion measurements in the likelihood term. Secondly, explicit motion measurements are used to drive the sampling process towards the most likely regions of the state space. Overall, the proposed model allows to handle abrupt motion changes and to filter out visual distractors when tracking objects with generic models based on shape or color distribution. Experimental results obtained on head tracking, using several sequences with moving camera involving large dynamics, and compared against the CONDENSATION algorithm, have demonstrated superior tracking performance of our approach.**[1]

***Index Terms*— Visual tracking, particle filter, sequential monte carlo, importance sampling, motion.**

I. Introduction

VISUAL tracking is an important problem in computer vision, with applications in teleconferencing, visual surveillance, gesture recognition, and vision based interfaces [2]. Although tracking has been intensively studied in the literature, it still represents a challenging task in adverse situations, due to the presence of ambiguities (e.g. when tracking an object in a cluttered scene or when tracking multiple instances of the same object class), the noise in image measurements (e.g. lighting chnages), and the variability of the object class (e.g. pose variations).

In the pursuit of robust tracking, Sequential Monte Carlo methods [2]–[4] have shown to be a successful approach. In this temporal Bayesian framework, the probability of an object configuration given the observations is represented by a set of weighted random samples, called particles. This representation allows in principle to simultaneously maintain multiple hypotheses in the presence of ambiguities, unlike algorithms that keep only one configuration state [5], which are therefore sensitive to single failures in the presence of ambiguities or fast or erratic motion.

In this paper, we address two important issues related to tracking with a particle filter. The first issue refers to the specific form of the observation likelihood, that relies on the conditional independence of observations given the state sequence. The second one refers to the choice of an appropriate proposal distribution, which, unlike the prior dynamical model, should take into account the new observations. To handle these issues, we propose a new particle filter tracking method based on visual motion. Our method relies on a new graphical model allowing for the natural introduction of implicit or explicit motion information in the likelihood term, and on the exploitation of explicit motion measurements in the proposal distribution. the above issues, our approach, and their benefits, is given in the following paragraphs.

The definition of the observation likelihood distribution is perhaps the most important element in visual tracking with a particle filter. This distribution allows for the evaluation of the likelihood of the current observation given the current object state, and relies on the specific object representation. The object representation corresponds to all the information that characterizes the object like the target position, geometry, appearance, color, etc. Parametrized shapes like splines [2] or ellipses [6], and color distributions [5]–[8], are often used as target representation. One drawback of these generic representations is that they can be quite unspecific, which augments the chances of ambiguities. One way to improve the robustness of a tracker consists of combining low-level measurements such as shape and color [6].

The generic conditional form of the likelihood term relies on a standard hypothesis in probabilistic visual tracking, namely the independence of observations given the state sequence [2], [6], [9]–[13]. In this paper, we argue that this assumption can be inaccurate in the case of visual tracking. As a remedy, we propose a new model that assumes that the current observation depends on the current and previous object configurations as well as on the past observation. We show that under this more general assumption, the obtained particle filtering algorithm has similar equations than the algorithm based on the standard hypothesis. To our knowledge, this has not been shown before, and so it represents the first contribution of this article. The new assumption can thus be used to naturally introduce implicit or explicit motion information in the observation likelihood term. The introduction of such data correlation between successive images will turn generic trackers like shape or color histogram trackers into more specific ones.

The authors are with the IDIAP Research Institute, Rue du Simplon 4, 1920 Martigny, Switzerland

[1] A shorter version of this paper appeared at the ICPR 2004 conference [1].

Another important distribution to define when designing a particle filter is the proposal distribution, that is, the function that predicts the new state hypotheses where the observation likelihood will be evaluated. In general, an optimal choice [3], [14] consists of drawing samples from the more likely regions taking into account both the dynamical model, which characterizes the prior on the state sequence, and the new observations. However, simulating from the optimal law is often difficult when using standard likelihood terms. Thus, a common assumption in particle filtering consists in using the dynamics as proposal distribution. With this assumption, the variance of the noise process in the dynamical model implicitly defines a search range for the new hypotheses. This assumption raises difficulties in modeling dynamics since this term should fulfill two contradictory objectives. On one hand, as prior distribution, dynamics should be tight to avoid the tracker being confused by distractors in the vicinity of the true object configuration, a situation that is likely to happen for unspecific object representations such as generic shapes or color distributions. On the other hand, as proposal distribution, dynamics should be broad enough to cope with abrupt motion changes. Furthermore, this proposal distribution does not take into account the most recent observations. probably have a low likelihood, which results in low sampling efficiency. Overall, such a particle filter is likely to be distracted by background clutter. To address these issues, we propose to use explicit motion measures in the proposal function. One benefit of this approach will be to increase the sampling efficiency by handling unexpected motion, allowing for a reduced noise variance in the prediction process. Combined with the new observation likelihood term, using our proposal distribution will reduce the sensitivity of the particle filter algorithm to the different noise variances setting in the proposal and prior since, when using larger values, potential distractors should be filtered out by the introduced correlation and visual motion measurements. Finally, our proposal allows to implement the intuitive idea according to which the likely configurations with respect to an object model are evolving in conformity with the visual motion.

The rest of the paper is organized as follows. In the next Section, we discuss the state-of-the-art and relate it to our work. For sake of completeness, in Section III, we describe the standard particle filter algorithm. Our approach is motivated in Section IV, while Section V describes the specific parts of our model in details. Experiments and results are reported in Section VI. Section VII concludes the article with some discussion and future work.

II. Related work

In this article, the first contribution refers to the introduction of a new graphical model for particle filtering. This model allows for the modeling of temporal dependencies between observations. In practice, it lead us to naturally introduce motion observation within the data likelihood.

The use of motion for tracking is not a new idea. Motion-based trackers, essentially deterministic, integrate two-frame motion estimates over time. However, without any object model, it is almost impossible to avoid some drift after a few seconds of tracking. For long term tracking, the use of appearence-based models such as templates [9], [15], [16] lead to more robust results. However, a template representation do not allow for large changes of appearence over time.

To handle appearance changes, an often difficult template adaptation step is needed [17], [18], or more complex global appearence models are used (e.g. eigen-spaces [19] or examplars [13], [20]), which poses the problem of learning these models, either off-line [10], [13] or on-line [20]. For instance, in [17], a generative model relying on the past frame template, a long term template, and a non-Gaussian noise component is proposed. Adaptation is performed through the estimation of the optimal state parameters -comprising the spatial 2D localization and the long-term template-, via an EM algorithm that identifies the stable regions of the template as a byproduct. A similar approach is taken in [18], where the gray level of each template pixel is updated using a Kalman filter, and the adaptation is blocked whenever the innovation is too large. In these two cases, although partial and total occlusion can be handled, nothing prevents the tracker from long term drifts. This drift happens when the 2D visual motion does not match perfectly the real state evolution. This corresponds to the problematic case, reported in [17], of a turning head remaining at the same place in the image; in [18], tracked objects (mainly high resolution faces and people) undergo very little pose changes. Another interesting approach towards adaptation using motion is proposed in [21] where, in a particle filter framework, a color model is adapted on-line. Assuming a static camera, a motion detection module is employed to select the instants more suitable for adaptation, which leads to good results.

In the present article, however, the method we propose is not template-based, i.e. no *reference* appearence template is employed or adapted (see discussion at the end of subsection V-C.2). The implementation of our model aims at evaluating, either explicitly or implicitly, the similarity between the visual motion estimated from low-level information and the motion field induced by the state change. Our approach is thus different from the above ones, and more similar to the methods proposed in [22], [23]. In particular, the work in [22] addresses the difficult problem of people tracking using articulated models, and their use of the motion measures implicitly corresponds to the graphical model we propose here.

In the introduction, we raised the problems linked to the choice of the dynamical model as proposal. In the literature, several approaches have been proposed to address these issues. For instance, when available, auxiliary information generated from color [11], [21], [24], motion detection [24], or audio in the case of speaker tracking [24], [25], can be used to draw samples from. The proposal distribution is then expressed as a mixture of the prior and components of the likelihood distribution. An important advantage of this approach is to allow for automatic (re)initialization. However, one drawback of this approach is that, since these additional samples are not related to the previous samples, the evaluation of the transition prior term for one new sample involves all past samples, which can become very costly [11], [25]. [24] avoids this problem by defining the prior as a mixture of distributions that includes a uniform law component, and by relying on distinctive and discriminative likelihoods, allowing for reinitialization using the standard particle filter equations. Another auxiliary particle filter proposed in [26] avoids this problem. The idea is to use the likelihood of a first set of predicted samples at time $k+1$ to resample the seed samples at time k, and to then apply the standard prediction and evaluation steps on these seed samples. The feedback from the new data acts by increasing or decreasing the number of descendents of a sample according to its "predicted" likelihood. Such a scheme, however, works well only if the variance of the transition prior is small, which is usually not the case in visual tracking.

As an alternative, the work in [12] proposed to use the unscented particle filter to generate importance densities. Although attractive, it is still likely to fail in the presence of abrupt motion changes, and the method needs to convert likelihood evaluations (e.g. of shape) into state space measurements (e.g. translation, scale). This would be difficult with color distribution likelihoods and for some state parameters. In [12], only a translation state is considered. In [9], [27], all the equations of the filter are conditioned with respect to the images. This allows for the use of the inter-frame motion estimates as dynamical model instead of an auto-regressive model to improve the state prediction. Moreover, in their application (point tracking), thanks to the use of a linear observation model, the optimal proposal function can be employed. However, as in [12], measures in state space are needed, and only translations are thus considered. Although their utilization of explicit motion measures is similar to what we propose here, it was introduced in a different way (through the dynamics rather than the likelihood), and was in practice restricted to translation.

III. Particle filtering

There exist at least two ways of introducing particle filters. The first one is through Sequential Importance Sampling (SIS) [3], [4], and the second one is based on factored sampling [28] applied to the filtering distribution [2]. While both approaches lead to the same algorithm with the standard assumptions, it is interesting to notice that the two methods do not lend themselves to the same extensions. In this paper, we follow the SIS approach, as it allows for the proposed extension.

Particle filtering is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations. The key idea is to represent the required *posterior* probability density function (pdf) $\mathrm{p}(c_{0:k}|z_{1:k})$ of the state sequence $c_{0:k} = \{c_l, l = 0, \ldots, k\}$ up to time k conditionally to the observation sequence $z_{1:k} = \{z_l, l = 1, \ldots, k\}$, by a set of weighted samples $\{c^i_{0:k}, w^i_k\}_{i=1}^{N_s}$. Each sample (or particle) $c^i_{0:k}$ represents a potential trajectory of the state sequence, and w^i_k denotes its likelihood estimated from the sequence of observations up to time k. The weights are normalized ($\sum_i w^i_k = 1$) in order to obtain a discrete approximation of the true posterior :

$$\mathrm{p}(c_{0:k}|z_{1:k}) \approx \sum_{i=1}^{N_s} w^i_k \delta(c_{0:k} - c^i_{0:k}) \,. \tag{1}$$

Such a representation then allows to compute the expectation of any function f with respect to this distribution using a weighted sum :

$$\int f(c_{0:k}) \mathrm{p}(c_{0:k}|z_{1:k}) \mathrm{d}c_{0:k} \approx \sum_{i=1}^{N_s} w^i_k f(c^i_{0:k}) \tag{2}$$

and in particular, the mean of the hidden state sequence can be computed from the first order moment (i.e. by using $f(x) = x$). More specifically, the samples and the weights have to be chosen such that, for any function f, the right-hand side of (2) converges to the left-hand side when N_s tends towards infinity, even though the Dirac delta probability density function in the right-hand side of Eq. 1 may not converge pointwise to the true density $\mathrm{p}(c_{0:k}|z_{1:k})$. Since sampling directly form the posterior is usually impossible, the weights are chosen using the principle of Importance Sampling (IS). This consists in simulating the samples from an importance (a.k.a proposal) function, and then introducing a correction factor (the weight) to account for the discrepancy between the proposal and the true posterior. More precisely, denoting by $q(c_{0:k}|z_{1:k})$ the importance density, the proper weights in (1) are given by :

$$w^i_k \propto \frac{\mathrm{p}(c^i_{0:k}|z_{1:k})}{q(c^i_{0:k}|z_{1:k})} \,. \tag{3}$$

The goal of the particle filtering algorithm is the recursive propagation of the samples and estimation of the associated weights as each measurement is received sequentially. Applying Bayes' rule, we obtain the following recursive equation for the posterior :

$$\begin{aligned}\mathrm{p}(c_{0:k}|z_{1:k}) =& \frac{\mathrm{p}(z_k|c_{0:k}, z_{1:k-1})\mathrm{p}(c_k|c_{0:k-1}, z_{1:k-1})}{\mathrm{p}(z_k|z_{1:k-1})} \\ & \times \mathrm{p}(c_{0:k-1}|z_{1:k-1})\end{aligned} \tag{4}$$

Assuming a factorized form for the proposal (i.e. $q(c_{0:k}|z_{1:k}) = q(c_k|c_{0:k-1}, z_{1:k})q(c_{0:k-1}|z_{1:k-1})$) we obtain the following recursive update equation [3], [4]:

$$\begin{aligned} w^i_k &= \frac{\tilde{w}^i_k}{\mathrm{p}(z_k|z_{1:k-1})} \text{ with} \\ \tilde{w}^i_k &= w^i_{k-1} \, \frac{\mathrm{p}(z_k|c^i_{0:k}, z_{1:k-1})\mathrm{p}(c^i_k|c^i_{0:k-1}, z_{1:k-1})}{q(c^i_k|c^i_{0:k-1}, z_{1:k})} \,. \end{aligned} \tag{5}$$

where $\tilde{w}^i_k$ is the unnormalized weight of the particle i. The factor $p(z_k|z_{1:k-1})$ is constant with respect to the state values, and it is easy to show that this factor can be approximated by $\sum_{i=1}^{N_s} \tilde{w}^i_k$, so that the weights w^i_k are indeed correctly normalized. In order to simplify the general expression of Eq. 5, conditional dependencies between variables are usually modeled according to the graphical model of Figure 1a, which corresponds to the following assumptions :

H1 : The observations $\{z_k\}$, given the sequence of states, are independent. This leads to $\mathrm{p}(z_{1:k}|c_{0:k}) = \prod_{i=1}^{k} \mathrm{p}(z_k|c_k)$, which requires the definition of the data-likelihood $\mathrm{p}(z_k|c_k)$. In Eq. 5, this assumption translates in $\mathrm{p}(z_k|c_{0:k}, z_{1:k-1}) = \mathrm{p}(z_k|c_k)$.

H2 : The state sequence $c_{0:k}$ follows a first-order Markov chain model. In Eq. 5, this means that $\mathrm{p}(c_k|c_{0:k-1}, z_{1:k-1}) = \mathrm{p}(c_k|c_{k-1})$.

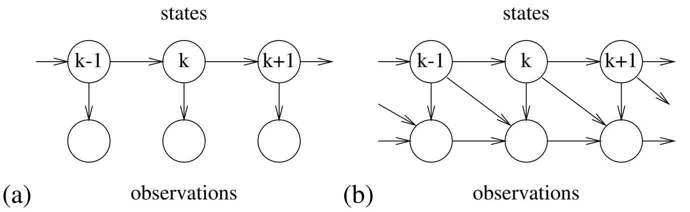


Fig. 1. Graphical models for tracking. (a) standard and (b) proposed model.

We then obtain the simplified weight update equation :

$$w_k^i \propto w_{k-1}^i \, \frac{\mathrm{p}(z_k|c_k^i)\mathrm{p}(c_k^i|c_{k-1}^i)}{q(c_k^i|c_{0:k-1}^i, z_{1:k})} \quad (\text{ and } \sum_i w_k^i = 1 \text{) }. \quad (6)$$

The set $\{c_{0:k}^i, w_k^i\}_{i=1}^{N_s}$ is then approximately distributed according to $\mathrm{p}(c_{0:k}|z_{0:k})$.

It is known that importance sampling is inefficient in high-dimensional spaces [14], which is the case of the state space $c_{0:k}$ as k increases. In practice, this leads to the continuous increase of the weight variance, concentrating the mass of the weights onto a few particles only. To solve this problem, it is necessary to apply an additional resampling step, whose effect is to eliminate the particles with low importance weights and to multiply particles having high weights. Several resampling schemes exist [14]. In our implementation, we used the one described in [3], and perform a systematic resampling. We finally obtain the particle filter displayed in Fig. 2.

```
1) Initialization
   - for $i = 1, \ldots, N_s$, sample $c_0^i \sim \mathrm{p}(c_0)$ and set $k = 1$.
2) Diffusion/propagation :
   - for $i = 1, \ldots, N_s$, sample $\tilde{c}_k^i \sim q(c_k^i|c_{0:k-1}^i, z_{1:k})$.
3) Weight updating
   - for $i = 1, \ldots, N_s$, evaluate the weight $w_k^i$ with Equation (5)
4) Selection resample with replacement $N_s$ particles
   - $\{c_k^j, \frac{1}{N_s}\} \leftarrow$ resample($\{\tilde{c}_k^i, w_k^i\}$)
   - set $k = k + 1$ and goto step 2.
```

Fig. 2. The generic particle filter algorithm.

The efficiency of a particle filter algorithm relies on the definition of a good proposal distribution. A temporally local strategy consists of choosing the importance function that minimizes the weight variance of the new samples at time k conditionally to trajectories $c_{1:k-1}^i$ and observations $z_{1:k}$. It can be shown [14] that this optimal function is given by

$$\begin{aligned} q(c_k|c_{0:k-1}^i, z_{1:k}) &= q(c_k|c_{k-1}^i, z_k) = \mathrm{p}(c_k|c_{k-1}^i, z_k) \\ &\propto \mathrm{p}(z_k|c_k)\mathrm{p}(c_k|c_{k-1}^i) \, , \end{aligned} \quad (7)$$

which leads to the following weight update equation :

$$w_k^i \propto w_{k-1}^i \; \mathrm{p}(z_k|c_{k-1}^i) \, . \quad (8)$$

In practice, sampling from $\mathrm{p}(c_k|c_{k-1}^i, z_k)$ and evaluating $\mathrm{p}(z_k|c_{k-1}^i)$ are only achievable in particular cases, involving for instance Gaussian noise and linear observation models [3], [14], [27]. As an alternative, a choice often made consists of selecting the *prior* as importance function. In that case, we have :

$$w_k^i \propto w_{k-1}^i \; \mathrm{p}(z_k|c_k^i) \, . \quad (9)$$

Although this model is intuitive and simple to implement, this choice, which does not take into account the current observations, has several drawbacks, especially with high-dimensional vector spaces or narrow likelihood models.

Finally, notice that while the weighted set $\{c_{0:k}^i, w_k^i\}_{i=1}^{N_s}$ allows for the representation of the posterior pdf $\mathrm{p}(c_{0:k}|z_{0:k})$, the set $\{c_k^i, w_k^i\}_{i=1}^{N_s}$, that can be obtained from it, is also a representative sample of the filtering distribution $\mathrm{p}(c_k|z_{0:k})$, thanks to simple marginalization.

IV. Our approach

In this Section, we propose a new method that embeds motion in the particle filter. This is first obtained by incorporating motion information into the measurement process. This can be achieved by modifying the traditional graphical model represented in Fig. 1a, by making the current observation dependent not only on the current object configuration but also on the object configuration and observation at the previous instant (see Fig. 1b). Secondly, we propose to use explicit motion measurements in order to obtain a better proposal distribution. In the following Subsections, we motivate our approach by pointing out the limitations of the basic particle filter.

A. Revisiting the hypotheses in particle filtering

The filter described in Fig. 2 is based on the standard probabilistic model for tracking displayed in Fig. 1a and corresponding to hypotheses H1 and H2 of the previous section.

In visual tracking, hypothesis H1 of conditional independence of temporal measurements given the states may not be very accurate. Keeping only two time instants for simplicity, the assumption implies that for all state sequences $c_{k-1:k}$ and data sequences $z_{k-1:k}$,

$$\mathrm{p}(z_k, z_{k-1}|c_k, c_{k-1}) = \mathrm{p}(z_k|c_k, c_{k-1})\mathrm{p}(z_{k-1}|c_k, c_{k-1}) \, .$$

This is a very strong assumption: in practice, some state sequences $c_{k-1:k}$ of interest (e.g. the "true" or "target" state sequences, or state sequences close to the mean state sequence) for which the data are correlated, and hence, for which the standard assumption does not hold. This can be illustrated as follows.

In most tracking algorithms, the state space includes the parameters of a geometric transformation $\mathcal{T}$. Then, the measurements consist of implicitly or explicitly extracting some part of the image by :

$$\tilde{z}_{c_k}(\mathbf{r}) = z_k(\mathcal{T}_{c_k}\mathbf{r}) \qquad \forall \mathbf{r} \in R \, , \quad (10)$$

where $\tilde{z}_{c_k} = z_k|c_k$, $\mathbf{r}$ denotes a pixel position, R denotes a fixed reference region, and $\mathcal{T}_{c_k}\mathbf{r}$ represents the application of the transform $\mathcal{T}$ parameterized by c_k to the pixel $\mathbf{r}$. The data likelihood is then usually computed from this local patch : $\mathrm{p}(z_k|c_k) = \mathrm{p}(\tilde{z}_{c_k})$. However, if c_{k-1} and c_k correspond to

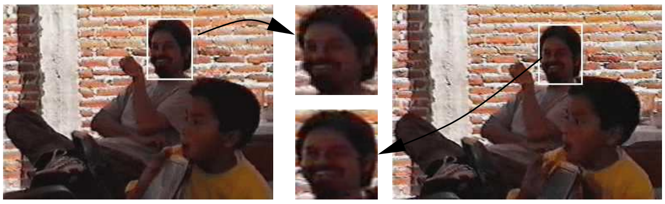


Fig. 3. Images at time t and $t+3$. The two local patches corresponding to the head and extracted from the two images are strongly correlated.

two consecutive states of a given object, it is reasonable to assume that :

$$\tilde{z}_{c_k}(\mathbf{r}) = \tilde{z}_{c_{k-1}}(\mathbf{r}) + \eta(\mathbf{r}) \qquad \forall \mathbf{r} \in R\,. \tag{11}$$

where $\eta(\mathbf{r})$ are prediction noise random variables, assumed to be symmetric with zero mean. This point is illustrated in Figure 3. Equation (11) is at the core of all motion estimation and compensation algorithms like MPEG and is indeed a valid hypothesis [29]. More formally, if we consider the patch $\tilde{z}_\bullet$ as a vector of i.i.d components, we can compute the normalized cross-correlation (NCC) between two data vectors $\tilde{z}_{c_{k-1}}$ and $\tilde{z}_{c_k}$, for state couples $c_{k-1:k}$ of interest, to study their dependencies. The NCC of two patches $\tilde{z}_1$ and $\tilde{z}_2$ is given by :

$$\mathrm{NCC}(\tilde{z}_1, \tilde{z}_2) = \frac{\sum_{\mathbf{r}\in R} (\tilde{z}_1(\mathbf{r}) - \bar{\tilde{z}}_1) \cdot (\tilde{z}_2(\mathbf{r}) - \bar{\tilde{z}}_2)}{\sqrt{\mathrm{Var}(\tilde{z}_1)}\sqrt{\mathrm{Var}(\tilde{z}_2)}}\,, \tag{12}$$

where $\bar{\tilde{z}}_1$ represents the mean of $\tilde{z}_1$.

To perform experiments, we defined, as ground truth (GT) object sequences, ellipses manually fitted to the head of persons in 2 sequences of 300 images each. Next, we considered state couples $(c_{k-1}, c_k) = (c^{gt}_{k-1}, c^{gt}_k + \vec{\delta})$, where c^{gt} denotes a GT object image position, and $\vec{\delta}$ corresponds to an offset around the GT state. Furthermore, the dimensions of the ellipse at time $k-1$ are used to define the ellipse at time k.

The dependency between measurements is illustrated in Fig. 4a and 4b, where the average NCC is ploted against the amplitude of $\vec{\delta}$, measured either in number of pixels, or in percentage of object size, where object size is defined as the average between the two ellipse's axis lengths. In the training data, object size ranges between 30 and 80 pixels, and there are between 600 and 12000 measurements per δ value. As can be seen, when the offset displacement reaches 50% of object size, correlation becomes close to 0. When the displacement is greater than 100%, the NCC should be 0 in average, as there is no more overlap between the two measurement vectors. Fig. 4c and 4d illustrates this further by displaying the histogram of the NCC for different values of δ. Again, while the histograms are peaked around 1 for small values for δ, it gradually moves towards a symmetric histogram centered at 0 with the increase of δ.

This issue bears similarities with the work on Bayesian correlation [30]. In such work, the dependence/independence of measurements (in this case, the output of a set of filters) at different spatial positions, given the object state, was studied. It was shown that independence was achieved as long as the supports of the filters were distant enough. For foreground object modeling, however, the obtained measurement distributions were not specific enough. The work in [31] further showed that the independence still holds *conditioned* on the availability of some form of object template to predict the filter output. In tracking terms, the patch extracted in the previous frame from the state at time $k-1$ plays the role of the conditioning template, as shown by Eq. (11), and the independence result of [31] states that the noise variables $\eta(\mathbf{r})$ and $\eta(\mathbf{r}')$ are independent when $|\mathbf{r}-\mathbf{r}'|$ is large enough.

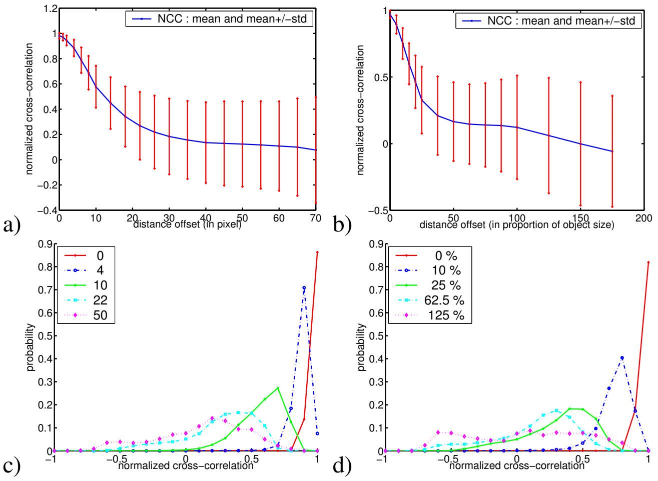


Fig. 4. (a) and (b) Average of the NCC coefficient for state couples at varying distance from the ground truth state values. In (a), the distance is measured in pixels, while in (b) it is measured in proportion of object size. (c) and (d) Empirical distribution of the NCC coefficients for different displacement distance, measured either in pixel (c), or in proportion of object size (d).

The above analysis illustrates that the independence of the data given the sequence of states is not a true assumption in general. More precisely :

$$\mathrm{p}(z_k|z_{1:k-1}, c_{0:k}) \neq \mathrm{p}(z_k|c_k)\,, \tag{13}$$

which means that we can not reduce the left hand side to the right one as usually done with the standard derivation of the particle filter equations. A more accurate model for visual tracking is thus represented by the graphical model of Fig. 1b.

The new model can be easily incorporated in the particle filter framework. First, note that all computation leading to Eq. 5 in Section III are general and do not depend on assumptions H1 and H2. Starting from there, replacing H1 by the new model gives :

$$\mathrm{p}(z_k|z_{1:k-1}, c_{0:k}) = \mathrm{p}(z_k|z_{k-1}, c_k, c^i_{k-1})\,. \tag{14}$$

If we keep H2, it is easy to see that the new weight update equation is given by :

$$w^i_k \propto w^i_{k-1} \frac{\mathrm{p}(z_k|z_{k-1}, c^i_k, c^i_{k-1})\mathrm{p}(c^i_k|c^i_{k-1})}{q(c^i_k|c^i_{0:k-1}, z_{1:k})} \tag{15}$$

in replacement of equation (6).

B. Proposal distribution and dynamical model

According to our new graphical model, and following the same arguments as in [3], [14], we can show that the optimal proposal distribution and the corresponding update rule are given by :

$$q(c_k|c^i_{k-1}, z_{1:k}) = \mathrm{p}(c_k|z_k, z_{k-1}, c^i_{k-1})$$
$$\propto \mathrm{p}(z_k|c_k, z_{k-1}, c^i_{k-1})\mathrm{p}(c_k|c^i_{k-1}) \quad (16)$$
$$\text{and} \ \ w^i_k \propto w^i_{k-1}\mathrm{p}(z_k|c^i_{k-1}, z_{k-1}) \, . \quad (17)$$

As their homologous Equations (7) and (8), these equations are difficult to be used in practice.

A possibility then consists of using the dynamical model (i.e. the prior) as the proposal. This suffers from the generic drawbacks mentioned in the introduction, and in visual tracking, from the unspecificity of some state changes, which often plays in favor of the use of simple dynamical models (e.g. constant speed models). Also, the low temporal sampling rate and the presence of fast and unexpected motions, due either to camera or object movements, render the noise parameter estimation problem difficult.

An alternative, that we adopt in this paper, consists of using as proposal a mixture model built from the *prior*, the output of several trackers [32], or observation likelihood distributions [24]. In our case, the likelihood term $\mathrm{p}(z_k|c_k, z_{k-1}, c^i_{k-1})$ comprises an object-related term and one motion term (see paragraph V-C). In this article, we will construct a proposal distribution from the latter. Moreover, as motivated by the rest of this section, this term happens to be more adapted to model state changes than dynamics relying only on state values.

The relevance of using a visual motion-based proposal rather than the dynamics is illustrated by the following experiments. Consider as state c the horizontal position of the head of the foreground person in the sequence displayed in Fig. 6, which has been hand held recorded and features a person moving around in an office, and denote by c^{gt} the GT value obtained from a manual annotation of the head position in 200 images. Furthermore, let us denote by ξ_k the state prediction error, whose expression is given by

$$\xi_k = c^{gt}_k - \hat{c}_k \, , \quad (18)$$

where $\hat{c}_k$ denotes the state prediction, computed by two methods. The first one uses a simple AR model :

$$\hat{c_k} = c^{gt}_{k-1} + \dot{c}_{k-1} \text{ with } \dot{c}_{k-1} = c^{gt}_{k-1} - c^{gt}_{k-2} \, , \quad (19)$$

where $\dot{c}$ denotes the state derivative and models the evolution of the state. In the second method, $\hat{c_k}$ is computed by exploiting the inter-frame motion to predict the new state value :

$$\hat{c}_k = c^{gt}_{k-1} + \dot{c}^{motion}_{k-1} \quad (20)$$

where $\dot{c}^{motion}_{k-1}$ is computed using the coefficients of an affine motion model robustly estimated on the region defined by c^{gt}_{k-1} (see Section V-B).

Fig. 5a reports the prediction error obtained with the AR model. As can be seen, this prediction is noisy. The standard deviation of the prediction error, σ_ξ, is equal to 2.7. Furthermore, there are large peak errors (up to 30% of the head width)[2]. To cope with these peaks, the noise variance in the dynamics has to be overestimated to avoid particles lying near the ground truth to be too disfavored. Otherwise, only particles lying near the -erroneous- predicted states may survive the resampling step. However, a large noise variance has the effect of wasting many particles in low likelihood areas or spreading them on local distractors, which can ultimately lead to tracking failures. On the other hand, exploiting the inter-frame motion leads to a reduction of both the noise variance (σ_ξ=0.83) and the error peaks (Fig. 5b).

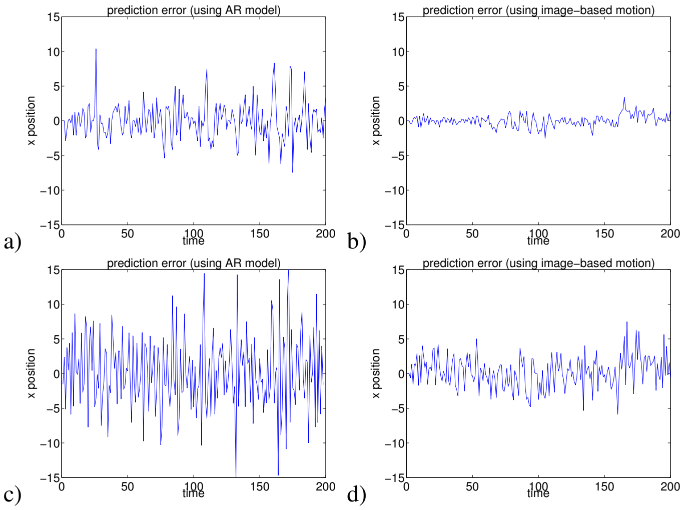


Fig. 5. (a) Prediction error of the x position, when using an AR2 model . (b) Prediction error, but exploiting the inter-frame motion estimation. (c) resp. (d), same as (a) resp. (b) but now adding a random Gaussian noise (std=2 pixels) on the GT measurements used for prediction. With the AR model (Fig. c) both the previous state and state derivative estimates are affected by noise (σ_ξ=5.6), while with visual-motion (Fig. d) the noise mainly affects the previous measurement (σ_ξ=2.3).

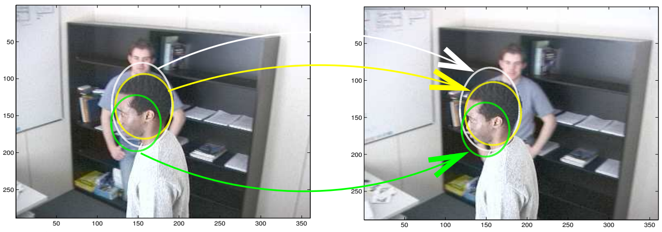


Fig. 6. Example of motion estimates between two images from noisy states. The 3 ellipses correspond to different state values. Although the estimation support regions only cover part of the head and enclose textured background, the head motion estimate is still good.

There is another advantage of using image-based motion estimates. Let us first note that the previous state values (here c_{k-1}, c_{k-2}) used to predict the new state value $\hat{c}_k$ are affected by noise, due to measurement errors and uncertainty. Thus, in the standard AR approach, both the state c_{k-1} and state derivative $\dot{c}_{k-1}$ in Eq. 19 are affected by this noise, resulting in large errors (Fig. 5c). When using the inter-frame motion estimates, the estimation $\dot{c}^{motion}_{k-1}$ is almost not affected by noise (whose effect is to slightly modify the support region used to estimate the motion), as illustrated in Fig. 6, resulting again in a lower noise variance process (Fig. 5d).

Thus, despite needing more computation resources, inter-frame motion estimates are usually more precise than autoregressive models to predict new state values; as a conse-

[2] Higher order models were also tested. Although they usually led to a variance reduction of the prediction error, they also increased the amplitude and duration of the error peaks.

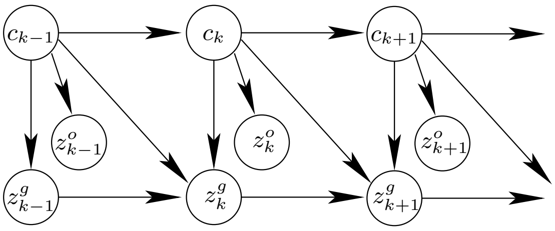


Fig. 7. Specific graphical model for our implementation.

quence, they are a better choice when designing a proposal function. This observation is supported by experiments on other state parameters -vertical position, scale-, and on other sequences. Finally, this observation can also be applied to a set of particles. If these are localized on modes of a distribution related to visual measurements, their prediction according to the visual motion will generally place them around the new modes associated with the current image.

V. The implemented model

The graphical model of Fig. 1b is generic. In this paper, our specific implementation will be based on the graphical model of Fig. 7, whose elements are described more precisely in the rest of this section.

A. Object representation and state space

We follow an image-based standard approach, where the object is represented by a region R centered at the coordinates origin, subject to some valid geometric transformation, and characterized either by a shape or by a color distribution. In the experimental Section, we illustrate and evaluate our method on face tracking sequences, which uses an elliptical region as object region R. For geometric transformations, we have chosen a subspace of the affine transformations comprising a translation $\mathbf{T} = (\mathrm{T}_x, \mathrm{T}_y)$, a scaling factor s, and an aspect ratio e :

$$\mathcal{T}_\alpha \mathbf{r} = \begin{pmatrix} \mathrm{T}_x + x s_x \\ \mathrm{T}_y + y s_y \end{pmatrix} , \qquad (21)$$

where $\mathbf{r} = (x, y)$ denotes a point position in the reference frame, $\alpha = (\mathbf{T}, s, e)$, and :

$$s = \frac{s_x + s_y}{2} \, , \; e = \frac{s_x}{s_y} \, , \; s_x = \frac{2es}{1+e} \; \text{ and } \; s_y = \frac{2s}{1+e} \qquad (22)$$

A state is then defined as $c_k = (\alpha_k, \alpha_{k-1})$. Note that we did not employ a rotation parameter in our state-space. This is due to the fact that an elliptical region remains globally almost unchanged under rotation when the aspect ratio is close to one. Thus, the estimation of such parameter would be rather underconstrained given the object likelihood models we will employ (edge measurements or color histograms). In addition, increasing the size of the state-space makes the sampling more difficult, without any particular benefits in our case. Nevertheless, with other shapes, or in other applications, the use of the rotation parameter -as well as other parameters- might be necessary, and the methodology provided below could then easily be adapted.

B. Motion estimation

As mentioned in the previous Section, we use inter-frame motion estimates both as observations and to sample the new state values. More precisely, we estimate an affine displacement model $\vec{d}_\Theta$ parameterized by $\Theta = (a_i), i = 1..6$ and defined by:

$$\vec{d}_\Theta \mathbf{r} = \begin{pmatrix} a_1 + a_2 x + a_3 y \\ a_4 + a_5 x + a_6 y \end{pmatrix} . \qquad (23)$$

Such a model, though less general than full 3D ones, represents a good compromise between intra-frame motion modeling and the efficiency of its estimation.
The estimation of the parameter Θ relies on a gradient-based multiresolution robust estimation method described in [33][3]. To ensure the goal of robustness, we minimize an M-estimator criterion with a hard-redescending function [34]. The constraint is given by the usual assumption of brightness constancy of a projected surface element over its 2D trajectory [35]. As displacements between two frames can be large, we use a discrete formulation of this constraint. Thus, the estimated parameter vector is defined as:

$$\widehat{\Theta}(c_{k-1}) = \operatorname*{argmin}_{\Theta} \; E(\Theta) = \operatorname*{argmin}_{\Theta} \sum_{\mathbf{r} \in R(c_{k-1})} \rho \left(\mathrm{DFD}_\Theta(\mathbf{r}) \right) \quad (24)$$

$$\text{with} \quad \mathrm{DFD}_\Theta(\mathbf{r}) = I_k(\mathbf{r} + \vec{d}_\Theta \mathbf{r}) - I_{k-1}(\mathbf{r}) , \qquad (25)$$

where I_{k-1} and I_k are the images, and $\rho(\cdot)$ is a robust estimator, bounded for high values of its argument (specifically, we use Tukey's biweight function). The minimization takes advantage of a multiresolution framework and an incremental scheme based on the Gauss-Newton method. More precisely, at each incremental step l (at a given resolution level, or from a resolution level to a finer one), we have: $\Theta = \widehat{\Theta}_l + \Delta\Theta_l$. Then, a linearization of $\mathrm{DFD}_\Theta(\mathbf{r})$ around $\widehat{\Theta}_l$ is performed, leading to a residual quantity $\mathrm{res}_{\Delta\Theta_l}(\mathbf{r})$ linear w.r.t. $\Delta\Theta_l$:

$$\mathrm{res}_{\Delta\Theta_l}(\mathbf{r}) = \vec{\nabla} I_k(\mathbf{r} + \vec{d}_{\widehat{\Theta}_l} \mathbf{r}) \vec{d}_{\Delta\Theta_l} \mathbf{r} + I_k(\mathbf{r} + \vec{d}_{\widehat{\Theta}_l} \mathbf{r}) - I_{k-1}(\mathbf{r}) \quad (26)$$

where $\vec{\nabla} I_k(\mathbf{r})$ denotes the spatial gradient of the intensity function at location $\mathbf{r}$ and at time k. Finally, we substitute for the minimization of $E(\Theta)$ the minimization of an approximate expression E_a, which is given by $E_a(\Delta\Theta_l) = \sum \rho(\mathrm{res}_{\Delta\Theta_l}(\mathbf{r}))$. This error function is minimized using an Iterative-Reweighted-Least-Squares procedure, with 0 as an initial value for $\Delta\Theta_l$. For more details about the method and its performances, the reader is referred to [33].
This algorithm allows us to get a robust and accurate estimation of the motion model. Owing to the robustness of the estimator, an imprecise definition of the region $R(c_{k-1})$ involved in Eq.24 due to a noisy state value does not sensibly affect the estimation (see Fig. 6). From these motion estimates, we can measure the variation α^m_{k-1} of our state-space coefficients between the two instants. Assuming that the coordinates in Eq. 23 are expressed with respect to the object center (according to the definition of $\mathcal{T}$, translated from

[3] We use the code available at http://www.irisa.fr/vista

the origin to the position $\mathbf{T}$ in the image), we propose the following derivative estimates :

$$\begin{cases} \dot{\mathrm{T}}_x = a_1 \\ \dot{\mathrm{T}}_y = a_4 \end{cases}, \begin{cases} \dot{s}_x = a_2 s_x \\ \dot{s}_y = a_6 s_y \end{cases} \text{ and } \begin{cases} \dot{s} = \frac{s}{1+e}(a_2 e + a_6) \\ \dot{e} = e(a_2 - a_6) \end{cases} \tag{27}$$

Thus, the measure of the parameter variations can be defined as $\alpha^m = (\dot{\mathrm{T}}_x, \dot{\mathrm{T}}_y, \dot{s}, \dot{e})$. Additionally, the value of the predicted geometric parameters, denoted by α^p, is then given by:

$$\alpha_k^p = \alpha_{k-1} + \alpha_{k-1}^m \tag{28}$$

Although not used in the reported experiments, the covariance matrix of the estimated parameters can also be computed. With model-based approaches involving more state parameters, this would be useful to account for uncertainty and underconstrained optimization.

C. Data likelihood modeling

To implement the new particle filter, we assume that the measurements z_k are of two types: object measurements z_k^o (i.e. edges or color), and patch gray level measurements z_k^g. Then, we consider the following data likelihood :

$$\begin{aligned} \mathrm{p}(z_k|z_{k-1}, c_k, c_{k-1}) &= \mathrm{p}(z_k^o, z_k^g | z_{k-1}^o, z_{k-1}^g, c_k, c_{k-1}) \\ &= \mathrm{p}(z_k^o | z_k^g, z_{k-1}^o, z_{k-1}^g, c_k, c_{k-1}) \mathrm{p}(z_k^g | z_{k-1}^o, z_{k-1}^g, c_k, c_{k-1}) \\ &= \mathrm{p}(z_k^o|c_k) \mathrm{p}(z_k^g | z_{k-1}^g, c_k, c_{k-1}) \end{aligned} \tag{29}$$

where the last derivations exploit the properties of the graphical model of Fig. 7. Two assumptions were made to derive this model. The first one assumed that object observations are independent of patch observations given the state sequence measurements. This choice decouples the model of the dependency existing between two images, whose implicit goal is to ensure that the object trajectory follows the optical flow field implied by the sequence of images, from the shape or appearence object model. When the object is modeled by a shape, our assumption is valid since shape observations will mainly involve measurements on the border of the object, while the correlation term will apply to the regions inside the object. When a color representation is employed, the assumption is valid as well, as color measurements can usually be considered as being independent of gray-scale measurements. The second assumption we made is that object measurements are uncorrelated over time. When considering shape measurements, the assumption is quite valid as the temporal auto-correlation function of contours is peaked. However, with the color representation [5], [8], the temporal independence assumption might not hold. Better models need to be searched for to handle this case.

We describe the specific observations models as follows.

1) **Visual object measurement***:* For the experiments, we considered both contour models or color models.

Shape model :

The observation model assumes that objects are embedded in clutter. Edge-based measurements are computed along L normal lines to a hypothesized contour, resulting for each line l in a vector of candidate positions $\{\nu_m^l\}$ relative to a point lying on the contour ν_0^l. With some usual assumptions [2], the shape likelihood can be expressed as

$$\mathrm{p}(z_k^o|c_k) \propto \prod_{l=1}^{L} \max\left(K_{sh}, \exp(-\frac{\|\hat{\nu}_m^l - \nu_0^l\|^2}{2\sigma_{sh}^2})\right), \tag{30}$$

where $\hat{\nu}_m^l$ is the nearest edge on l, and K_{sh} is a constant used when no edges are detected.

Color model :

As color models we used color distributions represented by normalized histograms in the HSV space and gathered inside the candidate region $R(c_k)$ associated with the state c_k. To be robust to illumination effects, we only considered the HS values. Then, a normalized multidimensional histogram was computed, resulting in a vector $\mathbf{b}(c_k) = (\mathrm{b}^j(c_k))_{j=1..N}$, where $N = N_h \times N_s$ with N_h and N_s representing the number of bins along the hue and saturation dimensions respectively ($N_h = N_s = 8$), and where the index j corresponds to a couple (h, s) with h and s denoting hue and saturation bin numbers. At time k, the candidate color model $\mathbf{b}(c_k)$ is compared to a reference color model $\mathbf{b}_{ref}$. As a distance measure, we employed the Bhattacharyya distance measure [5], [8]:

$$D_{bhat}(\mathbf{b}(c_k), \mathbf{b}_{ref}) = \left(1 - \sum_{j=1}^{N} \sqrt{\mathrm{b}^j(c_k)\mathrm{b}_{ref}^j}\right)^{1/2} \tag{31}$$

and assumed that the probability distribution of the square of this distance for a given object follows an exponential law,

$$\mathrm{p}(z_k^o|c_k) \propto \exp\{-\lambda_{bhat} \, D_{bhat}^2(\mathbf{b}_k(c_k), \mathbf{b}_{ref})\}\,. \tag{32}$$

We used the histogram computed in the first frame as reference model, which implicitely assumes that the color distribution has to remain constant throughout the sequence. This is a reasonable assumption when dealing with cases when lighting does not change dramatically over time, and color distributions are known to be robust to deformation of the object [5], [8]. However, in more complex situations, it might be useful to employ several reference distributions to model completely different object appearances (e.g. face seen from front or back), or to use online adaptation [6], [21].

2) **Image correlation measurement***:* To model this term, we used two possibilities :

- The first one consists of extracting measures in the parameter space. Usually, this is achieved by thresholding and/or extracting local maxima of some interest function [24], [27]. In our case, this corresponds to the extraction of peaks of a correlation map, as done in [27] for translations. One advantage of such a method is to provide a well-behaved likelihood (i.e. involving only a few well identified modes). One drawback is that the extraction process can be time consumming.
- In the second approach, gray-level patches are directly compared after having warped them according to the state values (see Eq.(10)). The advantages of this method are to supply more "detailed" likelihoods that can be computed directly from the data.

In this paper, we employ both options, by assuming that observations are made of the measured parameter variations α^m_{k-1} obtained using the estimated motion, and of the local patches $\tilde{z}^g_{c_k}$. We model the correlation term as :

$$\mathrm{p}(z^g_k|z^g_{k-1}, c_k, c_{k-1}) \propto \mathrm{p}_{c1}(\alpha^m_{k-1}, \alpha_k, \alpha_{k-1})\mathrm{p}_{c2}(\tilde{z}^g_{c_k}, \tilde{z}^g_{c_{k-1}}) \tag{33}$$

To model the first term, we assume the following measurement equation:

$$\alpha^m_{k-1} = \alpha_k - \alpha_{k-1} + noise \tag{34}$$

Given both the previous and current state values, and assuming a Gaussian noise, the pdf of this measurement is given by:

$$\begin{aligned}\mathrm{p}_{c1}(\alpha^m_{k-1}, \alpha_k, \alpha_{k-1}) &= \mathcal{N}(\alpha^m_{k-1}; \alpha_k - \alpha_{k-1}, \Lambda_{\xi_p}) \\ &= \mathcal{N}(\alpha^p_k; \alpha_k, \Lambda_{\xi_p})\end{aligned} \tag{35}$$

where $\mathcal{N}(.;\mu,\Lambda)$ represents a Gaussian distribution with mean μ and covariance matrix Λ, $\Lambda_{\xi_p} = diag(\sigma_{\xi^2_p,j})$ is the covariance of the measurements, and the derivation of the last expression in the equation exploits Eq. 28. The second term in Eq. 33 is modeled by:

$$\mathrm{p}_{c2}(\tilde{z}^g_{c_k}, \tilde{z}^g_{c_{k-1}}) = \mathrm{Z}^{-1} \exp^{-\lambda_{cor} D_c{}^2(\tilde{z}^g_{c_k}, \tilde{z}^g_{c_{k-1}})} \tag{36}$$

$$\mathrm{Z} = \int_{z',z''} \exp^{-\lambda_{cor} D_c{}^2(z',z'')} \mathrm{d}z' \mathrm{d}z'' \tag{37}$$

where D_c denotes a distance between two image patches, Z is a normalization constant whose value can be computed from (37), where the integral runs over pairs of consecutive patches corresponding to the same tracked object extracted in training sequences [13]. In practice, however, we did not compute this value and assumed it to be constant for all object patches. The first probability term in Eq. 33 compares the predicted parameters with the sampled values using a Gaussian noise process (cf last expression in Eq. 35). The second term introduces a non-Gaussian model, by comparing directly the patches defined by c_k and c_{k-1} using the similarity distance D_c. It has been derived by assuming that all patches are equally probable. Although the use of those two terms is somewhat redundant, it proved to be a good choice in practice and its purpose can be illustrated using Fig. 6. While all the three predicted configurations will be weighted equally according to p_{c1}, the second term p_{c2} will downweight the two predictions (green and white ellipses) whose corresponding support region is covering part of the background, which is undergoing a different motion than the head.

The definition of p_{c2} requires the specification of a patch distance. Many such distances have been defined and used in the literature [13], [15], [22]. The choice of the distance should take into account the followings considerations :

1) the distance should still model the underlying motion content, i.e. the distance should increase as the error in the predicted configuration grows;
2) the random nature of the prediction process in the SMC filtering will rarely produce configurations corresponding to exact matches. This is particularly true when using a small number of samples;
3) particles covering both background and object, each undergoing different motions, should have a low likelihood.

For these purposes, we found out in practice that it was preferable not to use robust norms such as L1 saturated distance or a Haussdorf distance [13]. Additionally, we needed to avoid distances which might *a priori* favor patches with specific content. This is the case of the L2 distance, which corresponds to an additive Gaussian noise model in Eq.(11) and generally provides lower scores for tracked patches with large uniform areas[4]. Instead, we used a distance based on the normalized-cross correlation coefficient (Eq. (12)) defined as :

$$D_c(\tilde{z}_1, \tilde{z}_2) = 1 - \mathrm{NCC}(\tilde{z}_1, \tilde{z}_2) \tag{38}$$

Regarding the above equation, it is important to emphasize again that the method is not performing template matching, as in [15]. No object template is learned off-line or defined at the begining of the sequence, and the tracker does not maintain a single template object representation at each instant of the sequence. Thus, the correlation term is not object specific (except through the definition of the reference region R). A particle placed on the background would thus receive a high weight if the predicted motion is in adequation with the background motion. Nevertheless, the methodology can be extended to be more object dependent, by using more object specific regions R and by allowing the region R to vary over time, as is done in articulated object tracking [22].

D. Dynamics definition

To model the prior, we use a standard second order AR model (Eq. 19) for each of the components of α. However, to account for outliers (i.e. unexpected and abrupt changes) and reduce the sensitivity of the prior in the tail, we model the noise process with a Cauchy distribution, $\rho_c(x,\sigma^2) = \frac{\sigma}{\pi(x^2+\sigma^2)}$. This leads to

$$\mathrm{p}(c_k|c_{k-1}) = \prod_{j=1}^{4} \rho_c\left(\alpha_{k,j} - (2\alpha_{k-1,j} - \alpha_{k-2,j}), \sigma_{\xi^2_d,j})\right). \tag{39}$$

where $\sigma_{\xi^2_d,j}$ denotes the dynamics noise variance of the j^{th} component.

E. Proposal distribution

As motivated in Section IV-B, the definition of the proposal function $q(c_k|c^i_{0:k-1}, z_{1:k})$, given a past trajectory $c^i_{0:k-1}$, relies on the estimated motion. More precisely, a new state sample $c_k = (\alpha_k, \alpha_{k-1})$ is drawn by letting $\alpha_{k-1} = \alpha^i_{k-1}$, and drawing α_k from $q(\alpha_k|\alpha^i_{k-1}, z_k, z_{k-1})$, defined by:

$$q(\alpha_k|\alpha^i_{k-1}, z_k, z_{k-1}) = \mathcal{N}(\alpha_k; \alpha^p_k(\alpha^i_{k-1}), \Lambda_{\xi_p}) \tag{40}$$

which means that we sample new transform parameters around the predicted value. Note that, as done by others [24], [25], [32], we could have defined our proposal as a mixture, with, in our case, the prior model and the above proposal as

[4] This issue is related to our assumption of equally probable patches. Given our likelihood model for joint tracked patches, Eq. (36), this assumption is only approximate.

components. Such an approach would be interesting when the motion estimation process could be susceptible to failures, e.g. when tracking small or textureless objects, or in cases of strong or total occlusion (assuming in this case that the likelihood modeling can handle such a situation). Similarly, these failure conditions might be partially handled by exploiting the covariance matrix of the estimated motion parameters, which would normally exhibit large values in such situations. The proposal covariance matrix values Λ_{ξ_p} could be increased to reflect such cases. However, this failure conditions will not be our case. Besides, using only the visual motion proposal along with a fixed covariance matrix will allow us to better illustrate its contribution to the tracking performance.

VI. Results

In this Section, we first describe the different tracker models evaluated and the parameterization we used. We then present qualitative and quantitative results on five different sequences involving head tracking. Visual results should be appreciated by looking directly at typical video results that can be found on our website[5].

A. Trackers and setup

To differentiate the different elements of the model, we considered three kinds of trackers :

- condensation tracker M1: this tracker corresponds to the standard CONDENSATION algorithm [2], with the object likelihood p_o (Eq. 30 or 32) combined with the same AR model with Gaussian noise for the proposal and the prior.
- implicit correlation tracker M2: it corresponds to CONDENSATION, with the addition of the implicit motion likelihood term in the likelihood evaluation (i.e now equal to $\mathrm{p}_o.\mathrm{p}_{c2}$). This method does not use explicit motion measurements.
- motion proposal tracker M3: it is the full model. The samples are drawn from the motion proposal, Eq. 40, and the weight update is performed using Eq. 5. After simplification, the update equation becomes :

$$w_k^i \propto w_{k-1}^i \; \mathrm{p}_o(z_k^o|c_k^i)\mathrm{p}_{c2}(\tilde{z}^g_{c_k^i}, \tilde{z}^g_{c_{k-1}^i})\mathrm{p}(c_k^i|c_{k-1}^i) \quad (41)$$

  For this model, the motion estimation is not performed for all particles since it is robust to variations of the support region. At each time, the particles are clustered into K_m clusters. The motion is estimated using the mean of each cluster and exploited for all the particles of the cluster.
- deterministic robust motion tracker M4: this tracker, whose state-space is the same as for the preceeding ones, works as follows. At time t, given the current value of the state, an affine motion model is estimated, as described in SubsectionV-B, and exploited to predict the value of the state at time $t+1$, as given by Eq. 27 and 28.

For 200 particles, the shape-based M1 tracker runs in real time (on a 2.5GHz P IV machine), M2 at around 20 image/s, and M3 around 8 image/s. Tracker M4 runs in real time.

parameters	L	σ_{sh}	K_{sh}	λ_{bhat}	λ_{cor}	K_m
values	16	5	$\exp^{-2}$	20	20	$\max(20, \frac{N_s}{10})$

TABLE I
PARAMETER SETTING.

B. Parameter setting

As in any other tracking algorithms, we need to set the value of several parameters, whose choice can have an influence on the results. In this paper, we decided to evaluate the sensitivity of the results to the most influential parameters in our opinion: the noise parameters in the dynamical and proposal model and the number of particles N_s, while keeping all other parameters fixed.

The values of the common parameters are given in Table I. They were chosen based on previous experience ([25]), and in accordance with the values found in other works [2], [8]. While these parameters are by no means universal, they are sensible for many applications.

For the shape likelihood, we used the same parameters as in [25], which dealt with the audio-visual tracking of human-heads in a meeting room. The number of search lines L is related to the independence assumption of edge measurements, which is itself dependent on the expected size of the object in the image. With a too large number of lines, neighboor measures will be correlated, which would violate the independence hypothesis, while a too small number would result in a poor modeling of the shape. The σ_{sh} parameter relates to the precision of our modeling of a head contour as an ellipse, where a small value would assume that head is perfectly elliptical. More generally, this term has a direct influence on the landscape form of the shape likelihood function: with a small value, this function will exhibit sharper modes, with a higher selectivity with respect to the tracked object, but also less chances for the particle filter to keep track of several modes, and higher chances of locking onto erroneous distractors. In practice, we found that values ranging from 4 to 8 were adequate and did not affect importantly the results. Given the chosen value of K_{sh}, of σ_{sh}, and the specific form of the likelihood, Eq. 30, the utility search range along each line is 10 pixels inside and outside the contour. The parameter K_{sh} can be related to the probability of both not detecting a contour despite being in a correct configuration (e.g. due to the absence of contrast), and randomly detecting a contour anywhere along the search line (e.g. due to noise) [2]. Small values of K_{sh} lead to a shape likelihood less tolerant to the occurence of the above events while large values lead to a less discriminative likelihood in good conditions.

The selected value of the color parameter λ_{bhat} was the same as in [8], which used a similar discretization of the color space, and validated by experience. As for the σ_{sh} parameter, λ_{bhat} acts directly on the sharpness of the likelihood, and the same comment applies. As the correlation distance is in the same range and behaves similarly to the Bhattacharyya distance, we used the same value as λ_{bhat} for λ_{cor}. Finally, for K_m, we did not thoroughly test other values as the current one working reasonably. In practice, it might be interesting to test lower values to save computational cost.

[5] *www.idiap.ch/∼odobez/IPpaper/EmbeddingMotion.html*

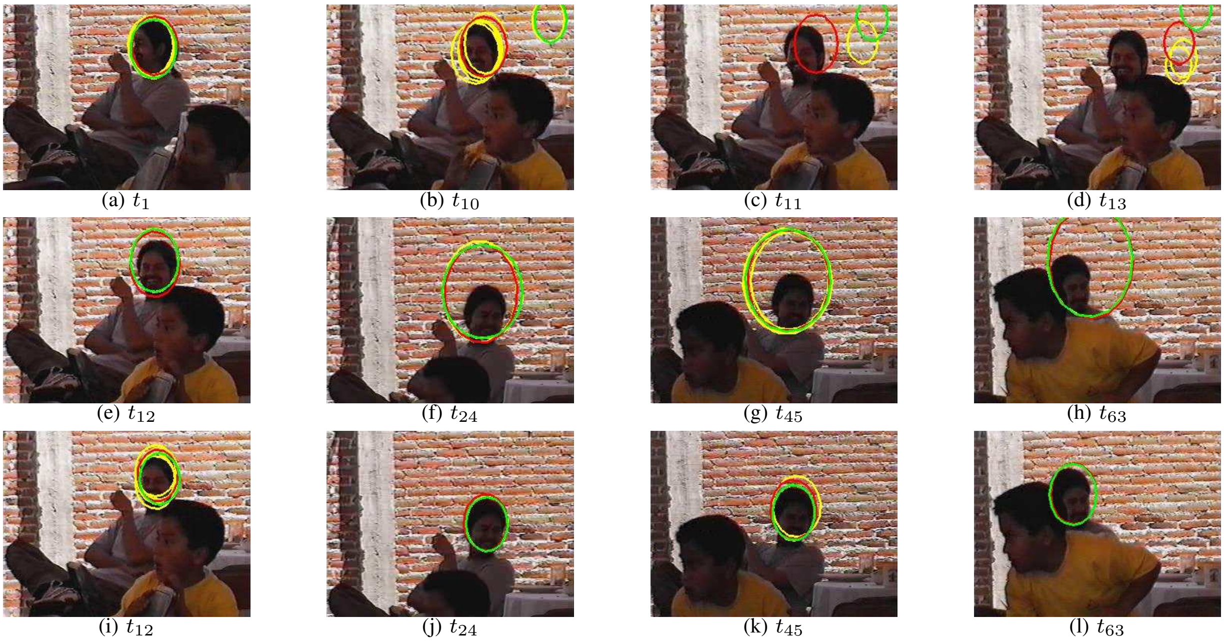


Fig. 8. Head tracking 1 : first row : shape-based tracker M1. Second row : shape-based tracker M2. Third row : shape-based tracker M3. In dark gray (red), mean state; in medium gray (green), mode state; in light gray (yellow), likely particles.

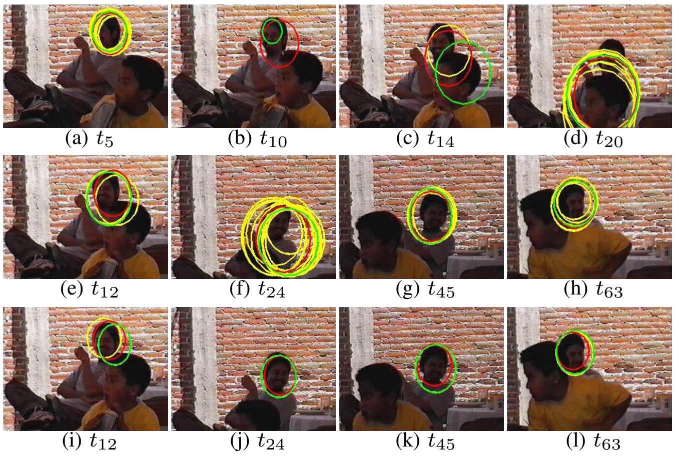


Fig. 9. Head tracking sequence 1 : color-based model. First row : M1, second row : M2, last row : M3. All experiments (including those of Fig. 8) with N_s=250 and $(\sigma_{\mathbf{T}}, \sigma_s) = (5, 0.01)$. In dark gray (red), mean state; in medium gray (green), mode state; in light gray (yellow), likely particles.

Finally, for the dynamic components, we will use the following values. First, as the motion proposal term is more reliable than the prior to constraint trajectories, we will set the prior noise $\sigma_{\xi_d,j}$ to three times the proposal noise $\sigma_{\xi_p,j}$ in the M3 tracker. In all experiments, the noise standard deviations in the proposal distribution (the Gaussian prior in M1 and M2, the motion proposal, Eq. 40, in M3) will be denoted $\sigma_{\mathbf{T}}$ for each of the translation components and σ_s for the scale parameter. The aspect ratio noise component is kept fixed, with a value of 0.01.

C. Tracking results

Sequence 1: The first sequence (Fig. 8 and 9), containing 64 images of size 240×320, illustrates qualitatively the benefit of the method in the presence of strong ambiguities. The sequence features a highly textured background producing very noisy shape measurements, camera and head motion, change of appearence of the head, and partial occlusion. Whatever the number of particles or the noise variance in the dynamical model, the shape-based tracker M1 alone is unable to perform a correct tracking after time t_{12}. In contrast, tracker M2 is able to do the tracking correctly on a large majority of runs when using small dynamics ($(\sigma_{\mathbf{T}}, \sigma_s) = (1, 0.005)$). However, with an increase of the noise variance, it fails (see second row of Fig. 8) : the observations are clearly multimodal, and the head motion is only occasionaly different from the background, which makes it especially hard for the correlation term to keep configurations enclosing only the head. Using tracker M3, however, leads to correct tracking, even with large noise values. There might be two reasons for this. The first one consists of the use of the correlation likelihood measure in parameter space. The second one is due to its ability to better maintain multimodality[6]. Consider a mode that is momentarily represented by only a few particles. With a “blind” proposal, these particles are spread with few chances to hit the object likelihood mode, decreasing their probability of survival in the next selection step. On the other hand, with the motion proposal, these chances are increased. Considering now color-based trackers, we observe that M1 usually succeeds for small dynamics but fails with standard dynamics (e.g. dynamics used in [24]), as shown in the first row of Fig. 9). This is due to the presence of the brick color and more importantly, the face of the boy. Exploiting correlation leads to successful tracking, but with a lower precision when using M2 (see images 9(e)

[6] In [36], it has been shown on simulated experiments that even when the true density is a two Gaussian mixture model with the same mixture weight for each Gaussian, and with the appropriate likelihood model, the standard particle filter loses rapidly one of the modes.

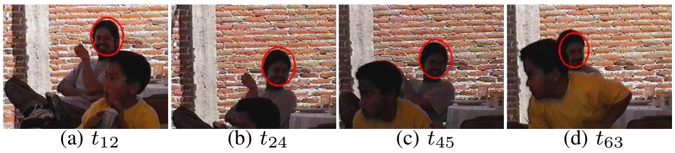
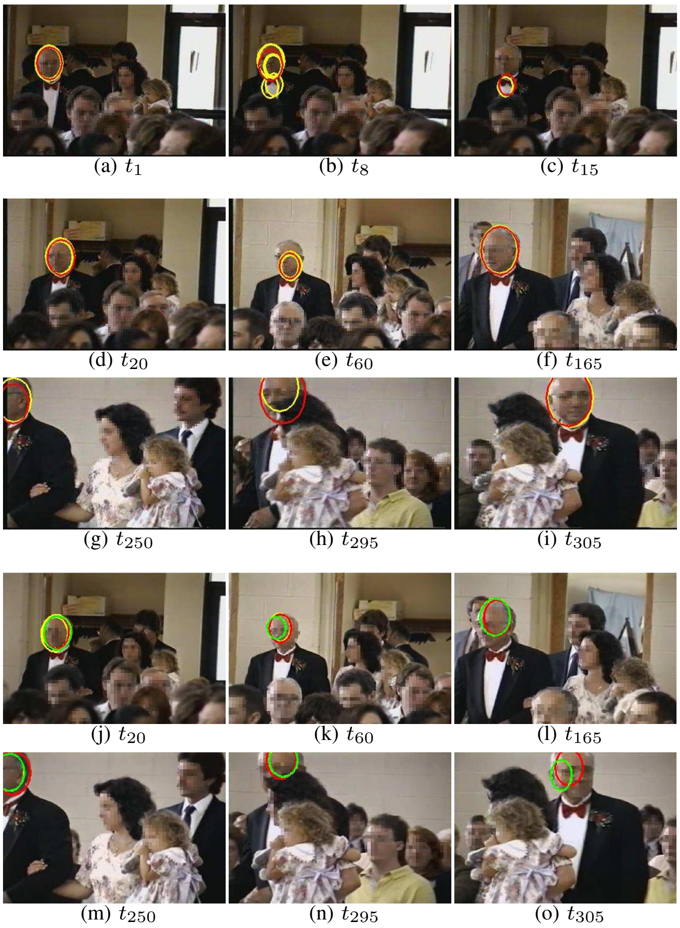

(a) t_{12} (b) t_{24} (c) t_{45} (d) t_{63}

Fig. 10. Head tracking sequence 1: deterministic motion tracker M4.


(a) t_1 (b) t_8 (c) t_{15}

(d) t_{20} (e) t_{60} (f) t_{165}

(g) t_{250} (h) t_{295} (i) t_{305}

(j) t_{20} (k) t_{60} (l) t_{165}

(m) t_{250} (n) t_{295} (o) t_{305}

Fig. 11. Head tracking sequence 2 (N_s=500) : top row : shape-based tracker M1. Second and third rows : shape-based tracker M2. Last two rows : color-based tracker M2. In dark gray (red), mean shape. In light gray (yellow), highly likely particles.

to 9(h)), than with M3 (images 9(i) to 9(l)). Finally, as shown in Fig.10, the deterministic motion tracker works well. This is due to the short length of the sequence, the presence of enough structure in the tracked object, and the precision of the estimator.

Sequence 2: The second sequence is a 330 frame sequence (Fig. 11) extracted from a hand-held home video. Figure 13 reports the tracking performance of the three first trackers for different dynamics and number of particles. At each frame, the resulting tracked region R_t (obtained from the mean state value) is considered as successful if the recall and precision are both higher than 25%, where these rates are defined by :

$$R_{\cap,t} = R_{gt,t} \cap R_t \, , r_{prec} = \frac{|R_{\cap,t}|}{|R_t|} \, , \, r_{rec} = \frac{|R_{\cap,t}|}{|R_{gt,t}|} \qquad (42)$$

where $R_{gt,t}$ is the ground truth region, and $| \cdot |$ denotes the set cardinality operator. Despite being low, the selected rate of 25% is sufficient to identify tracking failures, which is the goal of this study. Once the tracking status is established, tracking precision could be assessed with various measures [37].

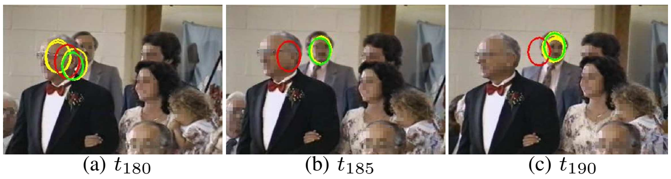
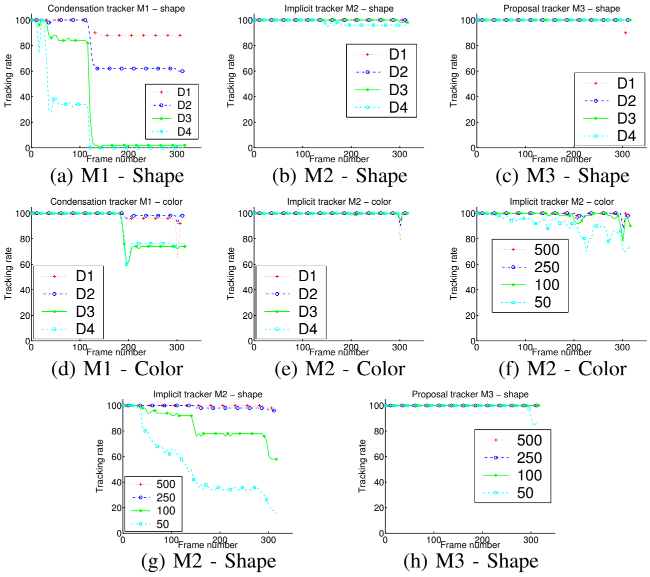

(a) t_{180} (b) t_{185} (c) t_{190}

Fig. 12. Head tracking sequence 2 (N_s=500) : failure case with the color-based tracker M1.



(a) M1 - Shape (b) M2 - Shape (c) M3 - Shape

(d) M1 - Color (e) M2 - Color (f) M2 - Color

(g) M2 - Shape (h) M3 - Shape

Fig. 13. Head tracking sequence 2 : successful tracking rate (in %, computed over 50 trials with different random seeds). Experiments (a) to (e) : parameter sets D1 to D4 correspond to N_s=500, with dynamics (σ_T, σ_s) : D1 (2,0.01), D2 (3,0.01), D3 (5,0.01) D4 (8,0.02). Experiments (f) to (h), different number of particles are tested (500/250/100/50) using the D3 (5,0.01) noise values.

As can be seen, while the shape-based tracker M1 performs quite well for tuned dynamics (parameter set D1), it breaks down rapidly, even for slight increases of dynamics variances (parameters D2 to D4). Fig. 11 illustrates a typical failure due to the small size of the head at the begining of the sequence, the low contrast at the left of the head, and the clutter. On the other hand, the shape-based tracker M2 performs well under almost all circumstances, showing its robustness against clutter, partial measurements (around time t_{250}) and partial occlusion (end of the sequence). Only when the number of samples is low (see Fig. 13(g)) does the tracker fail. These failures are occuring at different parts of the sequence. Finally, in all experiments, the shape-based tracker M3 produces a correct tracking rate. When looking at the color-based tracker M1, we can see that it performs much better than its shape equivalent (compare Fig. 13(d) and 13(a)). However, due to the presence of a person in the background, it fails around 25% of the time with standard noise values as illustrated in Fig.12. Incorporating the motion leads to perfect tracking, though leading to less precisely located estimates than in the shape case (see Fig.11(j) to Fig.11(o)). Besides, with a very small number of samples (N_s=50, see 13(f)), the M2 tracker

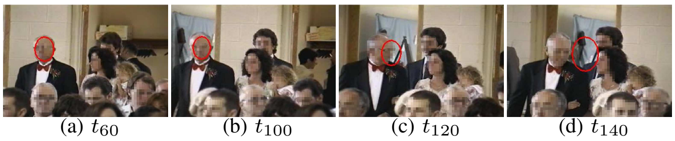
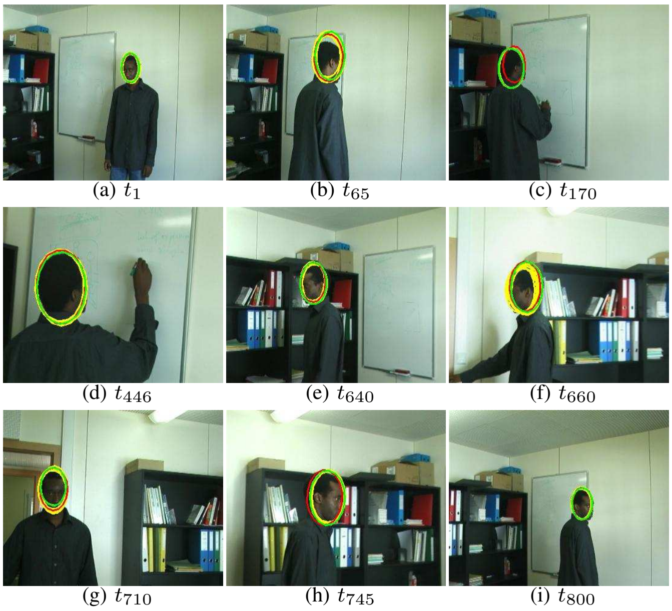

(a) t_{60} (b) t_{100} (c) t_{120} (d) t_{140}

Fig. 14. Head tracking sequence 2: deterministic motion tracker M4.


(a) t_1 (b) t_{65} (c) t_{170}

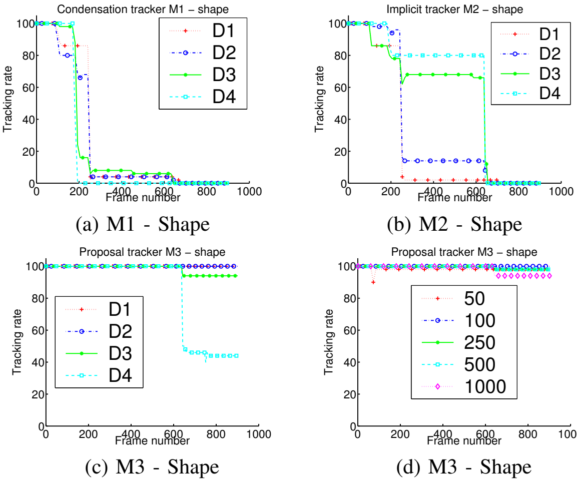
(d) t_{446} (e) t_{640} (f) t_{660}

(g) t_{710} (h) t_{745} (i) t_{800}

Fig. 15. Head tracking sequence 3. Tracker with motion proposal (N_s=500). In dark gray (red), mean shape; in medium gray (green), mode shape; in light gray (yellow), likely particles.

sometimes fails while the full model is always successful. Finally, Fig. 14 displays some images of the deterministic M4 tracker, and shows that the robust motion tracker is accurate enough to follow the head for more than 100 frames. However, as the man turns his head sideways, the motion estimator tracks the frontal part of the face as it oughts, which pushes the tracked region over the background and lead to failure. This sequence illustrates clearly that, while the motion is useful for short term tracking, an object model is necessary to avoid drifting. Adding such an object model to the motion component raises the issue of the fusion of these two information sources, an issue to which we provide a solution in this paper.

Sequence 3: The third sequence (Fig. 15) better illustrates the benefit of using the motion proposal approach. This 72s sequence acquired at 12 frame/s is specially difficult because of the occurence of several head turns[7](which prevents us from using the color trackers), and abrupt motion changes (translations, zooms in and out), and importantly, due to the absence of head contours as the head moves near (frames 160 to 200) or in front of the bookshelves (frames 620 to the end). Because of these factors, the shape-based tracker M1 fails due to a local ambiguity with the whiteboard frame (around frame 65), or because of camera jitter (frame 246) (cf Fig.16(a)). The M2 tracker works better, handling correctly the jitter situation when the dynamic noise is large enough, but fails when the head moves in front of the bookshelves, due to the temporally lack of head contours, combined with background clutter. In contrast, all these issues are resolved by the M3 tracker, which better capture the state variations, and allows a successful track of the head until the end of the sequence under almost all conditions (Fig. 16(c) and 16(d)). Figure 17 displays images of the result obtained with the M4 tracker. This tracker perfectly tracks the head until the first head turn, which generates some drift error. However, owing to the robustness of the estimator, as explained in Fig. 6, the tracker still partially follows the head, until a complete failure happens, as the drift becomes too large and the textured content of the background region dominates in the tracked region.



(a) M1 - Shape (b) M2 - Shape

(c) M3 - Shape (d) M3 - Shape

Fig. 16. Head tracking sequence 3 : successful tracking rate (in %, computed over 50 trials with different random seeds). Experiments 16(a) to 16(c) : parameter sets D1 to D4 correspond to N_s=1000, with dynamics $(\sigma_{\mathbf{T}}, \sigma_s)$: D1 (2,0.01), D2 (3,0.01), D3 (5,0.01) D4 (8,0.02). In experiments 16(d), different number of particles are tested using the D3 (5,0.01) noise values.

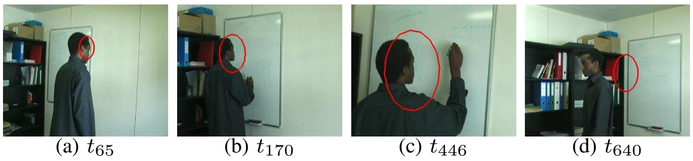

(a) t_{65} (b) t_{170} (c) t_{446} (d) t_{640}

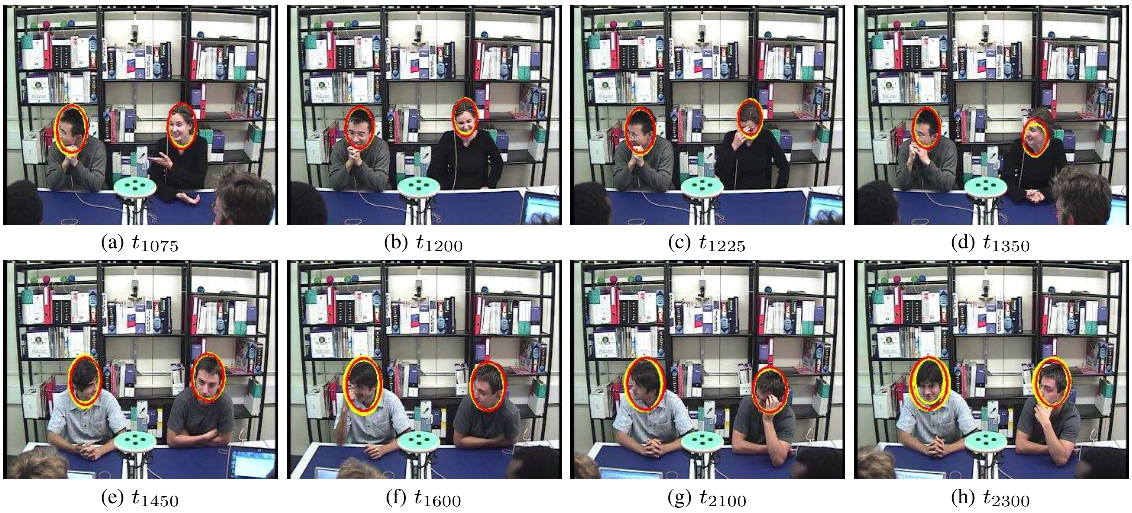
Fig. 17. Head tracking sequence 3: deterministic motion tracker.

Additional sequences: Fig. 18 displays some tracking results we obtain for the tracking of people in meetings with the M3 tracker. Although these sequences are less dynamic, they illustrate the robustness of the method to heavy background clutter, partial occlusion, and the large variations in head appearance and pose that can occur in a natural setting.

VII. Conclusion

We presented a methodology to embed data-driven motion measurements into particle filters. This was first achieved by proposing a new graphical model that accounts for the temporal correlation existing between successive images of the same object. We show that this new model can be easily

[7] Head turns are difficult cases for the new method, as in the extreme case, the motion inside the head region indicates a right (or left) movement while the head outline remains static, as illustrated by the failure of the motion tracker in the second example, Fig.14.


(a) t_{1075} (b) t_{1200} (c) t_{1225} (d) t_{1350}

(e) t_{1450} (f) t_{1600} (g) t_{2100} (h) t_{2300}

Fig. 18. Head tracking in meetings with the M3 tracker, $N_s = 250$.

handled by the particle filter framework. The new introduced observation likelihood term can be exploited to model the visual motion using either implicit or explicit measurements. Secondly, explicit motion estimates were exploited to predict more precisely the new state values. This data-driven approach allows for designing better proposals that take into account the new image. Altogether, the algorithm allows to better handle unexpected and fast motion changes, to remove tracking ambiguities that arise when using generic shape-based or color-based object models, and to reduce the sensitivity to the different parameters of the prior model.

The conducted experiments have demonstrated the benefit of exploiting the proposed scheme. However, this should not obliterate the fact that the tracking performance depends on the choice of a good and robust object model. This was also illustrated in the reported experiments. The color tracker, when its use is appropriate, performs better than its shape equivalent. However, the reference histogram model in this case was extracted by hand from the first frame of the sequence. In practice, the tracking performance may depend on how well this reference histogram has been learned, and the automatic initialization and online adaptation of this model need to be addressed, e.g. using similar schemes as in [6], [21]. In addition, the development of a probability density model that jointly accounts for temporal color consistency and object modeling may improve the results. This idea might be worth exploring in the future. More generally, thus, when dealing on a specific object tracker, like head tracker for instance, building more precise or adaptive object likelihood may further improve the proposed method. This can be achieved by developing better probability density functions to model the likelihood of observations of different nature, or measured at different spatial or temporal positions, as well as simultaneously modeling in a principle way the temporal correlation between these observations.

Finally, we have showed that the exploitation of explicit motion measurements in the proposal improved the tracking efficiency. The described approach is general. For instance, it can be used to track deformable objects, by exploiting the integration of motion measurements along the shape curve, as described in [38]. However, in this case, the usefulness and the robustness of the low-level motion measurements to model the temporal variation of fine scale parameters need to be demonstrated. The use of mixture of proposals [32] relying on different cues (prior, visual motion, color), or of an hybrid scheme, in which one part of the state parameters (e.g. translation, scale, rotation,...) are sampled from a data driven motion proposal, while the other part is drawn from a standard AR model, might be more appropriate in these situations.

Acknowledgments

The authors would like to thank both the Swiss National Fund for Scientific Research that supported this work through the Swiss National Center of Competence in Research (NCCR) on Interactive Multimodal Information Management (IM)2. This work was partly supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811, publication 130).
The authors also thank the Eastman Kodak Company for providing the second (wedding) sequence.

References

[1] J-M. Odobez and D. Gatica-Perez, “Embedding motion in model-based stochastic tracking,” in *17th Int. Conf. Pattern Recognition (ICPR)*, Cambridge, UK, Aug. 2004, pp. II:815–818.

[2] A. Blake and M. Isard, *Active Contours*, Springer-Verlag, 1998.

[3] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking,” *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[4] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, 2001.

[5] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2000, pp. II:142–149.

[6] Y. Wu and T. Huang, “A co-inference approach for robust visual tracking,” in *Proc.* 8^{th} *IEEE Int. Conf. Computer Vision*, Vancouver, July 2001, pp. II:26–33.

[7] Y. Raja, S. McKenna, and S. Gong, “Colour model selection and adaptation in dynamic scenes,” in *Proc. of* 5^{th} *European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 1406*, 1998, pp. 460–474.

[8] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. of 7th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 2350*, Copenhaguen, Denmark, June 2002, pp. 661–675.

[9] E. Arnaud, E. Mémin, and B. Cernushi Frias, "Filtrage conditionnel pour la trajectographie dans des séquences d'images - application au suivi de points," in *14ème Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle*, Toulouse, France, Jan. 2004.

[10] M. J. Black and A. D. Jepson, "A probabilistic framework for matching temporal trajectories: Condensation-based recognition of gestures and expressions," in *Proc. of 5th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 1406*, H. Burkhardt and B. Neumann, Eds., Freiburg, Germany, 1998, pp. 909–924, Springer-Verlag.

[11] M. Isard and A. Blake, "ICONDENSATION : Unifying low-level and high-level tracking in a stochastic framework," in *Proc. of 5th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 1406*, 1998, pp. 893–908.

[12] Y. Rui and Y. Chen, "Better proposal distribution: object tracking using unscented particle filter," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, Dec. 2001, pp. II:786–793.

[13] K. Toyama and A. Blake, "Probabilistic tracking in a metric space," in *Proc. 8th IEEE Int. Conf. Computer Vision*, Vancouver, July 2001, pp. II:50–57.

[14] A. Doucet, S. Godsill, and C. Andrieu, "On sequential monte carlo sampling methods for bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[15] J. Sullivan and Rittscher J., "Guiding random particles by deterministic search," in *Proc. 8th IEEE Int. Conf. Computer Vision*, Vancouver, July 2001, pp. I:323–330.

[16] H. Tao, H.S. Sawhney, and R. Kumar, "Object tracking with bayesian estimation of dynamic layer representations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 1, pp. 75–89, 2001.

[17] A.D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust on-line appearance models for visual tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 10, pp. 661–673, Oct. 2003.

[18] H. T. Nguyen, M. Worring, and R. van den Boomgaard, "Occlusion robust adaptive template tracking," in *Proc. 8th IEEE Int. Conf. Computer Vision*, Vancouver, July 2001, pp. I:678–683.

[19] M.J. Black and A.D. Jepson, "Eigentracking : robust matching and tracking of articulated objects using a view based representation," *Int. J. Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

[20] A. Rahimi, L.P. Morency, and T. Darrell, "Reducing drift in parametric motion tracking," in *Proc. 8th IEEE Int. Conf. Computer Vision*, Vancouver, July 2001, pp. I:315–322.

[21] J. Vermaak, P. Pérez, M. Gangnet, and A. Blake, "Towards improved observation models for visual tracking: Selective adaptation," in *Proc. of 7th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 2350*, Copenhague, Danemark, 2002, pp. 645–660.

[22] H. Sidenbladh and M.J. Black, "Learning image statistics for bayesian tracking," in *Proc. 8th IEEE Int. Conf. Computer Vision*, Vancouver, Canada, July 2001, pp. II:709–716.

[23] H. Sidenbladh, M.J. Black, and D.J. Fleet, "Stochastic tracking of 3d human figures using 2d image motion," in *Proc. of 6th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 1843*, Dublin, Ireland, June 2000, pp. II:702–718.

[24] P. Pérez, J. Vermaak, and A. Blake, "Data fusion for visual tracking with particles," *Proc. IEEE*, vol. 92, no. 3, pp. 495–513, 2004.

[25] D. Gatica-Perez, G. Lathoud, I. McCowan, and J.-M. Odobez, "A Mixed-State I-Particle Filter for Multi-Camera Speaker Tracking," in *IEEE Int. Conf. on Computer Vision Workshop on Multimedia Technologies for E-Learning and Collaboration (ICCV-WOMTEC)*, 2003.

[26] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.

[27] E. Arnaud and E. Mémin, "Optimal importance sampling for tracking in image sequences:application to point tracking," in *Proc. of 8th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 3023*, Prague, Czech Republic, May 2004, pp. III:302–314.

[28] U. Grenander, Y. Chow, and D.M. Keenan, *HANDS. A Pattern Theoretical Study of Biological Shapes*, Springer-Verlag, New-York, 1991.

[29] A.M. Tekalp, *Digital video processing*, Signal Processing series. Prentice Hall, 1995.

[30] J. Sullivan, A. Blake, M. Isard, and J. MacCormick, "Object localization by bayesian correlation," in *Proc. 7th IEEE Int. Conf. Computer Vision*, 1999, pp. II:1068–1075.

[31] J. Sullivan, A. Blake, and J. Rittscher, "Statistical foreground modeling for object localisation," in *Proc. of 6th European Conf. Computer Vision, Lecture Notes in Compter Science, vol. 1843*, 2000, pp. II:307–323.

[32] Y. Chen and Y. Rui, "Real-time Speaker Tracking using Particle Filter Sensor Fusion," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 485–494, Mar. 2004.

[33] J.-M. Odobez and P. Bouthemy, "Robust multiresolution estimation of parametric motion models," *Journal of Visual Communication and Image Representation*, vol. 6, no. 4, pp. 348–365, Dec. 1995.

[34] P.J. Hubert, *Robust statistics*, Wiley, 1981.

[35] B.K.P. Horn and B.G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, pp. 185–203, 1981.

[36] J. Vermaak, A. Doucet, and P. Pérez, "Maintaining multi-modality through mixture tracking," in *Proc. 9th IEEE Int. Conf. Computer Vision*, Nice, France, June 2003, pp. II:1110–1116.

[37] K. Smith, D. Gatica-Perez, J.M. Odobez, and S. Ba, "Evaluating multi-object tracking," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Workshop on Empirical Evaluation Methods in Computer Vision (CVPR-EEMCV)*, San Diego, June 2005.

[38] C. Kervrann, F. Heitz, and P. Pérez, "Statistical model-based estimation and tracking of non-rigid motion," in *Proc. 13th Int. Conf. Pattern Recognition*, Vienna, Austria, August 1996, pp. D:244–248.

Jean-Marc Odobez (M'03) was born in France in 1968. He graduated from the Ecole Nationale Supérieure de Télécommunications de Bretagne (ENSTBr) in 1990, and received his Ph.D degree in Signal Processing and Télécommunication from Rennes University, France in 1994. He performed his dissertation research at IRISA/INRIA Rennes on dynamic scene analysis (image stabilization, object detection and tracking, motion segmentation) using statistical models (robust estimators, 2D statistical labeling with Markov Random Field). He then spent one year as a post-doctoral fellow at the GRASP laboratory, University of Pennsylvania, USA, working on visually guided navigation problems. From 1996 until september 2001, he was associate professor at the Université du Maine, France. In 2001, he joined the IDIAP Research Insitute as a senior researcher, where he is working mainly on the development of statistical methods and machine learning algorithms for multimedia signal analysis and computer vision problems.

Daniel Gatica-Perez (S'01, M'02) received the B.S. degree in Electronic Engineering from the University of Puebla, Mexico in 1993, the M.S. degree in Electrical Engineering from the National University of Mexico in 1996, and the Ph.D. degree in Electrical Engineering from the University of Washington, Seattle, in 2001. He joined the IDIAP Research Institute in January 2002, where he is now a senior researcher. His interests include multimedia signal processing and information retrieval, computer vision, and machine learning applied to these domains.

Sileye O. Ba obtained a master in applied mathematics oriented signal processing of Dakar University in 2000. In 2002 he completed a master in mathematics, computer vision and machine learning of E.N.S. Cachan in Paris. Since October 2002 he is a phd student at the IDIAP Research Institute working on object tracking and event recognition in video sequences using sequential Monte Carlo methods.