Dynamic Partitioned Sampling For Tracking With Discriminative Features

Stefan Duffner[1]
stefan.duffner@idiap.ch
Jean-Marc Odobez[1]
jean-marc.odobez@idiap.ch
Elisa Ricci[2]
eliricci@fbk.eu

[1] Idiap Research Institute
Martigny, CH

[2] Bruno Kessler Foundation - FBK-irst
Technologies of Vision Research Lab
Povo, Italy

Abstract

We present a multi-cue fusion method for tracking with particle filters which relies on a novel hierarchical sampling strategy. Similarly to previous works, it tackles the problem of tracking in a relatively high-dimensional state space by dividing such a space into partitions, each one corresponding to a single cue, and sampling from them in a hierarchical manner. However, unlike other approaches, the order of partitions is not fixed a priori but changes dynamically depending on the reliability of each cue, *i.e.* more reliable cues are sampled first. We call this approach Dynamic Partitioned Sampling (DPS). The reliability of each cue is measured in terms of its ability to discriminate the object with respect to the background, where the background is not described by a fixed model or by random patches but is represented by a set of informative "background particles" which are tracked in order to be as similar as possible to the object. The effectiveness of this general framework is demonstrated on the specific problem of head tracking with three different cues: colour, edge and contours. Experimental results prove the robustness of our algorithm in several challenging video sequences.

1 Introduction

Developing a visual tracking system that is robust and adaptive to changing conditions is one of the main challenges in computer vision. In the past, several strategies have been proposed which try to improve tracking robustness by fusing multiple complementary cues (*e.g.* colour, edge, contours). In case of tracking in a Bayesian framework, the simplest method to integrate multiple cues consists in considering them independently and multiplying the corresponding likelihood functions (see *e.g.* [3, 4]). However, this approach does not take into account the reliability of each cue. In theory, a tracking algorithm should lower the contribution of uncertain features and use only the cues which are considered reliable. This idea has been previously exploited in [13, 17], where a measure of cue confidence is estimated at each frame in order to allow the system to adapt to varying external conditions. Other approaches consider separate particle filters for each cue and model inter-filter dependencies explicitly with a graphical model [8, 19]. Another series of works (which are in the spirit more similar to ours) attempt to define an order of cue relevance and generate particles of "downstream"

BMVC 2009 doi:10.5244/C.23.71

cues in the region individuated as likely to contain the target by the "upstream" cues [14, 20]. Obviously, the main drawback of these approaches is that fixing the order of importance of cues in advance is highly suboptimal. It would be better if the order is changed dynamically depending on the reliability of each cue.

The estimation of cue reliability (or uncertainty) and, in general, the conception of a measure that quantifies tracking performance is a very challenging task. Several reliability measures have been proposed in the past. For example, in [2, 13], the uncertainty of each single-cue tracker is measured in terms of the spatial spread of the associated particles. Alternatively, in [17] a score is assigned to each feature which quantifies the difference between the tracking result obtained by the cue alone and the result obtained using all the cues. Another possibility consists in measuring a feature's ability to discriminate between the foreground (FG) object and background (BG). For example in [5, 21] an online feature selection method for tracking based on a discriminative measure called variance ratio is proposed.

In this paper, a new algorithm to combine several cues in a particle filter framework is proposed. In contrast to previous works, we introduce an adaptive approach where the cue reliability is dynamically estimates and used to drive the particle generation process. Most reliable cues are used to roughly estimate the region containing the target and to restrict the "search space" that is explored with the less confident features. We call this approach Dynamic Partitioned Sampling (DPS). Moreover, a novel measure of reliability is proposed which quantifies the cue's ability to discriminate the target w.r.t. the BG. The method is able to adapt to a non-static BG, as BG is represented by samples that are updated at each frame using a simple filtering approach. This approach is demonstrated in the context of head tracking with three cues (colour, edge and contours): a challenging problem where, as demonstrated by previous works [11], cues integration is crucial.

The paper is organised as follows. In Section 2 particle filter and partitioned sampling are introduced. Section 3 describes the proposed tracking approach while in Section 4 we present our notion of cue confidence. Experimental results are reported in Section 5. Finally, in Section 6, conclusions are drawn.

2 Particle filters and partitioned sampling

Let us denote by $\mathbf{x}_t$ the hidden state which represents the object configuration and by $\mathbf{y}_t$ the associated observation extracted from the image at time t. In a Bayesian framework, tracking can be formulated as the problem of estimating the posterior probability density function (pdf) $p(\mathbf{x}_t|\mathbf{y}_{0:t})$ for the position $\mathbf{x}_t$ given a sequence of image observations $\mathbf{y}_{0:t}$.

Particle filters and importance sampling. In particle filtering, the pdf if approximated by a set of N weighted samples (the particles) $\{\mathbf{x}_t^i, \omega_t^i\}$ where ω_t^i represent the normalised weights corresponding to the estimates of the state $\mathbf{x}_t^i$. The pdf estimation is realised recursively by the following three steps: prediction, update and resampling. During the prediction step each particle $\mathbf{x}_t^i$ is sampled from an appropriate proposal distribution $q(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{0:t})$. In the simplest case, the proposal corresponds to the dynamical model $(q(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{0:t}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}))$. In the update stage, the particle weights are recomputed according to the observation model $p(\mathbf{y}_t|\mathbf{x}_t^i)$ as $\omega_t^i = \omega_{t-1}^i p(\mathbf{y}_t|\mathbf{x}_t^i) F(\mathbf{x}_t^i, \mathbf{x}_{0:t-1}^i, \mathbf{y}_{0:t})$ where $F(\mathbf{x}_t^i, \mathbf{x}_{0:t-1}^i, \mathbf{y}_{0:t}) = p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)/q(\mathbf{x}_t^i|\mathbf{x}_{0:t-1}^i, \mathbf{y}_{0:t})$. Finally, resampling provides the elimination of particles that have small weights and the replication of those with higher weights.

Partitioned sampling. A drawback of importance sampling is that the number of particles needed to approximate the pdf grows exponentially with the dimensionality of the state

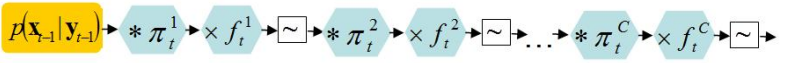

Figure 1: Partitioned Sampling ($*$ denotes the convolution with the dynamical model $\pi_t^i = p(\mathbf{y}_{i,t}|\mathbf{x}_{i,t})$, $\times$ the multiplication by the likelihood $f_t^i = p(\mathbf{y}_{i,t}|\mathbf{x}_{i,t})$, $\sim$ standard resampling)

space. To avoid such problem, many approaches have been proposed in the past such as annealed particle filtering [7], Markov Chain Monte Carlo method [9] and Partitioned Sampling (PS) [12]. In particular PS (Fig. 1) is an approach based on a hierarchical decomposition of the state space, which consists of a sequence of simple particle filters, so that each filter estimates parts of the state space independently.

As the processing order of the sub-spaces is fixed, a common problem with PS is the so-called *impoverishment* effect: due to the iterative resampling, samples that have a low weight at the early stages of the processing disappear although they might be important to model the distribution at later stages. Thus, the order of sampling influences the final result. Smith and Gatica-Perez [18] alleviated this problem by an approach called Distributed Partitioned Sampling, where a mixture distribution is used to combine the results of several PS solutions, each of them using a different sampling order. This approach is well suited if no information is given a priori on which sampling order is the best. However, in this paper, we use a separate measure to determine at each time which partitions of the state space are more reliable, and sample first the reliable partitions and then the others. In this way, the dynamic nature of our algorithm helps to reduce the impoverishment effect since a more reliable partitions at time t will probably be a less reliable one later on.

3 Dynamic Partitioned Sampling

We define the state space $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_C)$ where C is the number of cues. Each state model $\mathbf{x}_i$ is described by its own state variables, and its likelihood $p(\mathbf{y}_{i,t}|\mathbf{x}_{i,t})$ is defined independently from the other models. It is clear that $\mathbf{x}_1..\mathbf{x}_C$ are not independent since they describe the same target but, as opposed to a unique state vector for all the cues, this representation allows for more flexibility and a more precise likelihood model. Note that in contrast to the original PS algorithm that partitions the state space in every dimension we choose to divide $\mathbf{x}$ into only C partitions corresponding to the sub-states of the different cues: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_C$. To describe the interactions between the sub-states of different cues, we introduce a pairwise Markov Random Field (MRF) prior in the dynamical model [10], *i.e.* we define $p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \prod_{c\in\mathscr{C}} p(\mathbf{x}_{c,t}|\mathbf{x}_{c,t-1}) \prod_{c_i,c_j\in\mathscr{C}} \phi(\mathbf{x}_{c_i,t}, \mathbf{x}_{c_j,t})$, where $p(\mathbf{x}_{c,t}|\mathbf{x}_{c,t-1})$ denotes the dynamics of the c-th cue, $\phi(\mathbf{x}_{c_i,t}, \mathbf{x}_{c_j,t})$ is the pairwise interaction potential between the pair of cues (c_i, c_j), and $\mathscr{C}$ denotes the set of cues.

Similarly to classical PS, we define a hierarchy between cues and use the upstream models to restrict the search space of the downstream ones. However, in DPS the order of cues is not fixed but is determined at time t by their reliability $R_{c,t-1}$ (which we discuss in the next section) estimated at time $t-1$, *i.e.* the most reliable cues are processed first. The idea is that in this way the inference is more effective because samples are first drawn from more reliable sub-spaces and "guide" the sampling in the remaining, less reliable, dimensions. To this aim we order cues according to their reliability and consider a proposal distribution which depends on this order, *i.e.* $q(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{0:t}) = \prod_{c\in\mathscr{C}} q_c(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{0:t}, R_{c,t-1})$. In DPS for the most reliable cue r_1, the samples are simply drawn from its dynamical model ($q_{r_1}(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{0:t}, R_{r_1,t-1}) = p(\mathbf{x}_{r_1,t}|\mathbf{x}_{r_1,t-1})$) and weighted according to the observation like-

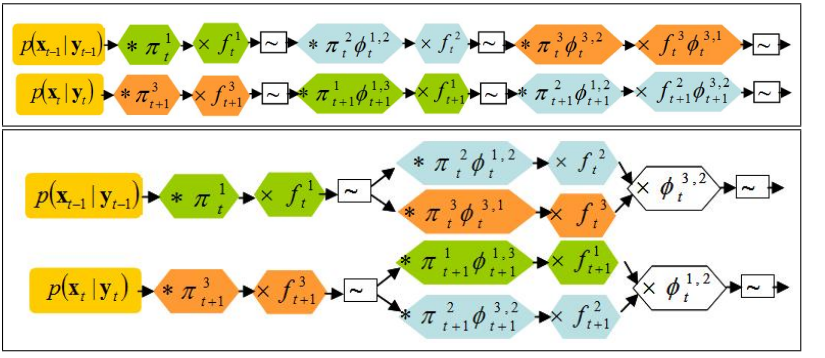

Figure 2: DPS-*chain*(top) and DPS-*tree*(bottom). Different colours represent different partitions. In the diagram $\phi_t^{c_i,c_j}$ denotes $\phi(\mathbf{x}_{c_i,t},\mathbf{x}_{c_j,t})$.

lihood $p(\mathbf{y}_{r_1,t}|\mathbf{x}_{r_1,t})$ as with a standard particle filter. On the other hand, for the subsequent cues r_i ($\{r_i : R_{r_i-1,t-1} < R_{r_i-1,t-1}, r_i \in \mathscr{C}\}$), the samples are drawn from the proposal functions $q_{r_i}(\mathbf{x}_t|\mathbf{x}_{0:t-1},\mathbf{y}_{0:t},R_{r_i,t-1}) = p(\mathbf{x}_{r_i,t}|\mathbf{x}_{r_i,t-1})\phi(\mathbf{x}_{r_i,t},\mathbf{x}_{r_i-1,t})$, *i.e.* the product of the dynamics with the potential function w.r.t. the previous cue. As we define both $p(\mathbf{x}_{c,t}|\mathbf{x}_{c,t-1})$ and $\phi(\mathbf{x}_{c_i,t},\mathbf{x}_{c_j,t})$ as Gaussian distributions, we can easily sample from their product. For downstream cues, the weighing is performed by their likelihoods $p(\mathbf{y}_{r_i,t}|\mathbf{x}_{r_i,t})$ and, in case of the last cue r_C, also by the evaluation of the potential $\phi(\mathbf{x}_{r_1,t},\mathbf{x}_{r_C,t})$ modelling the interactions between the states of the first and the last cues. Fig. 2 illustrates DPS in the case of 3 cues.

The *dynamic chain* structure defined above assumes that all the cues (except the last one) are reliable to some minimum extent. However, there could be situations where we cannot rely on more than a single cue, and restricting the search space according to what is "suggested" by an uncertain cue could be inappropriate. In this case, an alternative is to adopt a *tree* representation of the hierarchy, where we consider the most reliable cue as the *leading* cue L and we assume the others to be at the same level. We call this approach DPS-*tree*, as opposed to the DPS-*chain* discussed above. While for the leading cue DPS-*tree* and DPS-*chain* are equivalent, in DPS-*tree* (Fig. 2) for the downstream cues the proposal is defined as $q_{r_i}(\mathbf{x}_t|\mathbf{x}_{0:t-1},\mathbf{y}_{0:t},R_{r_i,t-1}) = p(\mathbf{x}_{r_i,t}|\mathbf{x}_{r_i,t-1})\phi(\mathbf{x}_{r_i,t},\mathbf{x}_{L,t})$. The weighing is performed by the likelihoods $p(\mathbf{y}_{r_i,t}|\mathbf{x}_{r_i,t})$ and an additional term $\hat{\phi}(\mathbf{x}_t)$, where constraints $\phi(\mathbf{x}_{r_i,t},\mathbf{x}_{r_j,t})$ that do not involve the leading cue are evaluated, *i.e.* $\hat{\phi}(\mathbf{x}_t) = \prod_{r_i,r_j\neq L}\phi(\mathbf{x}_{r_i,t},\mathbf{x}_{r_j,t})$.

3.1 Implementation details

In this work, we consider three cues (colour, edge and contours). However, it is worth noting that the proposed framework can be extended to more cues. In the following, we briefly describe the specific features of our particle filter.

3.1.1 State space and dynamical model

State space. The state vector $\mathbf{x} = (\mathbf{x}_{col},\mathbf{x}_{edge},\mathbf{x}_{con})$ jointly describes the position and head pose of the target. For colours, the state $\mathbf{x}_{col} = (t_x,t_y,s_x,e_y)$ indicates head location and size. The same type of state vector is used for the contour cue which is described by an ellipse model. For the edge, we define $\mathbf{x}_{edge} = (t_x,t_y,s_x,e_y,\theta)$, *i.e.* we include a discrete variable θ that estimates the head pose of the target. Previous works [1] have shown that explicitly modelling the head pose improves tracking robustness.

Dynamical model. Regarding $p(\mathbf{x}_{c,t}|\mathbf{x}_{c,t-1})$, standard first order auto-regressive models are used to represent the dynamics of the translation components (t_x, t_y) of the states, the scale s_x and the excentricity e_y. For the discrete variable θ, $p(\theta_t|\theta_{t-1})$ is a transition table learned from training data and based on the distance between adjacent poses.

As discussed above, a MRF models the interactions among the states of the three cues. We define $\phi(\mathbf{x}_{c_i,t}, \mathbf{x}_{c_j,t}) = \mathcal{N}(\frac{\mathbf{x}_{c_i,t} - \mathbf{x}_{c_j,t}}{s_{c_i,t}}; \mu^{\theta}_{c_i,c_j}; \Sigma^{\theta}_{c_i,c_j})$, where $\mathcal{N}$ is the Gaussian kernel function, $\mu^{\theta}_{c_i,c_j}$ and $\Sigma^{\theta}_{c_i,c_j}$ are respectively the mean vector and the covariance matrix, and $s_{c_i,t}$ is the current scale factor of cue c_i. Since our algorithm estimates also the head pose of the target we consider different constraints (and therefore different parameters $\mu^{\theta}_{c_i,c_j}$ and $\Sigma^{\theta}_{c_i,c_j}$) for different poses. Gaussian parameters are estimated off-line from an opportune training set of video sequences. To this end, we run three single-cue trackers separately on various training sequences (for about 13000 frames in total) and then, for each pose, compute a multivariate Gaussian model for the ϕ between the respective states.

Filter output. The estimate of the variables of interest (in this case the mean of the distribution) is obtained for each cue by averaging over all particles. Furthermore, pose estimation is realised only considering as particle weights the values of the edge likelihoods.

3.1.2 Observation model

An observation $\mathbf{y}_t$ is composed by colour, edge and contour features. For feature c, the likelihood function which quantifies the consistency of the reference model $\mathbf{r}_c$ with the current observation $\mathbf{y}_{c,t}$ is defined as:

$$p(\mathbf{y}_{c,t}|\mathbf{x}_{c,t}) = e^{-\lambda_c D_c(\mathbf{y}_{c,t}, \mathbf{r}_c)} \tag{1}$$

where D_c is an appropriate distance function depending on the selected cue.

Colour. Colours are described by histograms in the HSV space. We denote the colour histogram representing the target by $\mathbf{y}_{col,t}$ and the histogram of the template by $\mathbf{r}_{col}$ which corresponds to the histogram of the object region initialised at the first frame. The discrepancy between the candidate object and the template is based on the Bhattacharyya distance $D_{col}(\mathbf{y}_{col,t}, \mathbf{r}_{col}) = 1 - \sum_u \sqrt{\mathbf{y}_{col,t}(u)\mathbf{r}_{col}(u)}$.

Edge. Texture features are based on histograms of oriented gradients (HOG)[6]: descriptors which are at the same time robust under varying illumination and fast to compute. In particular we design opportune multi-level HOGs in order to discriminate between different head orientations: we partition the image into 2×2 (first level) and 4×4 (second level) non overlapping blocks of 2×2 cells and compute and HOG on each cell. The final HOG descriptor is obtained by the concatenation of these histograms. The suitability of multi-level HOG features for head pose classification is demonstrated in [16] and is not further discussed here due to lack of space. It is worth noting that both edge and colour histograms are computed efficiently with integral histograms [15].

Similarly to [16] multiple pose-specific reference models $\mathbf{r}^{\theta}_{edge}$ are learned off-line using a publicly available database of faces with annotated pose. The texture likelihood function $p(\mathbf{y}_{edge,t}|\mathbf{x}_{edge,t})$ is defined as in Eqn. 1, where $D_{edge}(\mathbf{y}_{edge,t}, \mathbf{r}^{\theta}_{edge}) = \sum_p ||\mathbf{y}^p_{edge,t} - \mathbf{r}^{p,\theta}_{edge}||^2$, *i.e.* the sum of the Euclidean distances between corresponding HOGs of elementary cells associated to the template and to the current observation.

Contour. Contour observation is based on edge measurements. Edge-based measurements are performed in 1D along L normal line segments to a hypothesised contour. The

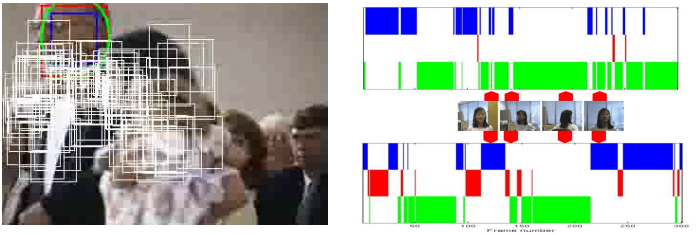

Figure 3: (a) A snapshot showing the BG samples (white rectangles) for the colour cue. (b) Leading cue for the sequence *girl* as computed from particle spread (top) and from FG/BG ratio (bottom).

latter is defined as an ellipse and constitutes the reference model $\mathbf{r}_{con}$. On each normal line segment l, we take the nearest edge position $y^l_{con,t}$ and calculate the Euclidean distance to the respective point on the ellipse r^l_{con}. In other words, the likelihood function $p(\mathbf{y}_{con,t}|\mathbf{x}_{con,t})$ is defined as in Eqn. 1 where $D_{con}(\mathbf{y}_{con,t},\mathbf{r}_{con}) = \sum_{l=1}^{L} ||y^l_{con,t} - r^l_{con}||^2$.

4 Cue reliability

To define reliability, we follow the same ideas in [5, 21]: the features which are more promising for tracking are those which allow better discrimination between the object and the BG. However differently from [5] our analysis is not limited to colour features, and opposite to [21] the proposed reliability measure is embedded into a particle filter framework. Typically, in particle filtering the BG is described by a fixed model or by a set of random patches. However, while a fixed model might be inappropriate since the BG can change over time, also extracting at each frame a set of random BG samples in the neighbourhood of the target location can be sub-optimal since only few of them (*i.e.* those close in appearance to the FG) are important to measure the discriminative ability of a cue. Therefore, in order to represent more effectively the BG information, we decide to use only *informative* BG samples, *i.e.* samples which are as similar as possible to the target.

More specifically, together with the object tracking algorithm described above, we also define C simple *auxiliary* BG trackers. For cue c, a set of N_{BG} weighted BG particles $\{\mathbf{x}^{b,BG}_{c,t}, \omega^{b,BG}_{c,t}\}$ are initialised at random positions in the image region surrounding the tracked object. Then, the BG samples are updated dynamically using a simple filtering approach. That means, at each frame the particles are weighted according to their likelihood $\omega^{b,BG}_{c,t} = \omega^{b,BG}_{c,t-1} p(\mathbf{y}^{b,BG}_{c,t}|\mathbf{x}^{b,BG}_{c,t})$ where the likelihood is defined as in Eqn. 1. However, in this case the reference model of the BG is the observation corresponding to the mean of the pdf $\bar{\mathbf{x}}_{c,t}$ estimated by the DPS tracker at current frame, *i.e.* $\mathbf{r}^{BG}_{c,t} = \mathbf{y}_{c,t}(\bar{\mathbf{x}}_{c,t})$. In this way, the changes in appearance of the target are taken into account by the BG tracker. Then, the BG sam-

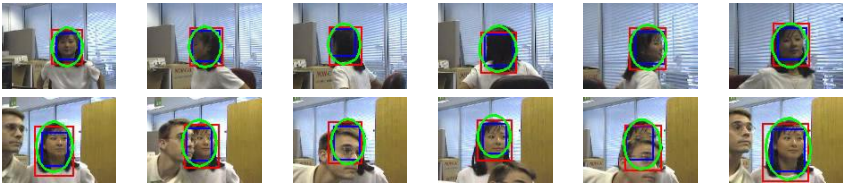

Figure 4: Results of head tracking with DPS-*tree* under change of pose and occlusion. The circles and the rectangles are the estimates of the colour (blue), texture (red) and contour (green) cues.

Method	M1L	M1R	M2L	M2R	M3L	M3R
DPS-*tree*	**5**	**6**	7	**5**	**7**	**5**
PS-*tree* (colour leading)	10	11	12	14	9	11
DPS-*chain*	7	7	**5**	6	9	**5**
PS ($col \rightarrow edge \rightarrow con$)	9	11	6	8	12	18

Table 1: Average centre location error (pixels) on annotated CLEAR sequences.

ples are updated according to their dynamical model $p(\mathbf{x}_{c,t}^{b,BG}|\mathbf{x}_{c,t-1}^{b,BG})p_0(\mathbf{x}_{c,t}^{b,BG},\bar{\mathbf{x}}_{c,t})$, where $p(\mathbf{x}_{c,t}^{b,BG}|\mathbf{x}_{c,t-1}^{b,BG})$ is a standard first order autoregressive model and $p_0(\mathbf{x}_{c,t}^{b,BG},\bar{\mathbf{x}}_{c,t})$ is a prior which allows to exclude from the search space the target region estimated by the FG particle filter. Finally, resampling is performed to eliminate uninformative BG samples. Fig. 3.a shows an image with BG samples for the colour cue. This clearly illustrates that the BG samples group around ambiguous image regions, *i.e.* regions with face-like colours.

This collaborative scheme of FG and BG trackers allows us to define the reliability:

$$R_{c,t} = \sum_{b=1}^{N_{BG}} w_b \delta \left[\log \frac{p(\mathbf{y}_{c,t}|\bar{\mathbf{x}}_{c,t})}{p(\mathbf{y}_{c,t}^{b,BG}|\mathbf{x}_{c,t}^{b,BG})} > T_c \right] \tag{2}$$

where δ is an indicator function and T_c is a user-defined threshold. In practice, if $T_c = 0$ and $w_b = 1$, $R_{c,t}$ counts how many times the observation corresponding to the current target estimate $\mathbf{y}_{c,t}(\bar{\mathbf{x}}_{c,t})$ is more similar to the FG template than to a BG sample. The weights w_b are defined as $w_b = exp(-d(\bar{\mathbf{x}}_{c,t},\mathbf{x}_{c,t}^{b,BG}))$, where d represents the Euclidean distance between the location of the mean particle of the FG particle filter and that of the BG particle $\mathbf{x}_{c,t}^{b,BG}$. In this way, we favour BG samples that are spatially close to the object: the idea is that the closer a BG sample is, the more it is able to distract the FG tracker.

5 Experimental results

Qualitative evaluation. We first analyse the performance of our algorithms in the publicly available sequence *girl* [3]: a standard benchmark for evaluating trackers with multiple features (see *e.g.* [13, 19]). Both proposed algorithms DPS-*tree* and DPS-*chain* successfully handle the change of pose of the target and are robust to the occlusion which arises at the end of the sequence. Fig. 4 shows the output of DPS-*tree* but the reader can refer to the supplementary material for other videos showing our tracking results.

Fig. 5 (top row) depicts the results of our implementation of a standard colour based tracker: as soon as the target changes its pose the tracking algorithm fails. This demonstrates that the use of several cues is justified in this particular sequence. As second baseline, we consider a particle filter that tracks all the cues in a joint state space where dependencies between cues are defined by a MRF and where the likelihood is the product of the single cue likelihoods (as the model in [9]). On the sequence *girl*, this approach fails many times (Fig. 5, 2nd row), probably due to the fact that the ability of a cue to discriminate with respect to the BG is ignored and the target is lost when unreliable cues dominate on reliable ones.

To show the benefit of the proposed dynamical scheme, we further compare DPS-*tree* with a particle filter also based on a hierarchical (tree) sampling scheme but with a fixed leading cue. We call this approach PS-*tree* for short. Fig. 5 (3rd row) shows the associated results in case of the contour cue leading: when the target changes pose the tracker loses it although it is able to recover within a few frames. This uncertain behaviour is observed

several times along this sequence. Similar results are also obtained if colour and texture are selected as leading cues. Finally, we compare our results with those of DPS-*tree* but considering a different measure of cue reliability: the inverse of the uncertainty introduced in [13] which is based on the particle spread. DPS with the proposed FG/BG reliability outperforms DPS with reliability based on particle-spread, as we can see comparing Fig. 4 with Fig. 5 (bottom row). We argue that although the measure in [13] works well for standard particle filters, PS alters significantly the particle spread and thus makes this measure inappropriate. This problem can be observed in Fig. 3.b, where the leading cue is plotted at each frame. In this case, using the measure in [13], the prediction of the leading cue (Fig. 3.b, top plot) is much more noisy than ours (Fig. 3.b, bottom plot). For example, our FG/BG reliability clearly identifies when the target turns: the colour cue becomes unreliable and the control of the tracking is taken mainly by the contour.

Quantitative evaluation. We first consider two annotated sequences: *two_people* (Fig. 7) and *dad* (Fig. 6). Both sequences are quite challenging for the change of pose of the target and the occurrence of occlusions but also for the presence of camera zooming-unzooming. We quantitatively evaluate the performance of our algorithms. We denote by GS the area of the ground truth bounding box and by TS the surface of the bounding box estimated by the tracker. At each frame we evaluate the F-measure $F = (2 * PR)/(P+R)$ where P is the precision $P = (TS \cap GS)/TS$ and R the recall $R = (TS \cap GS)/GS$. We also compute S, *i.e.* we count the number of frames where the target is successfully tracked normalised by the total number of frames. By successfully tracking we mean that the intersection between GS and TS is not empty. We compare our methods with several baselines: a joint MRF particle filter, standard PS with the order of cues fixed a priori (*e.g.* $colour \rightarrow contour \rightarrow edge$), PS-*tree* with fixed leading cue, DPS-*tree* with leading cues randomly chosen at each frame and DPS-*tree* with leading cue chosen with reliability in [13]. All experiments have been performed with $N = 500$ particles. The results of the PS with leading cue selected randomly are taken on the average of 50 iterations. The performance of all the algorithms are reported in Table 2. For all the methods the values correspond to the bounding box associated to the contour cue. As shown in the table our approaches clearly outperforms the other methods. For the sequence *dad* DPS-*tree* is the only method where tracking is achieved for the entire sequence, despite the severe occlusion occurring at the end.

Finally we evaluate the performance of the DPS trackers on the sequences of the CLEAR06 corpus (*http://isl.ira.uka.de/clear06/*). It consists of three sequences with 6 people holding meetings, thus there is less movement. However, they illustrate the robustness of our approach in a natural setting with heavy background clutter, partial occlusion due to hand gesture, and large variations in head pose. In general, in these sequences, a colour based particle filter or a particle filter with a hierarchical sampling scheme with fixed cues order may suffice to achieve reasonable tracking accuracy. Nevertheless, DPS guarantees a more accurate and stable localisation. Fig. 8 illustrates the tracking results on a typical sequence: in this case the head is localised correctly despite the occlusion. Table 1 shows the results of DPS-*tree* and DPS-*chain* compared with the *best* results obtained with a sampling scheme based on a fixed order. It is clear that the use of a dynamic sampling scheme is crucial. Note that in this case the performance is measured in terms of distance between the centre location of the ground truth and the estimated bounding boxes. Comparing the results obtained with the proposed approaches, we found that DPS-*tree* performs slightly better than DPS-*chain* but we did not observe significant differences among them probably due to the fact that only three cues are examined. An extensive evaluation of the proposed algorithms in case of more than three cues will be subject of future works.

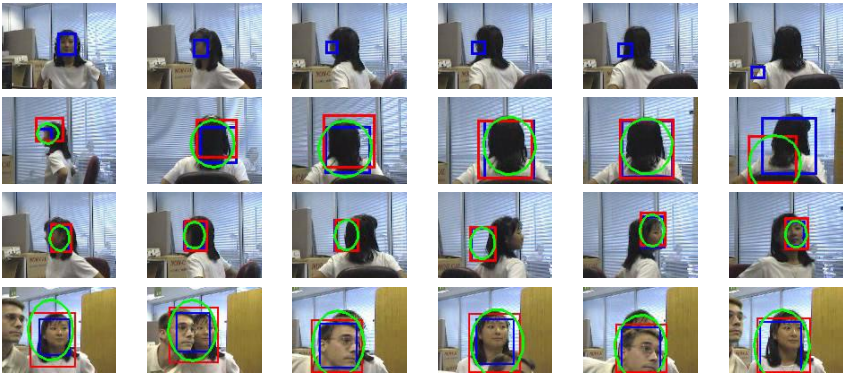

Figure 5: Results with colour based particle filter (first row), joint MRF particle filter (2nd row), PS-*tree* with a fixed leading cue (contour) (3rd row), DPS-*tree* with reliability in [13].

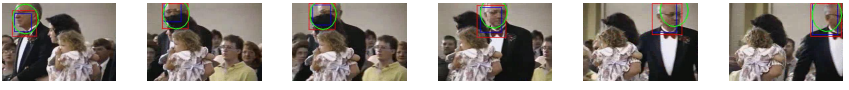

Figure 6: Results of head tracking with DPS-*tree* on the sequence *dad*.

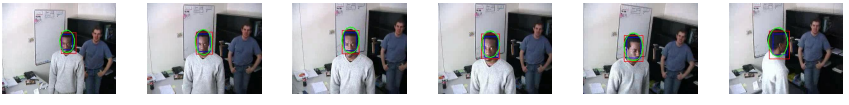

Figure 7: Results of head tracking with DPS-*chain* on the sequence *two_people*.

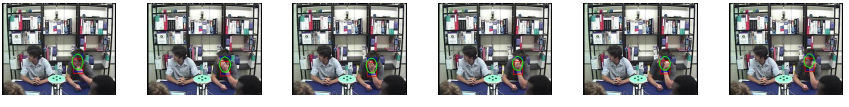

Figure 8: Results of head tracking with DPS-*chain* on the sequence *M1R* of *CLEAR* corpus.

6 Conclusions

We proposed a novel tracking algorithm based on adaptive hierarchical sampling scheme: the Dynamic Partitioned Sampling. This approach allows a clever fusion of multiple cues resulting in a system with increased robustness. The method is demonstrated in the context of head tracking: colour, edge and contours are used jointly to track the target in a way such that the estimation of the state space of the most reliable cues is used to drive the particle generation process for the others. The reliability is estimated dynamically with a novel measure which quantifies how well a cue discriminates the target from BG. The experimental results demonstrate the effectiveness of DPS for challenging sequences with pose changes, partial occlusions and moving camera.

Acknowledgement

This work was performed within the Integrated Project TA2, Together Anytime, Together Anywhere (website: http://www.ta2-project.eu). TA2 receives funding from the European Commission under the EU's Seventh Framework Programme, grant agreement number 214793. The authors gratefully acknowledge the European Commission's financial support and the productive collaboration with the other TA2 consortium partners.

	two_people	*dad*
DPS-*chain* (FG/BG)	**0.77** (1)	0.73 (0.98)
DPS-*tree* (FG/BG)	0.76 (1)	**0.81** (1)
PS $(col \rightarrow edge \rightarrow con)$	0.68 (1)	0.68 (0.92)
PS $(con \rightarrow col \rightarrow edge)$	0.67 (0.98)	0.52 (0.74)
PS $(edge \rightarrow col \rightarrow con)$	0.59 (0.89)	0.45 (0.74)
PS-*tree* (colour leading)	0.69 (1)	0.74 (0.95)
PS-*tree* (contour leading)	0.69 (1)	0.55 (0.76)
PS-*tree* (edge leading)	0.65 (0.95)	0.51 (0.76)
DPS-*tree* (random)	0.74 (1)	0.69 (0.93)
DPS-*tree* ([13])	0.72 (1)	0.71 (0.95)
joint MRF	0.47 (0.84)	0.65 (0.74)

Table 2: Average tracking results (F) on annotated sequences. S in parenthesis.

References

[1] S. O. Ba and J.-M. Odobez. A probabilistic framework for joint head tracking and pose estimation. In *Proceedings of the International Conference on Pattern Recognition*, volume 4, pages 264–267, 2004.

[2] V. Badrinarayanan, P. Perez, F. Le Clerc, and L. Oisel. Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues. In *Proceedings of the International Conference on Computer Vision*, pages 1–8, October 2007.

[3] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 232–237, Santa Barbara, CA, USA, 1998.

[4] P. Brasnett, L. Mihaylova, N. Canagarajah, and D. Bull. Particle filtering with multiple cues for object tracking in video sequences. In *Proceedings of the SPIE*, volume 5685, pages 430–441, 2005.

[5] R. T. Collins, Y. Liu, and M. Leordeanu. On-line selection of discriminative tracking features. *IEEE Trans. on Pattern Analysis And Machine Intelligence*, 27(10):1631–1643, 2005.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, San Diego, CA, USA, 2005.

[7] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 126–133, 2000.

[8] W. Du and J. H. Piater. A probabilistic approach to integrating multiple cues in visual tracking. In *Proceedings of the European Conference on Computer Vision*, pages 225–238, 2008.

[9] Z. Khan, T. Balch, and F. Dellaert. Mcmc-based particle filter for tracking multiple interacting targets. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(11): 1805–1918, November 2005.

[10] S. Z. Li. *Markov Random Field Modeling in Computer Vision*. Springer, London, UK, 1995.

[11] Y. Li, H.Z. Ai, C. Huang, and S.H. Lao. Robust head tracking with particles based on multiple cues fusion. In *Proceedings of the ECCV workshop on Human Computer Interaction*, volume 1, pages 29–39, 2006.

[12] J. MacCormick and M. Isard. Partitioned sampling, articulated objects, and interface-quality hand tracker. In *Proceedings of the European Conference on Computer Vision*, volume 2, pages 3–19, 2000.

[13] E. Maggio, F. Smeraldi, and A. Cavallaro. Adaptive multi-feature tracking in a particle filtering framework. *IEEE Trans. on Circuits and Systems for Video Technology*, 10(7): 1348–1359, October 2007.

[14] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE*, 92(3):495–513, 2004.

[15] F. Porikli. Integral histogram: A fast way to extract higtograms in cartesian spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 829–836, 2005.

[16] E. Ricci and J.M. Odobez. Learning large margin likelihoods for realtime head pose tracking. In *Proceedings of the IEEE International Conference of Image Processing*, 2009.

[17] C. Shen, A. van den Hengel, and A. Dick. Probabilistic multiple cue integration for particle filter based tracking. In *Proceedings of the 7th Digital Image Computing : Techniques and Applications*, pages 10–12, 2003.

[18] K. Smith and D. Gatica-Perez. Order matters: A distributed sampling method for multi-object tracking. In *Proceedings of the British Maschine Vision Conference*, London, September 2004.

[19] Y. Wu and T. S. Huang. Robust visual tracking by integrating multiple cues based on co-inference learning. *International Journal of Computer Vision*, 58(1):55–71, 2004.

[20] C. Yang, R. Duraiswami, and L. Davis. Fast multiple object tracking via a hierarchical particle filter. In *Proceedings of the International Conference on Computer Vision*, volume 1, pages 212–219, 2005.

[21] Z. Yin, F. Porikli, and R. Collins. Likelihood map fusion for visual object tracking. In *Proceedings of the IEEE Workshop on Applications of Computer Vision*, volume 1, pages 1–7, 2008.