

Data Fusion for Visual Tracking with Particles

Patrick Pérez, Jaco Vermaak, and Andrew Blake

INVITED PAPER

Abstract—The effectiveness of probabilistic tracking of objects in image sequences has been revolutionized by the development of particle filtering. Whereas Kalman filters are restricted to Gaussian distributions, particle filters can propagate more general distributions, albeit only approximately. This is of particular benefit in visual tracking because of the inherent ambiguity of the visual world that stems from its richness and complexity.

One important advantage of the particle filtering framework is that it allows the information from different measurement sources to be fused in a principled manner. Although this fact has been acknowledged before, it has not been fully exploited within a visual tracking context. Here we introduce generic importance sampling mechanisms for data fusion and discuss them for fusing color with either stereo sound, for tele-conferencing, or with motion, for surveillance with a still camera. We show how each of the three cues can be modeled by an appropriate data likelihood function, and how the intermittent cues (sound or motion) are best handled by generating proposal distributions from their likelihood functions. Finally, the effective fusion of the cues by particle filtering is demonstrated on real tele-conference and surveillance data.

Index Terms—Visual tracking, data fusion, particle filters, sound, color, motion

I. INTRODUCTION

VISUAL tracking entails the detection and recursive localization of objects, or more generally features, in video sequences. The tracking of objects has become an ubiquitous elementary task in both online and offline image based applications, including visual servoing (e.g., [34]), surveillance (e.g., [19]), gestural human-machine interfaces and smart environments (e.g., [38], [11], [33]), video compression (e.g., [56]), augmented reality and visual effects (e.g., [35]), motion capture (e.g., [36]), environmental imaging (e.g., [12], [37]), and many more.

Recently Sequential Monte Carlo Methods [15], [18], [20], [31], otherwise known as Particle Filters, have become popular tools to solve the tracking problem. Their popularity stems from their simplicity, flexibility, ease of implementation, and modeling success over a wide range of challenging applications. Within a visual tracking context these methods have been pioneered in the seminal paper by Isard and Blake [20], in which the term CONDENSATION was coined. This has subsequently led to a vast body of literature, which we shall not attempt to review here. See examples in [7], [13], [14], [26], [28], [41], [43], [45], [46], [48], [49].

One important advantage of the sequential Monte Carlo framework is that it allows the information from different measurement sources to be fused in a principled manner. Although this fact has been acknowledged before, it has not

been fully exploited within a visual tracking context, where a host of cues are available to increase the reliability of the tracking algorithm. Data fusion with particle filters has been mostly confined to skin color and edge cues inside and around simple silhouette shapes in the context of face and hand tracking [21], [50], [58], [59].

In this paper we present a particle filter based visual tracker that fuses three cues in a novel way: color, motion and sound (Fig. 1). More specifically, we will introduce color as the main visual cue and fuse it, depending on the scenario under consideration, with either sound localization cues or motion activity cues. The generic objective is to track a specified object or region of interest in the sequence of images captured by the camera. We employ weak object models so as not to be too restrictive about the types of objects the algorithm can track, and to achieve robustness to large variations in the object pose, illumination, motion, etc. In this generic context, contour cues are less appropriate than color cues to characterize the visual appearance of tracked entities. The use of edge based cues indeed requires that the class of objects to be tracked is known *a priori* and that rather precise silhouette models can be learned beforehand. Note however that such conditions are met in a number of tracking applications where shape cues are routinely used [2], [3], [25], [30], [40], [44], [53].

Color localization cues are obtained by associating some reference color model with the object of interest. This reference model is then compared, in some sense, to similar models extracted from candidate regions in the image, and the smaller the discrepancy between the candidate and reference models, the higher the probability that the object is located in the corresponding image region. The color reference model can be obtained from some automatic detection module, or by allowing the user to label the object of interest by hand. The model can then be defined in parametric form, using for instance mixtures of Gaussians (e.g., [23], [55], [59]). In this paper we use instead a histogram based color model inspired by the powerful deterministic color trackers by Bradski [4] and Comaniciu *et al.* [9], [10]. The likelihood is built on the histogram distance between the empirical color distribution in the hypothesized region and the reference color model [39].

Along the same lines we also introduce motion cues based on histogramming successive frame differences. Using a form similar to the color likelihood, the motion likelihood is designed to favor regions exhibiting a temporal activity larger than the average temporal activity in the scene. It will prove to be particularly effective in drawing the attention of the tracker back to objects moving in front of a still background in cases where lock has been temporarily lost.

For audio-visual tracking the system setup consists of a single camera and a stereo microphone pair. The line connecting the microphones goes through the optical center of

P. Pérez and A. Blake are with Microsoft Research, Cambridge.

J. Vermaak is with the Signal Processing Laboratory in the Cambridge University Engineering Department

Manuscript received March 21, 2003; revised November 19, 2003.

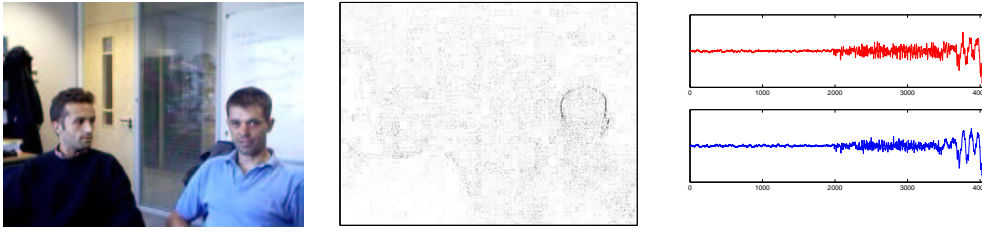


Fig. 1. **Three types of raw data.** We consider color based tracking combined with either motion cues (for surveillance with a static camera) or stereo sound cues (for speaker face tracking in a tele-conference setup). The corresponding measurements are respectively: (Left) RGB color video frames, (Middle) absolute luminance differences for pairs of successive frames, and (Right) stereo pairs of sound signal sections.

the camera, and is orthogonal to the camera optical axis. Sound localization cues are then obtained by measuring the Time Delay of Arrival (TDOA) between signals arriving at the two microphones comprising the pair. The TDOA gives an indication of the bearing of the sound source relative to the microphone pair. Given the configuration of the system this bearing can in turn be related to a horizontal position in the image (Fig. 2).

The color cues tend to be remarkably persistent and robust to changes in pose and illumination. They are, however, more prone to ambiguity, especially if the scene contains other objects characterized by a color distribution similar to that of the object of interest. The motion and sound cues, on the other hand, tend to be intermittent, but are very discriminant when they are present, *i.e.*, they allow the object to be located with low ambiguity.

The localization cues impact the particle filter based tracker in a number of ways. As is standard practice, we construct a likelihood model for each of the cues. These models are assumed to be mutually independent, an assumption that can be justified in the light that any correlation that may exist between the color, motion and sound of an object is likely to be weak. The intermittent and discriminant nature of the motion and sound cues make them excellent candidates for the construction of detection modules and efficient proposal distributions. We will exploit this characteristic extensively.

Finally, the differing nature of the cues and the configuration of the system allow us to experiment with the order and manner in which the cues are incorporated. For example, since the sound cue only gives localization information in the horizontal direction of the image, we can search this direction first, and confine the search in the remainder of the state-space to regions for which the horizontal image component have been deemed highly likely to contain the object of interest. This strategy is known as Partitioned Sampling [32], and allows for a more efficient exploration of the state-space.

The remainder of the paper is organized as follows. Section II briefly outlines the Bayesian Sequential Estimation framework, and shows how a Monte Carlo implementation thereof leads to the Particle Filter. It also presents some alternative particle filter architectures for cases where information from multiple measurement sources are available. Section III presents and discusses all the ingredients of our proposed data fusion tracker based on color, motion and sound. This section is concluded with a summary of the tracking algorithm. Section IV presents some tracking scenarios, and illustrates

the performance of the tracking algorithm under a variety of conditions. The usefulness of each localization cue, and their combined impact are evaluated. Finally, we conclude the paper in Section V with a summary and some suggestions for future research.

II. SEQUENTIAL MONTE CARLO AND DATA FUSION

Sequential Monte Carlo techniques for filtering time series [15], [18], [20], [31], and their use in the specific context of visual tracking [22], have been described at length in the literature. In what follows we give a brief summary of the framework, and discuss in some detail the architectural variations that are afforded by the presence of multiple measurement sources.

Denote by \mathbf{x}_n and \mathbf{y}_n the hidden state of the object of interest and the measurements at discrete time n , respectively. For tracking the distribution of interest is the posterior $p(\mathbf{x}_n|\mathbf{y}_{1:n})$, also known as the filtering distribution, where $\mathbf{y}_{1:n} = (\mathbf{y}_1 \cdots \mathbf{y}_n)$ denotes all the observations up to the current time step. In Bayesian Sequential Estimation the filtering distribution can be computed according to the two step recursion¹

prediction step:

$$p(\mathbf{x}_n|\mathbf{y}_{1:n-1}) = \int p(\mathbf{x}_n|\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}|\mathbf{y}_{1:n-1})d\mathbf{x}_{n-1} \quad (1)$$

filtering step:

$$p(\mathbf{x}_n|\mathbf{y}_{1:n}) \propto p(\mathbf{y}_n|\mathbf{x}_n)p(\mathbf{x}_n|\mathbf{y}_{1:n-1}), \quad (2)$$

where the prediction step follows from marginalisation, and the new filtering distribution is obtained through a direct application of Bayes' rule. The recursion requires the specification of a dynamic model describing the state evolution, $p(\mathbf{x}_n|\mathbf{x}_{n-1})$, and a model that gives the likelihood of any state in the light of the current observation, $p(\mathbf{y}_n|\mathbf{x}_n)$, along with the following conditional independence assumptions:

$$\mathbf{x}_n \perp \mathbf{y}_{1:n-1}|\mathbf{x}_{n-1} \text{ and } \mathbf{y}_n \perp \mathbf{y}_{1:n-1}|\mathbf{x}_n. \quad (3)$$

The recursion is initialized with some distribution for the initial state $p(\mathbf{x}_0)$. Once the sequence of filtering distributions is known point estimates of the state can be obtained according to any appropriate loss function, leading for example to the Maximum *a Posteriori* (MAP) estimate, $\arg \max_{\mathbf{x}_n} p(\mathbf{x}_n|\mathbf{y}_{1:n})$,

¹Notation " \propto " means that the conditional distribution on the left is proportional to the function on the right up to a multiplicative "constant" that may depend on the conditioning argument.

and to the Minimum Mean Square Error (MMSE) estimate, $\int \mathbf{x}_n p(\mathbf{x}_n | \mathbf{y}_{1:n}) d\mathbf{x}_n$.

The tracking recursion yields closed-form expressions in only a small number of cases. The most well-known of these is the Kalman filter [1] for linear Gaussian likelihood and state evolution models. The models encountered in visual tracking are often non-linear, non-Gaussian, multi-modal, or any combination of these. One reason for this stems from the fact that the observation model often specifies which part of the data is of interest given the state, leading to $p(\mathbf{y}_n | \mathbf{x}_n)$ being non-linear and often multi-modal with respect to the state \mathbf{x}_n . This renders the tracking recursion analytically intractable, and approximation techniques are required.

Sequential Monte Carlo methods [15], [18], [20], [31], otherwise known as Particle Filters, have gained a lot of popularity in recent years as a numerical approximation to the tracking recursion for complex models. This is due to their simplicity, flexibility, ease of implementation, and modeling success over a wide range of challenging applications.

The basic idea behind particle filters is very simple. Starting with a weighted set of samples $\{\mathbf{x}_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^{N_p}$ approximately distributed according to $p(\mathbf{x}_{n-1} | \mathbf{y}_{1:n-1})$, new samples are generated from a suitably chosen proposal distribution, which may depend on the old state and the new measurements, *i.e.*, $\mathbf{x}_n^{(i)} \sim q(\mathbf{x}_n | \mathbf{x}_{n-1}^{(i)}, \mathbf{y}_n)$, $i = 1 \dots N_p$. To maintain a consistent sample the new importance weights are set to²

$$w_n^{(i)} \propto w_{n-1}^{(i)} \frac{p(\mathbf{y}_n | \mathbf{x}_n^{(i)}) p(\mathbf{x}_n^{(i)} | \mathbf{x}_{n-1}^{(i)})}{q(\mathbf{x}_n^{(i)} | \mathbf{x}_{n-1}^{(i)}, \mathbf{y}_n)}, \text{ with } \sum_{i=1}^{N_p} w_n^{(i)} = 1. \quad (4)$$

The new particle set $\{\mathbf{x}_n^{(i)}, w_n^{(i)}\}_{i=1}^{N_p}$ is then approximately distributed according to $p(\mathbf{x}_n | \mathbf{y}_{1:n})$.

Approximations to the desired point estimates can then be obtained by Monte Carlo techniques. From time to time it is necessary to resample the particles to avoid degeneracy of the importance weights, that is the concentration of most of the weight on a single particle. In absence of resampling, this phenomenon always occurs in practice, dramatically degrading the sample based approximation of the filtering distribution, along with the quality of any point estimate based on it. The resampling procedure essentially multiplies particles with high importance weights, and discards those with low importance weights, while preserving the asymptotic properties of the sample based approximation of the filtering distribution. This procedure can be applied at each time step, or be invoked only when a measure of the “quality” of the weights falls below a threshold. A full discussion of degeneracy and resampling falls outside the scope of this paper, but more detail can be found in [15]. The synopsis of the generic particle filter iteration is given in Tab. I.

The performance of the particle filter hinges on the quality of the proposal distribution. The Bootstrap Filter [18],

² This can be seen by considering sample trajectories $\mathbf{x}_{1:n}^{(i)}$ [15]. These are distributed according to $q(\mathbf{x}_1 | \mathbf{y}_1) \prod_{k=2}^n q(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{y}_k)$ instead of true target distribution $p(\mathbf{y}_{1:n})^{-1} p(\mathbf{x}_1, \mathbf{y}_1) \prod_{k=2}^n p(\mathbf{y}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{x}_{k-1})$. According to importance sampling theory [16], the discrepancy is compensated for by associating importance weights proportional to the ratio of the target distribution to the proposal distribution, which yields in this case the recursion in (4).

With $\{\mathbf{x}_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^{N_p}$ the particle set at the previous time step, proceed as follows at time n :

Proposition: simulate $\mathbf{x}_n^{(i)} \sim q(\mathbf{x}_n | \mathbf{x}_{n-1}^{(i)}, \mathbf{y}_n)$.

Update weights:

$$w_n^{(i)} \propto w_{n-1}^{(i)} \frac{p(\mathbf{y}_n | \mathbf{x}_n^{(i)}) p(\mathbf{x}_n^{(i)} | \mathbf{x}_{n-1}^{(i)})}{q(\mathbf{x}_n^{(i)} | \mathbf{x}_{n-1}^{(i)}, \mathbf{y}_n)}, \text{ with } \sum_{i=1}^{N_p} w_n^{(i)} = 1.$$

If resampling:

simulate $a_i \sim \{w_n^{(k)}\}_{k=1}^{N_p}$, and replace $\{\mathbf{x}_n^{(i)}, w_n^{(i)}\} \leftarrow \{\mathbf{x}_n^{(a_i)}, \frac{1}{N_p}\}$.

TABLE I
GENERIC PARTICLE FILTER.

which is the first modern variant of the particle filter, uses the state evolution model $p(\mathbf{x}_n | \mathbf{x}_{n-1})$ as proposal distribution, so that the new importance weights in (4) become proportional to the corresponding particle likelihoods. This leads to a very simple algorithm, requiring only the ability to simulate from the state evolution model and to evaluate the likelihood. However, it performs poorly for narrow likelihood functions, especially in higher dimensional spaces. In [15] it is proved that the optimal choice for the proposal distribution (in terms of minimizing the variance of the importance weights) is the posterior $p(\mathbf{x}_n | \mathbf{x}_{n-1}, \mathbf{y}_n) \propto p(\mathbf{y}_n | \mathbf{x}_n) p(\mathbf{x}_n | \mathbf{x}_{n-1})$. However, the normalizing constant for this distribution, $\int p(\mathbf{y}_n | \mathbf{x}_n) p(\mathbf{x}_n | \mathbf{x}_{n-1}) d\mathbf{x}_n = p(\mathbf{y}_n | \mathbf{x}_{n-1})$, is rarely available in closed form, making direct sampling from this optimal proposal distribution impossible. The challenge in particle filtering applications is then to design efficient proposal distributions that approximate the optimal choice as closely as possible. We will give careful consideration to this issue in the design of our tracker in Section III.

For multiple measurement sources the general particle filtering framework can still be applied. However, it is possible to devise strategies to increase the efficiency of the particle filter by exploiting the relation between the structure of the model and the information in the various measurement modalities. In what follows we will suppress the time index n for notational compactness. Assume that we have M measurement sources, so that the instantaneous measurement vector can be written as $\mathbf{y} = (\mathbf{y}^1 \dots \mathbf{y}^M)$. We will further assume that the measurements are conditionally independent given the state, so that the likelihood can be factorized as

$$p(\mathbf{y} | \mathbf{x}) = \prod_{m=1}^M p(\mathbf{y}^m | \mathbf{x}). \quad (5)$$

With this setting the generic filter summarized in Tab. I could be used as is, with the weight update involving M likelihood evaluations according to (5). However, the factorized structure of the likelihood can be better exploited. To this end, we introduce the following abstract framework: let us assume that the state evolution and proposal distributions decompose as

$$p(\mathbf{x} | \mathbf{x}') = \int p_M(\mathbf{x} | \mathbf{x}^{M-1}) \dots p_1(\mathbf{x}^1 | \mathbf{x}') d\mathbf{x}^1 \dots d\mathbf{x}^{M-1} \quad (6)$$

$$q(\mathbf{x} | \mathbf{x}', \mathbf{y}) = \int q_M(\mathbf{x} | \mathbf{x}^{M-1}, \mathbf{y}^M) \dots q_1(\mathbf{x}^1 | \mathbf{x}', \mathbf{y}^1) d\mathbf{x}^1 \dots d\mathbf{x}^{M-1}, \quad (7)$$

With $\{\mathbf{x}_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^{N_p}$ the particle set at the previous time step, proceed as follows at time n :

Initialize: $\{\mathbf{x}_n^{(i)}, w_n^{(i)}\}_{i=1}^{N_p} = \{\mathbf{x}_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^{N_p}$

Layered sampling: for $m = 1 \dots M$

- *Proposition:* simulate $\mathbf{x}^{m(i)} \sim q_m(\mathbf{x}^m | \mathbf{x}^{m-1(i)}, \mathbf{y}_n^m)$.
- *Update weights:*

$$w^{m(i)} \propto w^{m-1(i)} \frac{p(\mathbf{y}_n^m | \mathbf{x}^{m(i)}) p_m(\mathbf{x}^{m(i)} | \mathbf{x}^{m-1(i)})}{q_m(\mathbf{x}^{m(i)} | \mathbf{x}^{m-1(i)}, \mathbf{y}_n^m)}, \text{ with } \sum_{i=1}^{N_p} w^{m(i)} = 1.$$
- *If resampling:* simulate $a_i \sim \{w^{m(k)}\}_{k=1}^{N_p}$, and replace $\{\mathbf{x}^{m(i)}, w^{m(i)}\} \leftarrow \{\mathbf{x}^{m(a_i)}, \frac{1}{N_p}\}$.

Terminate: $\{\mathbf{x}_n^{(i)}, w_n^{(i)}\}_{i=1}^{N_p} = \{\mathbf{x}^{M(i)}, w^{M(i)}\}_{i=1}^{N_p}$.

TABLE II

GENERIC LAYERED SAMPLING PARTICLE FILTER TO FUSE M OBSERVATION MODALITIES.

where $\mathbf{x}^1 \dots \mathbf{x}^{M-1}$ are ‘‘auxiliary’’ state vectors. Eq. (6) simply amounts to a splitting of the original evolution model into M successive intermediary steps. This can, for example, be done when the state is M -dimensional and corresponding component-wise evolution models are independent, and/or when the evolution model is linear and Gaussian, and can thus be easily fragmented into M successive steps with lower variances.

If we make the approximation that the likelihood for the m -th measurement modality $p(\mathbf{y}^m | \mathbf{x})$ can be incorporated after applying the m -th state evolution model $p_m(\mathbf{x}^m | \mathbf{x}^{m-1})$, we can set up a recursion to compute the new target distribution that takes the form

$$\pi^m(\mathbf{x}^m) \propto \int w_m(\mathbf{x}^m, \mathbf{x}^{m-1}) q_m(\mathbf{x}^m | \mathbf{x}^{m-1}, \mathbf{y}^m) \cdot \pi^{m-1}(\mathbf{x}^{m-1}) d\mathbf{x}^{m-1}, \quad m = 1 \dots M \quad (8)$$

$$\text{with } w_m(\mathbf{x}^m, \mathbf{x}^{m-1}) = \frac{p(\mathbf{y}^m | \mathbf{x}^m) p_m(\mathbf{x}^m | \mathbf{x}^{m-1})}{q_m(\mathbf{x}^m | \mathbf{x}^{m-1}, \mathbf{y}^m)}, \quad (9)$$

where π^0 and π^M are respectively the previous and the new filtering distributions, $\mathbf{x}^0 = \mathbf{x}'$, and $\mathbf{x}^M = \mathbf{x}$. This recursion can be approximated with a layered sampling strategy, where at the m -th stage new samples are simulated from a Monte Carlo approximation of the distribution $q_m(\mathbf{x}^m | \mathbf{x}^{m-1}, \mathbf{y}^m) \pi^{m-1}(\mathbf{x}^{m-1})$, with an associated importance weight proportional to $w_m(\mathbf{x}^m, \mathbf{x}^{m-1})$ to yield a properly weighted sample. The synopsis of this generic layered sampling strategy for data fusion is given in Tab. II.

As it stands the layered sampling approach provides no obvious advantage over the standard particle filtering framework. Its true benefit arises in cases where the measurement modalities differ in the level of information they provide about the state. If the measurement modalities are then ordered from coarse to fine, the layered sampling approach will effectively guide the search in the state-space, with each stage refining the result from the previous stage.

In some special applications the likelihood and state evolution models are independent over a component-wise partition-

ing of the state-space, *i.e.*,

$$p(\mathbf{y} | \mathbf{x}) = \prod_{m=1}^M p(\mathbf{y}^m | \mathbf{x}_m) \quad (10)$$

$$p(\mathbf{x} | \mathbf{x}') = \prod_{m=1}^M p(\mathbf{x}_m | \mathbf{x}_m'), \quad (11)$$

with $\mathbf{x} = (\mathbf{x}_1 \dots \mathbf{x}_M)$. For models of this nature the layered sampling procedure, with

$$p_m(\mathbf{x}^m | \mathbf{x}^{m-1}) = p(\mathbf{x}_m | \mathbf{x}_m^{m-1}) \prod_{k \neq m} \delta_{\mathbf{x}_k^{m-1}}(\mathbf{x}_k^m) \quad (12)$$

in (6) and (9), is exact, and known as Partitioned Sampling [32]. It effectively replaces the search in the full state-space by a succession of M easier search problems, each in a lower dimensional space. We will make use of these strategies when designing our tracking algorithm in Section III.

III. DATA FUSION VISUAL TRACKER

In this section we describe in detail all the ingredients of our tracking algorithm based on color, motion and sound. We first present the system configuration and the object model, and then proceed to discuss the localization cues and their impact on the tracking algorithm in more detail. The section is concluded with a summary of the tracking algorithm.

A. Audio-Visual System Setup

The setup of the tracking system is depicted in Fig. 2. It consists of a single camera and a stereo microphone pair. The line connecting the microphones goes through the optical center of the camera, and is orthogonal to the camera optical axis. Note, however, that in our experiments the object of interest will not always be a talking head in front of the camera. In such cases sound measurements will generally be absent, and we will only rely on the visual cues.

The system requires only a small number of calibration parameters: the microphone separation d , the camera focal length f , the width of the camera image plane in the real world W , and the width in pixels of the digital image W . These parameters are normally easy to obtain. Nevertheless, the tracking algorithm we develop is robust to reasonable inaccuracies in the system setup and variations in the calibration parameters. Since it is probabilistic in nature, these errors are easily accommodated by explicitly modeling the measurement uncertainty in the corresponding likelihood models.

Our objective is to track a specified object or region of interest in the sequence of images captured by the camera. To this end the raw measurements available are the images themselves, and the audio signals captured at the two microphones comprising the pair. Since the tracking is performed in the video sequence the discrete time index n corresponds to the video frame number. As opposed to the video sequence that is naturally discretized, the audio samples arrive continuously, and there is no notion of natural audio frames. For the purposes of the tracking algorithm, however, we define the n -th audio frame as a window of N_s audio samples centered around the sample corresponding to the n -th video frame. If T_v

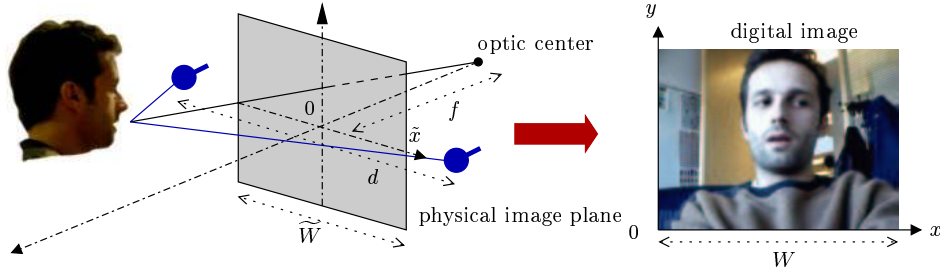


Fig. 2. **Setup for audio-visual tracking.** The system calibration parameters are the microphone separation d , the camera focal length f , the width of the image plane in the real world \tilde{W} , and the width in pixels of the digital image W .

and T_s denote the sampling period for video frames and audio samples, respectively, the center of the audio frame corresponding to the n -th video frame can be computed as

$$n_s = [(n-1)T_v/T_s + 1], \quad (13)$$

where $[\cdot]$ denotes the rounding operation. The number of samples in the audio frame N_s is normally taken such that the duration of the audio frame is roughly 50ms.

The raw image and audio frames are very high dimensional, and contain lots of information that is redundant with regard to object tracking. We thus pass the raw data through a signal processing front-end with the purpose of extracting features important for the tracking process. More specifically, we extract color and motion features from the image data, and Time Delay Of Arrival (TDOA) features from the audio data. With color being the most persistent feature of the object we will use it as the main visual cue and fuse it with the more intermittent motion and sound cues. In what follows we will denote the combined color, motion and sound measurements at time n by $\mathbf{y}_n = (\mathbf{y}_n^C, \mathbf{y}_n^M, \mathbf{y}_n^S)$. We will suppress the time index n in cases where there is no danger of ambiguities arising. The measurement procedures for each of the cues are described in detail in the relevant sections that follow.

B. Object Model

Our objective is to track a specified object or region of interest in the image sequence. We will aim to use models that make only weak assumptions about the precise object configuration, so as not to be too restrictive about the types of objects that can be tracked, and to achieve robustness to large variations in the object pose, illumination, motion, etc. In the approach adopted here the shape of the reference region, denoted by B^* , is fixed *a priori*. It can be an ellipse or rectangular box as in [4], [6], [9], but our modeling framework places no restrictions on the class of shapes that can be accommodated. More complex hand-drawn or learned shapes can be used if relevant.

Tracking then amounts to recursively estimating the parameters of the transformation to apply to B^* so that the implied region in each frame best matches the original reference region. Affinity or similitude transforms are popular choices. Since the color model we will describe in Section III-C is global with respect to the region of interest, we consider only translation and scaling of the reference region. This means that the reference region can be parameterized

as $B^* = (x^*, y^*, w^*, h^*)$, where (x^*, y^*) is the center of the reference region bounding box, and w^* and h^* are its width and height, respectively. We define the hidden state as $\mathbf{x}_n = (x_n, y_n, \alpha_n) \in \mathcal{X}$, with \mathcal{X} denoting the state-space, so that the corresponding candidate region becomes $B_{\mathbf{x}_n} = (x_n, y_n, \alpha_n w^*, \alpha_n h^*)$. The variables (x_n, y_n) thus form the center of the candidate region, and α_n acts as a scale factor.

Most objects move in a fairly predictable way. It is thus good practice in general to design the state evolution model $p(\mathbf{x}_n|\mathbf{x}_{n-1})$ to capture the salient features of the object motion. However, we desire our algorithm to be applicable to any object that may be of interest, including people, faces, motor vehicles, etc. Within such a large population of objects the variability in the characteristics of the object motion is likely to be high. Furthermore, we are interested in tracking in the image sequence, and not in the real world. The mapping of the motion from the three dimensional world to the two dimensional image representation is dependent on the system configuration and the direction of the motion, and is unknown in practice. We acknowledge these uncertainties by adopting a very weak model for the state evolution. More specifically, we assume that state components evolve according to mutually independent Gaussian Random Walk models. We augment these models with a small uniform component to capture the (rare) event where erratic motion in the real world is perceived as jumps in the image sequence. It also aids the algorithm in recovery of lock after a period of partial or complete occlusion. Thus the complete state evolution model can be written as

$$p(\mathbf{x}_n|\mathbf{x}_{n-1}) = (1 - \beta_u)\mathcal{N}(\mathbf{x}_n|\mathbf{x}_{n-1}, \mathbf{\Lambda}) + \beta_u\mathcal{U}_{\mathcal{X}}(\mathbf{x}_n), \quad (14)$$

where $\mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, $\mathcal{U}_{\mathcal{A}}(\cdot)$ denotes the uniform distribution over the set \mathcal{A} , $0 \leq \beta_u \leq 1$ is the weight of the uniform component, and $\mathbf{\Lambda} = \text{diag}(\sigma_x^2, \sigma_y^2, \sigma_\alpha^2)$ is the diagonal matrix with the variances for the random walk models on the components of the object state. The weight of the uniform component is typically set to be small. The object model is completed by the specification of the distribution for the initial state, and here we assume it to be uniform over the state-space, *i.e.*, $p(\mathbf{x}_0) = \mathcal{U}_{\mathcal{X}}(\mathbf{x}_0)$.

C. Color Cues

When a specific class of objects with distinctive shape is considered and a complete model of this shape can be learned

offline, contour cues are very powerful to capture the visual appearance of tracked entities [2], [3], [25], [30], [40], [44], [53]. They can, however, be dramatically contaminated by clutter edges, even when detailed silhouette models are used [52]. Also, they are not adapted to scenarios where there is no predefined class of objects to be tracked, or where the class of objects of interest does not exhibit very distinctive silhouettes. The two tracking scenarios we consider here fall respectively in these two categories.

When shape modeling is not appropriate, color information is a powerful alternative to characterize the appearance of tracked objects. As demonstrated for example in [4], [9], [10], [39], robust tracking can be achieved using only a simple color model constructed in the first frame, even in presence of dramatic changes of shape and pose. Hence, the color features of an object or a region of interest often form a very persistent localization cue. In the two tracking scenarios we are interested in, color information is naturally chosen as the primary ingredient. In this section we derive the color likelihood model that we use.

If color cues are powerful for tracking, their simplicity sometimes results in a lack of discriminative power when it comes to (re)initialize the tracker. In Sections III-D and III-E we show how motion or sound cues can be combined with color cues to resolve ambiguities and increase the robustness of the tracker in two distinct scenarios. In particular, the good *detection* properties offered by these two auxiliary modalities will be fully exploited in the design of good proposal densities.

Color localization cues are obtained by associating a reference color model with the object or region of interest. This reference model can be obtained by hand-labeling, or from some automatic detection module. To assess whether a given candidate region contains the object of interest or not, a color model of the same form as the reference model is computed within the region, and compared to the reference model. The smaller the discrepancy between the candidate and reference models, the higher the probability that the object is located inside the candidate region.

For the color modeling we use independent normalized histograms in the three channels of the RGB color space. We denote the B -bin reference histogram model in channel $c \in \{R, G, B\}$ by $\mathbf{h}_{ref}^c = (h_{1,ref}^c \cdots h_{B,ref}^c)$. Recall from Section III-B that the region in the image corresponding to any state \mathbf{x} is given by $B_{\mathbf{x}}$. Within this region an estimate for the histogram color model, denoted by $\mathbf{h}_{\mathbf{x}}^c = (h_{1,\mathbf{x}}^c \cdots h_{B,\mathbf{x}}^c)$, can be obtained as

$$h_{i,\mathbf{x}}^c = c_H \sum_{\mathbf{u} \in B_{\mathbf{x}}} \delta_i(b_{\mathbf{u}}^c), \quad i = 1 \cdots B, \quad (15)$$

where $b_{\mathbf{u}}^c \in \{1 \cdots B\}$ denotes the histogram bin index associated with the intensity at pixel location $\mathbf{u} = (x, y)$ in channel c of the color image \mathbf{y}^C , δ_a denotes the Kronecker delta function at a , and c_H is a normalizing constant such that $\sum_{i=1}^B h_{i,\mathbf{x}}^c = 1$.

The color likelihood model must be defined in such a way so as to favor candidate color histograms close to the reference histogram. To this end we need to define a distance metric on histogram models. As is the case in [9] we base this distance

on the Bhattacharyya similarity coefficient, defining it as

$$D(\mathbf{h}_1, \mathbf{h}_2) = \left(1 - \sum_{i=1}^B \sqrt{h_{i,1} h_{i,2}}\right)^{1/2}. \quad (16)$$

In contrast to the Kullback-Leibler divergence this distance is a proper metric, it is bounded within $[0, 1]$, and empty histogram bins are not singular. Based on this distance we finally define the color likelihood model as

$$p(\mathbf{y}^C | \mathbf{x}) \propto \exp \left(- \sum_{c \in \{R, G, B\}} D^2(\mathbf{h}_{\mathbf{x}}^c, \mathbf{h}_{ref}^c) / 2\sigma_C^2 \right). \quad (17)$$

The assumption that the squared distance is exponentially distributed is based on empirical evidence gathered over a number of successful tracking runs. The histogram based definition of the color likelihood is summarized in Fig. 3.

If the object of interest is comprised of patches of distinct color, such as the face and clothes of a person, the histogram based color model will still successfully capture the color information. However, all the information about the relative spatial arrangement of these different patches will be lost. Keeping track of the coarse spatial layout of the distinct color regions may benefit the performance of the tracking algorithm. Such a goal is easily achieved within our modeling framework by splitting the region of interest into subregions, each with its own reference color model. More formally, we consider the partition $B_{\mathbf{x}} = \cup_{j=1}^{N_R} B_{j,\mathbf{x}}$, associated with the set of reference color histograms $\{\mathbf{h}_{j,ref}^c : c \in \{R, G, B\}, j = 1 \cdots N_R\}$. The subregions are rigidly linked. It is, however, possible to introduce additional state variables to model the relative movement and scaling of the subregions, should this be necessary. By assuming conditional independence of the color measurements within the different subregions defined by the state \mathbf{x} the multi-region color likelihood becomes

$$p(\mathbf{y}^C | \mathbf{x}) \propto \exp \left(- \sum_{c \in \{R, G, B\}} \sum_{j=1}^{N_R} D^2(\mathbf{h}_{j,\mathbf{x}}^c, \mathbf{h}_{j,ref}^c) / 2\sigma_C^2 \right), \quad (18)$$

where the histogram $\mathbf{h}_{j,\mathbf{x}}$ is collected in the region $B_{j,\mathbf{x}}$.

This color likelihood model is in contrast with the foreground-background models developed in [23], [55]. For any hypothesized state the latter models evaluate the pixels (or grid nodes) covered by the object under some reference foreground model, and the remaining pixels (or grid nodes) under location dependent background models. The likelihood for the scene is obtained by multiplying the individual pixel (or grid node) likelihoods under the independence assumption. Even though this type of likelihood is more principled, it suffers from numerical instabilities, and in our experience the histogram based color model proposed here is a more powerful tracking cue.

The histogram based color measurements can also be used to construct an efficient proposal for the particle filter, guiding it towards regions in the state-space that are characterized by a color distribution similar to the object of interest [21], [39]. In our setting such a distribution will be of the same form as the one for the motion measurements described in Section III-D. However, due to the ambiguity inherent in the color

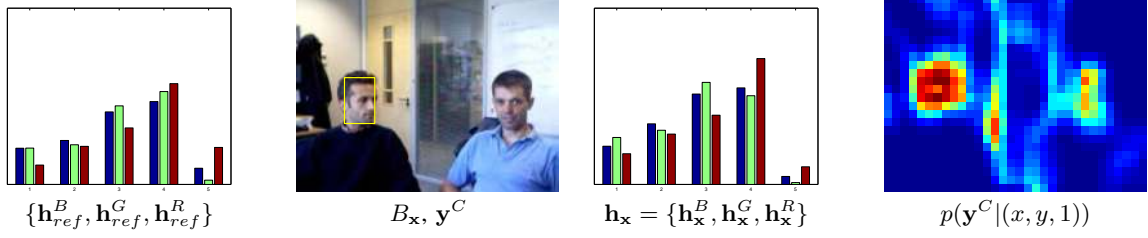


Fig. 3. **From color histograms to color likelihood.** (Left) A three-fold reference color histogram $\mathbf{h}_{ref} = \{\mathbf{h}_{ref}^B, \mathbf{h}_{ref}^G, \mathbf{h}_{ref}^R\}$ is either gathered in an initial frame (within a region picked manually or automatically) or learned offline (e.g., skin color). (Middle) Later in the sequence, and for a hypothesized state \mathbf{x} , the candidate color histogram $\mathbf{h}_{\mathbf{x}}$ is gathered within the region $B_{\mathbf{x}}$ and compared to the reference histogram using the Bhattacharyya similarity measure. (Right) The exponentiated similarity yields the color likelihood, plotted here on a subsampled grid as a function of the location only (scale factor fixed to $\alpha = 1$). Note the ambiguity: strong responses arise over both faces and a section of the door.

measurements for the two tracking scenarios considered here, we prefer to use the more intermittent, but less ambiguous, motion and sound measurements to construct proposal distributions.

D. Motion Cues

Beside color, instantaneous motion activity captures other important aspects of the sequence content, and has been extensively studied from various perspectives. In particular, the problem of motion detection, *i.e.*, the detection of objects moving relative to the camera, is covered by an abundant literature, which we shall not attempt to review here (see the review in [29]). In the case of a static camera, the basic ingredient at the heart of such an analysis is the absolute frame difference computed on successive pairs of images. This is this cue we consider here.

We propose to embed the frame difference information in a likelihood model similar to the one developed for the color measurements. It unifies the description and implementation, and ensures a similar order of magnitude for the two visual cues. Alternative models can easily be accommodated.

We will denote by \mathbf{y}_n^M the absolute difference of the luminances at times n and $n - 1$. As was the case for the color model, a measurement histogram $\mathbf{h}_{\mathbf{x}}^M = (h_{1,\mathbf{x}}^M \cdots h_{B,\mathbf{x}}^M)$ is associated with the region implied by the state \mathbf{x} . The region $B_{\mathbf{x}}$ within which the color information is collected will often lie inside the object of interest. In contrast, a large part of the motion activity generated by a moving object is concentrated along the silhouette of the object. To ensure that the silhouette is included we consider a larger region for the motion measurements, *i.e.*, $\tilde{B}_{\mathbf{x}} = (x, y, \alpha(w^* + \eta), \alpha(h^* + \eta))$, with η set to a few pixel units (5 in our experiments). The construction or learning of a reference histogram model for the motion measurements is not a straightforward task. The amplitude of these measurements depend on both the appearance of the object (its contours) and its current motion. If the examined region contains no movement, all the measurements will fall in the lower bin of the histogram. When movement commences the measurements usually fall in all bins, with no definitive pattern: uniform regions produce low absolute frame difference values, whereas higher values characterize the contours (both the silhouette and the photometric edges). To accommodate these variations we chose the reference motion

histogram \mathbf{h}_{ref}^M to be uniform, *i.e.*,

$$h_{i,ref}^M = \frac{1}{B}, \quad i = 1 \cdots B. \quad (19)$$

Similar to the color likelihood in (17), we define the motion likelihood as

$$p(\mathbf{y}^M | \mathbf{x}) \propto \exp(-D^2(\mathbf{h}_{\mathbf{x}}^M, \mathbf{h}_{ref}^M) / 2\sigma_M^2). \quad (20)$$

The construction of this likelihood is illustrated in Fig. 4.

Motion Proposal: In the majority of cases the perceived motion of the object of interest in the image sequence will satisfy some minimum smoothness constraints. In these cases a proposal distribution that mimics the characteristics of the state evolution model should be sufficient for successful tracking. However, it often happens that lock is lost due to short periods of partial or total occlusion, and it is then necessary to reinitialize the tracker. In another setting such as tele-surveillance, the objects of interest might be moving entities. In both cases it is useful to design a more sophisticated proposal distribution that exploits the motion measurements to allow jumps in the state-space to regions of significant motion activity.

We build such a proposal distribution by evaluating the histogram similarity measure on a subset of locations over the image, keeping the scale factor fixed to $\alpha = 1$. These locations are taken as the nodes of a regular grid for which the step size depends on the affordable computational load. Typically we used a step size of 10 pixel units. Using simple thresholding, locations that satisfy $D^2(\mathbf{h}_{(x,y,1)}^M, \mathbf{h}_{ref}^M) > \tau$ are retained. Based on these locations of high motion activity, denoted by $\mathbf{p}_i = (x_i, y_i)$, $i = 1 \cdots N_M$, we define a mixture proposal for the object location (x, y) as

$$\begin{aligned} q^M(x_n, y_n | x_{n-1}, y_{n-1}, \mathbf{y}_n^M) = & \beta_{RW} \mathcal{N}((x_n, y_n) | (x_{n-1}, y_{n-1}), (\sigma_x^2, \sigma_y^2)) \\ & + \frac{(1 - \beta_{RW})}{N_M} \sum_{i=1}^{N_M} \mathcal{N}((x_n, y_n) | \mathbf{p}_i, (\sigma_x^2, \sigma_y^2)). \end{aligned} \quad (21)$$

The first component is the same Gaussian random walk used for the (x, y) location in the state evolution model in (14), and ensures the smoothness of the motion trajectories. The second component is a mixture centered on the detected locations of high motion activity. When no such location is found ($N_M = 0$), the weight of the random walk component is set to one.

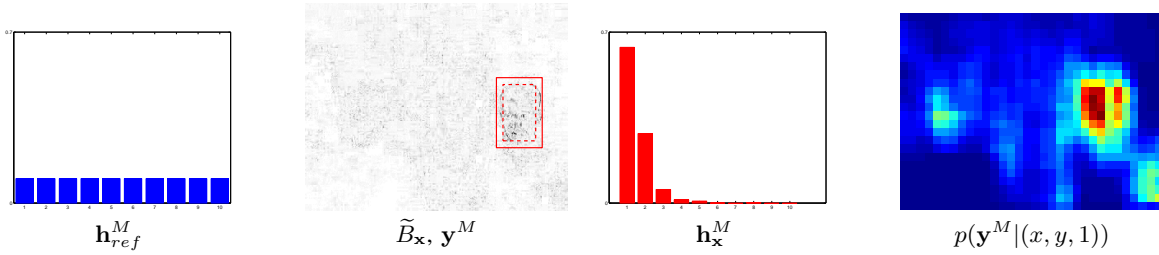


Fig. 4. **From frame difference histograms to motion likelihood.** (Left) Uniform reference histogram. (Middle) At a given instant of the sequence, and for a hypothesized state \mathbf{x} , the candidate motion histogram $\mathbf{h}_{\mathbf{x}}^M$ is gathered within the extended region $\tilde{B}_{\mathbf{x}}$, and compared to the reference histogram using the Bhattacharyya similarity measure. (Right) The exponentiated similarity yields the motion likelihood, plotted here on a subsampled grid as a function of the location only (scale factor fixed to $\alpha = 1$).

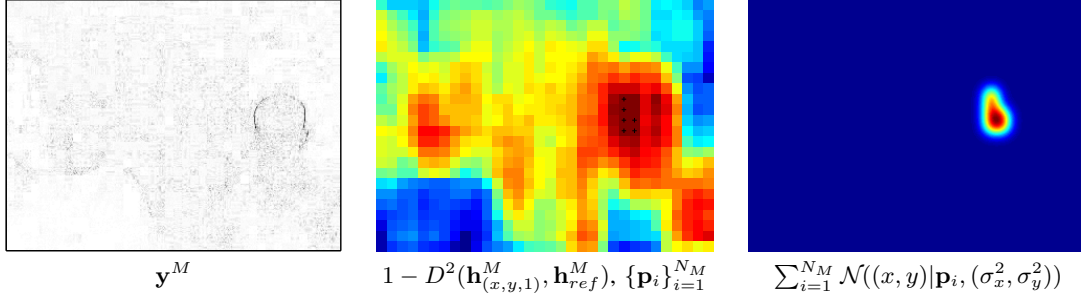


Fig. 5. **From frame difference histograms to motion proposal.** (Left) Absolute luminance frame difference at a given instant. (Middle) Similarity measure between the candidate histogram and the uniform reference histogram, plotted on a subsampled grid as a function of the location only (scale factor fixed to $\alpha = 1$). Locations of high motion activity are detected by thresholding this function ($\tau = 0.9$), and indicated with crosses. (Right) Mixture of Gaussians around the high motion activity locations, forming the motion based proposal distribution.

Fig. 5 illustrates the construction of this motion based proposal distribution.

E. Sound Cues

This section describes the sound localization cues, following [54]. As is the case for the motion cues, the sound cues are intermittent, but can be very discriminating when present. The sound localization cues are obtained by measuring the Time Delay of Arrival (TDOA) between the audio signals arriving at the two microphones comprising the pair. Due to the configuration of the system the TDOA gives an indication of the bearing of the sound source relative to the microphone pair, which can in turn be related to a horizontal position in the image. In what follows we first describe the TDOA measurement procedure. We then derive a likelihood model for the TDOA measurements. Finally, we develop an efficient TDOA based proposal for the particle filter, based on an inversion of the likelihood model. This proposal is especially useful for initialization and recovering of lock in cases where track is lost during brief periods of partial or total occlusion. It can also be used to shift the focus between different speakers as they take turns in a conversation.

1) *TDOA Measurement Process:* Numerous strategies are available to measure the TDOA between the audio signals arriving at spatially separated microphones. For our tracking application the TDOA estimation strategy is required to be computationally efficient, and should not make strong assumptions about the number of audio sources and the exact characteristics of the audio signals. One popular strategy that satisfies these requirements involves the maximization of the Generalized Cross-Correlation Function (GCCF) [5], [27].

This strategy, with various modifications, has been applied with success in a number of sound source localization systems, *e.g.*, [42], [47], [51], [57]. We subsequently adopt it to obtain TDOA estimates for our tracking algorithm. Computation of the GCCF is described at length in [5], [27], and we will omit the detail here.

The GCCF is essentially the correlation function between pre-whitened versions of the signals arriving at the microphones. The positions of the peaks in the GCCF can then be taken as estimates of the TDOAs for the sound sources within the acoustic environment of the microphones. Apart from the true audio sources, “ghost sources” due to reverberation also contribute to the peaks in the GCCF. Rather than attempting to remove these by further signal processing, we retain all the peaks above a predefined threshold as candidates for the true TDOA. More specifically, for any frame the sound measurement vector can be denoted as $\mathbf{y}^S = (D_1 \cdots D_{N_S})$, with $D_i \in \mathcal{D} = [-D_{\max}, D_{\max}]$, $i = 1 \cdots N_S$. The maximum TDOA that can be measured is easily obtained from the microphone separation d and the value used for the speed of sound c (normally 342 ms^{-1}) as $D_{\max} = d/c$. Note that the number of TDOA measurements varies with time, and quite frequently no TDOA measurements will be available. To cope with the ambiguity due to the presence of multiple candidates for the true TDOA we develop a multi-hypothesis likelihood model, which is described next.

2) *TDOA Likelihood Model:* In this section we develop a multi-hypothesis likelihood model for the TDOA measurements. Similar likelihood models have been developed before for radar based tracking in clutter [17], and tracking single [20] and multiple [32] objects in video sequences.

The TDOA measurements depend only on the x position in the image, *i.e.*, $p(\mathbf{y}^S|\mathbf{x}) = p(\mathbf{y}^S|x)$. Furthermore, for any hypothesis of the object x position in the image, a deterministic hypothesis for the TDOA can be computed as

$$D_x = g(x) = g_3 \circ g_2 \circ g_1(x), \quad (22)$$

with

$$\begin{aligned} \tilde{x} &= g_1(x) = \tilde{W}(x/W - 0.5) \\ \theta &= g_2(\tilde{x}) = \arctan(f/\tilde{x}) \\ D_x &= g_3(\theta) = D_{\max} \cos \theta. \end{aligned} \quad (23)$$

The function g_1 is a simple linear mapping that relates the x position in the image to the corresponding \tilde{x} position in the camera image plane. The width of the image W is measured in pixels, whereas the width of the image plane \tilde{W} is measured in metric units. Using a pinhole model for the camera, with f the focal length, the function g_2 then relates the \tilde{x} position in the camera image plane to the sound source bearing. Finally, the function g_3 makes use of the Fraunhofer approximation to relate the sound source bearing to the hypothesized TDOA. Since the mapping g is entirely deterministic the likelihood can be written as $p(\mathbf{y}^S|x) = p(\mathbf{y}^S|D_x)$. We will use this latter form of the likelihood in the exposition below.

We assume the candidate TDOA measurements to be independent, so that the likelihood can be factorized as

$$p(\mathbf{y}^S|D_x) = \prod_{i=1}^{N_S} p(D_i|D_x). \quad (24)$$

In practice, however, clutter measurements due to reverberation are expected to be at least somewhat coherent with the true source, thus violating the independence assumption. Accurate modeling of reverberation requires detailed knowledge about the composition and acoustic properties of the environment, which is difficult to obtain in practice, and thus not attempted here. Nevertheless, the model still performed well, as we will demonstrate in Section IV.

Of the TDOA measurements at most one is associated with the true source, while the remainder is associated with clutter. To distinguish between the two cases we introduce a classification label c_i , such that $c_i = T$ if D_i is associated with the true source, and $c_i = C$ if D_i is associated with clutter. The likelihood for a measurement from the true source is taken to be

$$p(D_i|D_x, c_i = T) = c_x \mathcal{N}(D_i|D_x, \sigma_D^2) \mathbb{I}_{\mathcal{D}}(D_i), \quad (25)$$

where $\mathbb{I}_{\mathcal{A}}(\cdot)$ denotes the indicator function for the set \mathcal{A} , and the normalizing constant c_x is given by

$$c_x = 2 \left(\operatorname{erf} \left(\frac{D_{\max} - D_x}{\sqrt{2}\sigma_D} \right) - \operatorname{erf} \left(\frac{-D_{\max} - D_x}{\sqrt{2}\sigma_D} \right) \right)^{-1}, \quad (26)$$

with $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2) dt$ the Gaussian error function. Thus within the range of admissible TDOA values the measurement is assumed to be the true TDOA corrupted by additive Gaussian observation noise of variance σ_D^2 . Empirical studies proved this to be a reasonable assumption, as the results in Fig. 6 show.

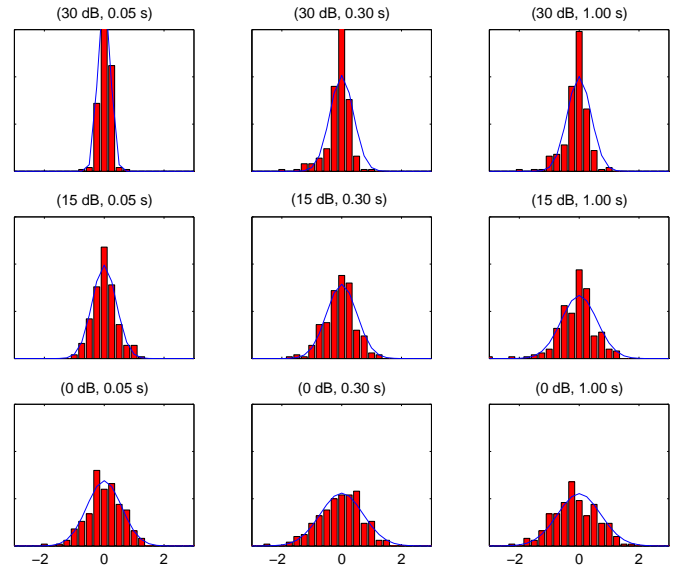


Fig. 6. **TDOA measurement statistics.** TDOA measurement error histograms and Gaussian approximations for a range of signal-to-noise levels and reverberation times.

Similar to what was done in [32] for example, the likelihood for measurements associated with clutter is taken to be

$$p(D_i|c_i = C) = \mathcal{U}_{\mathcal{D}}(D_i). \quad (27)$$

Thus the clutter is assumed to be uniformly distributed within the admissible interval, independent of the true source TDOA.

For N_S measurements there are a total of $N_S + 1$ possible hypotheses. Either all the measurements are due to clutter, or one of the measurements corresponds to the true source, and the remainder to clutter. More formally,

$$\begin{aligned} \mathcal{H}_0 &= \{c_i = C : i = 1 \cdots N_S\} \\ \mathcal{H}_i &= \{c_i = T, c_j = C : j = 1 \cdots N_S, j \neq i\}, \end{aligned} \quad (28)$$

with $i = 1 \cdots N_S$. The likelihoods for these hypotheses follow straightforwardly from (24), and are given by

$$\begin{aligned} p(\mathbf{y}^S|\mathcal{H}_0) &= \mathcal{U}_{\mathcal{D}^{N_S}}(\mathbf{y}^S) \\ p(\mathbf{y}^S|D_x, \mathcal{H}_i) &= c_x \mathcal{N}(D_i|D_x, \sigma_D^2) \mathbb{I}_{\mathcal{D}}(D_i) \mathcal{U}_{\mathcal{D}^{N_S-1}}(\mathbf{y}_{-i}^S), \end{aligned} \quad (29)$$

where \mathbf{y}_{-i}^S is \mathbf{y}^S with D_i removed.

However, for any set of measurements the correct hypothesis is not known beforehand, and the final likelihood is obtained by summing over all the possible hypotheses, *i.e.*,

$$p(\mathbf{y}^S|D_x) = \sum_{i=0}^{N_S} p(\mathcal{H}_i|D_x) p(\mathbf{y}^S|D_x, \mathcal{H}_i), \quad (30)$$

where $p(\mathcal{H}_i|D_x)$ is the prior probability for the i -th hypothesis. In what follows we fix the prior probability for the all clutter hypothesis to P_c , and set the prior probabilities for the remaining N_S hypotheses to be equal. Under these assumptions the likelihood for the TDOA measurements finally becomes

$$p(\mathbf{y}^S|D_x) \propto \frac{P_c}{2D_{\max}} + \frac{c_x(1 - P_c)}{N_S} \sum_{i=1}^{N_S} \mathcal{N}(D_i|D_x, \sigma_D^2) \mathbb{I}_{\mathcal{D}}(D_i). \quad (31)$$

In the case where no TDOA measurements are available the likelihood is simply set to $p(y^S|D_x) \propto 1$. The construction of the TDOA likelihood is illustrated in Fig. 7.

3) *TDOA Proposal*: As was the case for motion, it is possible to use the sound localization cues to design an efficient proposal distribution for the particle filter. Such a proposal would allow the tracker to recover lock after brief periods of partial or total occlusion. In another setting the objects of interest might be speakers participating in a video tele-conference. In this case a sound based proposal can aid the tracker to switch focus between the speakers as they take turns in the conversation.

These objectives can be achieved by designing a proposal distribution for the object x position that incorporates the TDOA measurements when they are available. Informally, the inverse of the mapping in (22) is easy to obtain. Each TDOA measurement can then be passed through the resulting inverse mapping g^{-1} to yield a plausible x position for the object. To capture the notion of smooth motion trajectories and exploiting the information in the TDOA measurements we define a TDOA based proposal for the object x position of the form

$$q^S(x_n|x_{n-1}, \mathbf{y}_n^S) = \beta_{RW} \mathcal{N}(x_n|x_{n-1}, \sigma_x^2) + \frac{(1 - \beta_{RW})}{N_S} \left| \frac{dg(x_n)}{dx_n} \right| \sum_{i=1}^{N_S} \mathcal{N}(g(x_n)|D_{i,n}, \sigma_D^2). \quad (32)$$

The first component is the same Gaussian random walk used for the x component in the state evolution model in (14), and ensures the smoothness of the motion trajectories. The second component is a mixture that incorporates the TDOA measurements, and is obtained by inverting the non-clutter part of the likelihood model in (31). All the TDOA measurements are equally weighted in the mixture. The derivative of the mapping g is easily obtained by using the chain rule. The weight of the random walk component is set to one if no measurements are available, in which case no jumps are allowed in the object x position.

Generating samples from the TDOA based proposal is straightforward. First a mixture component is picked by sampling from the discrete distribution equal to the mixture weights. Sampling from the random walk component is trivial. Sampling from the component for the i -th TDOA measurement can be achieved by first sampling a candidate delay according to $D_{x_n} \sim \mathcal{N}(D_{x_n}|D_{i,n}, \sigma_D^2)$, and then passing the resulting delay through the inverse of the mapping in (22), i.e., $x_n = g^{-1}(D_{x_n})$.

F. Tracker Architecture

We conclude this section by summarizing the composition of our tracking algorithm. We consider two main scenarios. The first, summarized in Tab. III, is a desktop setting such as depicted in Fig. 2, where we use color as the main cue, and fuse it with the information in the sound localization cue. Such a setting forms the basis for video tele-conferencing applications. The second setting, summarized in Tab. IV, is representative of surveillance and monitoring applications involving a static camera. Here sound localization cues will

With $\{\mathbf{x}_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^{N_p}$ the particle set at the previous time step, proceed as follows at time n :

Proposition: simulate $x_n^{(i)} \sim q^S(x_n|x_{n-1}^{(i)}, \mathbf{y}_n^S)$.

Update weights:

$w_n^{(i)} \propto w_{n-1}^{(i)} \frac{p(\mathbf{y}_n^S|x_n^{(i)})p(x_n^{(i)}|x_{n-1}^{(i)})}{q^S(x_n^{(i)}|x_{n-1}^{(i)}, \mathbf{y}_n^S)}$, with $\sum_{i=1}^{N_p} w_n^{(i)} = 1$.

Resample:

simulate $a_i \sim \{w_n^{(k)}\}_{k=1}^{N_p}$, and replace $\mathbf{x}_n^{(i)} \leftarrow \mathbf{x}_n^{(a_i)}$.

Proposition:

simulate $(y_n^{(i)}, \alpha_n^{(i)}) \sim p(y_n, \alpha_n|y_{n-1}^{(i)}, \alpha_{n-1}^{(i)})$, and assemble $\mathbf{x}_n^{(i)} \leftarrow (x_n^{(i)}, y_n^{(i)}, \alpha_n^{(i)})$.

Update weights: $w_n^{(i)} \propto p(\mathbf{y}_n^C|\mathbf{x}_n^{(i)})$, with $\sum_{i=1}^{N_p} w_n^{(i)} = 1$.

If resampling:

simulate $a_i \sim \{w_n^{(k)}\}_{k=1}^{N_p}$, and replace $\{\mathbf{x}_n^{(i)}, w_n^{(i)}\} \leftarrow \{\mathbf{x}_n^{(a_i)}, \frac{1}{N_p}\}$.

TABLE III

PARTICLE FILTER FOR VISUAL TRACKING BASED ON COLOR AND SOUND.

With $\{\mathbf{x}_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^{N_p}$ the particle set at the previous time step, proceed as follows at time n :

Proposition: simulate $\mathbf{p}_n^{(i)} \sim q^M(\mathbf{p}_n|\mathbf{p}_{n-1}^{(i)}, \mathbf{y}_n^M)$.

Update weights:

$w_n^{(i)} \propto w_{n-1}^{(i)} \frac{p(\mathbf{y}_n^M|\mathbf{p}_n^{(i)}, \alpha_{n-1}^{(i)})p(\mathbf{p}_n^{(i)}|\mathbf{p}_{n-1}^{(i)})}{q^M(\mathbf{p}_n^{(i)}|\mathbf{p}_{n-1}^{(i)}, \mathbf{y}_n^M)}$, with $\sum_{i=1}^{N_p} w_n^{(i)} = 1$.

Resample:

simulate $a_i \sim \{w_n^{(k)}\}_{k=1}^{N_p}$, and replace $\mathbf{x}_n^{(i)} \leftarrow \mathbf{x}_n^{(a_i)}$.

Proposition:

simulate $\alpha_n^{(i)} \sim p(\alpha_n|\alpha_{n-1}^{(i)})$, and assemble $\mathbf{x}_n^{(i)} \leftarrow (\mathbf{p}_n^{(i)}, \alpha_n^{(i)})$.

Update weights: $w_n^{(i)} \propto p(\mathbf{y}_n^C|\mathbf{x}_n^{(i)})$, with $\sum_{i=1}^{N_p} w_n^{(i)} = 1$.

If resampling:

simulate $a_i \sim \{w_n^{(k)}\}_{k=1}^{N_p}$, and replace $\{\mathbf{x}_n^{(i)}, w_n^{(i)}\} \leftarrow \{\mathbf{x}_n^{(a_i)}, \frac{1}{N_p}\}$.

TABLE IV

PARTICLE FILTER FOR VISUAL TRACKING BASED ON COLOR AND MOTION.

NOTATION \mathbf{p} STANDS FOR LOCATION (x, y) .

generally be absent, and we thus fuse color with the motion localization cue.

In both settings we employ a form of partitioned sampling. In the first setting the sound likelihood only gives information about the object x coordinate in the image. We thus simulate and resample for this component first, before simulating new values for the remaining state components (object y coordinate and scale factor α) and resampling with respect to the color likelihood. This implies that we do not search directly in the three dimensional state-space, but rather partition the inference into a search in a one dimensional space, followed by another in a two dimensional space. In general this increases the efficiency of the particle filter, allowing us to achieve the same accuracy for a smaller number of particles. We follow a similar strategy in the second setting where we fuse color and motion in that we first simulate and resample the location parameters with respect to the motion likelihood, before simulating the scale factor α and resampling with respect to the color likelihood.

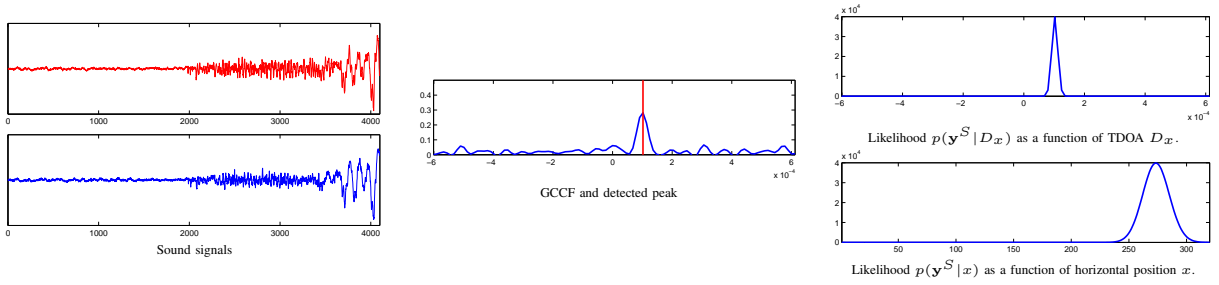


Fig. 7. **From sound measurements to TDOA likelihood.** (Left) A stereo pair of 4096 sound samples corresponding to the 1/30 seconds of one video frame. (Middle) GCCF of the two stereo signals, and detected peak. (Right) Associated likelihood as a function of the TDOA, and of the horizontal pixel coordinate in the image frame, respectively.

IV. DEMONSTRATIONS

In this section we will demonstrate the performance of our tracking algorithm on a number of challenging video sequences. We will first consider the behavior of the tracker when using each of the cues in isolation, and then show how the shortcomings of such single modality trackers can be eliminated by fusing the information from multiple cues. We will use the values in Tab. V for the fixed parameters of the likelihood, proposal and state evolution. Due to the efficiency of the motion and sound proposals, and the relatively low dimensionality of the state-space, good tracking results can be achieved with a reasonably small number of particles. For our experiments we use $N_p = 100$ particles.

A. Single Modality Tracking

1) *Color Only*: Since the color of an object is its most persistent feature, we use it as the main cue for our tracking algorithm. As is the case for the deterministic color based trackers proposed in [4], [6], [9], our probabilistic tracker using color only is able to robustly track objects undergoing complex changes of shape and appearance. This is exemplified by the sequence of the football player in the top row of Fig. 8. The robustness of the tracker to color clutter and large movements is further increased by its ability to incorporate multi-region color models, as is illustrated in the home video example in the bottom row of Fig. 8.

The downside of the color persistence is its lack of discriminating power in certain settings. In cases where the background contains objects of similar color characteristics to the object of interest, the likelihood will exhibit a number of modes. A typical example is given in the office scene in Fig. 3, where the wooden door is close in color space to the faces of the subjects being tracked. Under these circumstances the ambiguities might lead to inaccurate tracking, or in the worst case, a complete loss of lock.

In these scenarios the robustness of the tracking algorithm can be greatly increased by fusing color cues with other cues exhibiting complementary properties, *i.e.*, with lower persistence, but being less prone to clutter. As discussed before, motion and sound are two such cues. The former will prove to be a valuable addition to color in static camera settings, such as tele-surveillance and smart environment monitoring, whereas the latter will combine with color in tele-conferencing applications, where a calibrated stereo microphone pair can easily be

added to the broadcasting or recording equipment. We will first demonstrate the power and limitations of motion and sound as single modality tracking cues, and then investigate the two fusion scenarios.

2) *Motion Only*: In this section we illustrate the behavior of the tracker using the frame difference motion cue only. With this cue moving objects are tracked with a reasonable degree of accuracy, as the result for the sequence in Fig. 9 indicates. However the motion cue is intermittent, and when motion ceases there is no more localization information, and the particles diffuse under their dynamics, as is illustrated in Fig. 10. We will see in Section IV-B.1 how the fusion of motion with color effectively eliminates these shortcomings.

3) *Sound Only*: In this section we illustrate the behavior of the sound only tracker, *i.e.*, tracking only the horizontal position of a speaker in the image, based on the TDOA measurements obtained from the stereo microphone pair. As will be evident, this cue not only allows localization in the x coordinate, but also endows the tracker with the ability to switch focus between speakers as they take turns in the conversation.

We consider two sequences, each featuring two subjects in discussion in front of a camera. In the first sequence the environment is relatively noise-free, and speech is consistently detected when present. Fig. 11 shows snapshots of the tracking result for this sequence. It is clear that the sound based proposal defined in (32) allows the tracker to maintain focus on the current speaker to a sufficient degree of accuracy. It also facilitates the switching of focus between the speakers as they alternate in conversation. Note that the tracking is accurate even beyond the limit of the image plane, as is evident for the subject on the right.

This result is even more encouraging in the light that it was obtained using low cost off-the-shelf equipment. Extreme care in the placement of the microphones relative to the camera was not required. The system was only roughly calibrated, and proved to be robust to the exact values chosen for the intrinsic parameters of the camera. Furthermore, no explicit attempts were made to compensate for the reverberation or background noise.

In the second sequence, for which snapshots are given in Fig. 12, the signal-to-noise ratio is higher due to a higher level of air-conditioner noise in the background, and the subjects being further from the microphones. Due to the higher noise level speech is often undetected ($N_S = 0$). This is further

standard deviations of dynamics	$(\sigma_x, \sigma_y, \sigma_\alpha) = (5, 5, 10^{-2})$
uniform weight in dynamics	$\beta_u = 0.01$
random walk weight in proposals	$\beta_{RW} = 0.75$
standard deviations of likelihoods	$(\sigma_C, \sigma_M, \sigma_D) = (2 \times 10^{-2}, 2 \times 10^{-2}, 10^{-5})$
clutter probability in sound likelihood	$P_c = 10^{-3}$
speed of sound	$c = 342m s^{-1}$
microphone separation	$d = 0.2m$
focal length	$\tilde{f} = 0.02m$
width of the image plane	$\tilde{W} = 0.01m$

TABLE V
MODEL PARAMETERS FOR THE TRACKING EXPERIMENTS.



Fig. 8. **Color based tracking with single and multi-region reference models.** Using a global color reference model generated from a hand-selected region in the initial frame, a region of interest (player 75 in the top sequence, the child in the bottom sequence) can be tracked robustly despite large motions, significant motion blur, dramatic shape changes, partial occlusions, and distracting colors in the background (other players in the top sequence, the sand and the face of the mother in the bottom sequence). In the top sequence a single region was used, whereas two regions, corresponding to the face and shirt of the child, were used in the bottom sequence. The yellow box in each frame indicates the MMSE estimate.



Fig. 9. **Tracking moving objects with motion cues only.** The uniformly initialized particle filter locks on to the moving vehicle a few frames after it enters the scene, thanks to the motion based proposal. The tracker maintains lock on the moving vehicle as it drives through the parking lot. The yellow box represents the MMSE estimate.

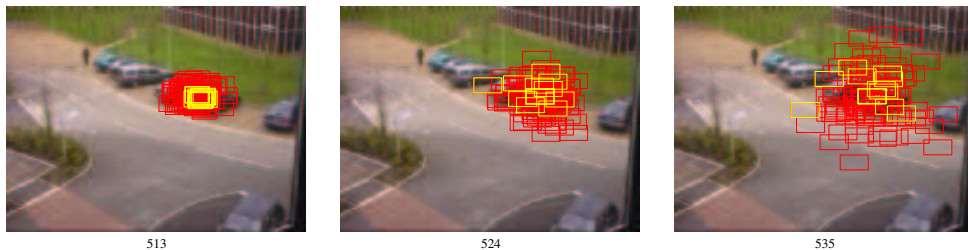


Fig. 10. **Intermittence of motion cues.** When the vehicle tracked in the sequence of Fig. 9 eventually stops, the motion cues disappear, and the particles diffuse under their dynamics. The rectangles indicate the hypothesized regions before resampling, with the yellow rectangles depicting the ten best particles.



Fig. 11. **Sound based tracking under accurate speech detection.** After the first period of silence (frames 1 to 17), where the uniformly initialized particles keep on diffusing, the sound based tracker consistently tracks the horizontal position of the current speaker. The vertical segments indicate the positions of all the particles at the current time step, with the yellow segments depicting the ten best particles. Only particles falling inside the image are displayed. Hence the particle cloud is partly to completely invisible when the tracker correctly follows the speaker on the right exiting and re-entering the field of view of the camera (e.g., frames 251, 274, 298).

exemplified by the graphs in Fig. 13 that show the speech detections for each speaker in relation to the ground truth. These detection failures result in a rapid diffusion of the particles, as is evident from a number of snapshots in Fig. 12.

Since the sound cue lacks persistence, either through detection failures or the absence of speech, the sound based tracker is unable to provide consistent tracking in time. In addition, the sound cue gives no information about the vertical location y or scale factor α of the object being tracked. We will see in Section IV-B.2 how the fusion of sound with color will solve these problems, while retaining the desirable properties of the sound localization cue.

B. Multiple Modality Tracking

1) *Color and Motion:* As we have mentioned in Section IV-A.1, the greatest weakness of the color localization cue is the ambiguity that results due to the presence in the scene of objects or regions with color features similar to those of the object of interest. An extreme example where tracking based on color only fails completely is given in Fig. 14.

By fusing the color and motion cues the ambiguity can be greatly reduced if the object of interest is indeed moving. This is exemplified by the likelihood maps for each of the individual cues and the combination of the cues in Fig. 15. As a further illustration the combination of color and motion allows accurate tracking of the vehicle in Fig. 16, where color only tracking failed. During periods of motion the tracker utilizes mainly the information in the motion cue, while relying on the color information for the localization when the motion ceases at the end of the sequence.

The utility of the motion based proposal is illustrated by the tracking results on the sequence in Fig. 17. In this case the localization information is ambiguous in both the color and motion cues, when considered individually. Without the event based motion proposal a uniformly initialized tracker simply

settles on one of the spurious modes of the combined likelihood. With the help of the motion based proposal, however, the tracker is able to lock on to the target (face) as soon as it enters the scene. Even though the motion based proposal continues to generate hypotheses in all the regions of significant motion activity (face, torso and monitor), those that do not comply with the color likelihood are eliminated during the resampling procedure, so that lock is maintained, as is shown in Fig. 18.

2) *Color and Sound:* To demonstrate the fusion of color and sound we consider again the second sequence presented in Section IV-A.3. We now initialize a reference color model composed of three subregions (for better positioning accuracy) on the face of the left subject, as depicted in Fig. 19. Using color only the particle filter, after a uniform initialization, locks on to one of the subjects at random, and maintains lock on this subject throughout the video sequence, as is shown in Fig. 19.

By incorporating the sound localization cues, the tracker is able to jump between the subjects as they take turns in the conversation, as is depicted in Fig. 20. During the absence of sound cues, either through detection failures or periods of silence, the tracker maintains accurate lock on the last active speaker due to the information in the color cues.

V. CONCLUSIONS

In this paper we introduced generic mechanisms for data fusion within particle filtering, and used them to develop a particle filter based tracker combining color, motion and sound localization cues in a novel way. Color, being the most persistent, was used as the main cue for tracking. The ambiguity inherent in color representations was resolved by the more intermittent, but less ambiguous, motion and stereo sound cues. We considered two main scenarios. In the first, color was combined with sound in a desktop setting that forms the basis for video tele-conferencing applications. In the second, representative of more general tracking applications such as surveillance, we fused color with the motion localization cue.



Fig. 12. **Sound based tracking under sporadic speech detection.** Due to the low signal-to-noise ratio speech is only occasionally detected (e.g., frames 18, 62, 191, 214, 219, 252). The remainder of the time no localization cues are available, and the particle filter simply diffuses under its dynamics (e.g., frames 47, 79, 200, 222, 284).

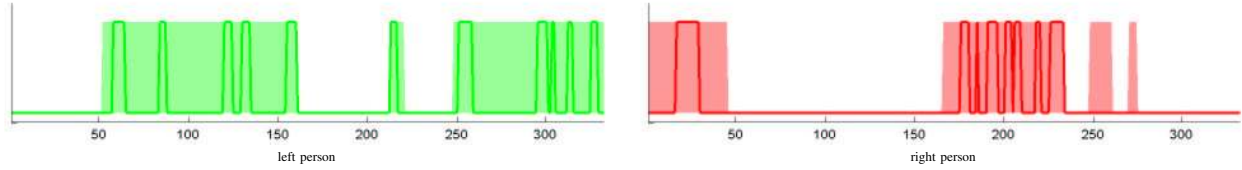


Fig. 13. **Speech detection against ground truth** for the two subjects of the sequence in Fig. 12. The left (resp. right) graph concerns the subject on the left (resp. right) in the scene. The lines indicate the speech detections, which all correspond to correct horizontal localization of the speaker in the image. The shaded area indicates the hand-labeled ground truth.



Fig. 14. **Color tracking under ambiguity.** The reference color histogram, generated from a region selected by hand around the moving vehicle in frame 74, leads to a high degree of ambiguity. Soon after a uniform initialization, the tracking algorithm gets stuck on a very strong local minimum of the color likelihood, corresponding to one of the parked vehicles. The yellow box represents the MMSE estimate.

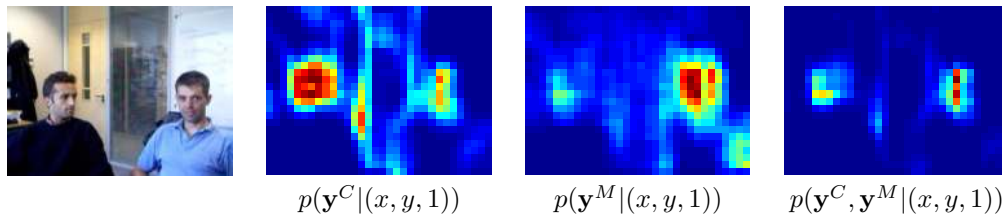


Fig. 15. **Color and motion compound likelihood.** Head movement, as captured by the motion likelihood, helps to reduce the ambiguity inherent in the color cues, as is exemplified by the three likelihood maps: color likelihood, motion likelihood, and the product of the two likelihoods, plotted as a function of the location only (scale factor fixed to $\alpha = 1$).

In both scenarios the combination of cues proved to be more robust than any of the cues individually.

As is standard practice, we defined independent likelihood models for each of the localization cues. We also constructed mixture proposal distributions based on the motion and sound cues, whenever these were available. These proposals act as detection and initialization modules for the particle filter. They also facilitate a more efficient exploration of the state-space, and aid the recovery of lock following periods of partial or complete occlusion. Within the context of multi-object tracking such event based proposals are essential for the detection of new objects when they appear in the scene. We also used different stratified sampling procedures to increase

the efficiency of the particle filter. These layered procedures effectively substitute the difficult estimation problem in the complete state-space with a sequence of easier estimation problems, each in a reduced space.

This combination of appropriate conditionally independent data models, event based proposal densities, and layered importance sampling could now be extended to other visual tracking scenarios involving the fusion of other information, such as shape cues when a predefined class of object has to be tracked.

Although not considered in this paper, the fusion of multiple measurement modalities is an essential requirement in adaptive systems. In such scenarios a high degree of confidence in one

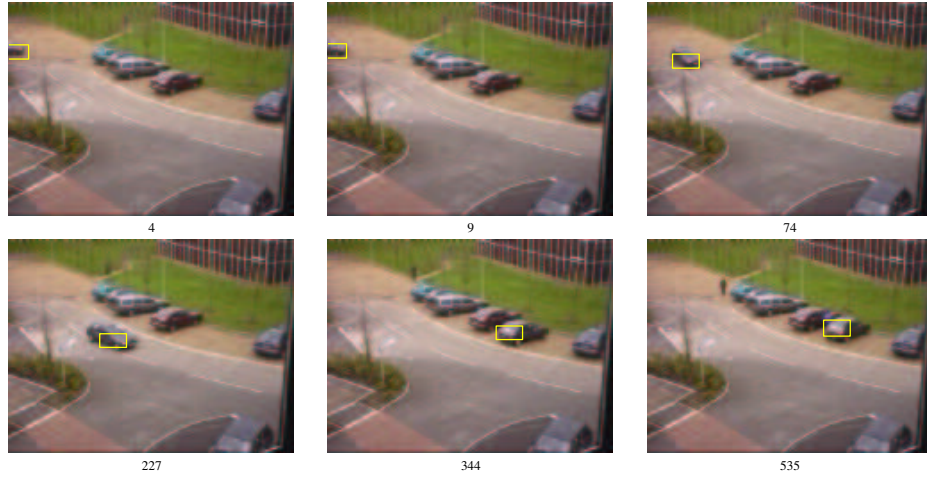


Fig. 16. **Fusing color and motion.** As was the case in Fig. 14, the reference color model is generated from a region selected by hand around the moving vehicle in frame 74. The motion based proposal allows the tracker to lock on to the moving vehicle as soon as it enters the scene. Lock is maintained throughout the period of motion, mostly due to the presence of the motion cues. When motion ceases towards the end of the sequence, the tracker relies on the color cues to maintain lock, in contrast to the motion only tracking in Fig. 10. The yellow box represents the MMSE estimate.



Fig. 17. **Using color and motion cues with the motion based proposal.** (Top) In this example of a person traversing the field of view, the motion cues are very ambiguous due to the activity around the face and torso of the moving person, and around the computer monitor. (Second row) The reference color model, initialized on the face of the person, leads to additional ambiguities. After a uniform initialization the particles settle on a spurious local mode of the color likelihood. It is only towards the end of the sequence, when the face approaches this region, that the particles lock on to the desired target. (Third row) Combining the color and motion modalities, while retaining the smooth proposal, does not alter the behavior of the color only tracker significantly (the particles move to the face a few frames earlier). (Bottom) In contrast, the combined tracker with the motion based proposal allows the tracker to lock on to the face as soon as it enters the scene, and to track it throughout. All regions of high motion activity are continually explored (e.g., face, torso and monitor), but those that do not comply with the color model are discarded during the resampling procedure. The rectangles indicate the hypothesized regions before resampling, with the yellow rectangles depicting the ten best particles.



Fig. 18. **Using color and motion cues with the motion based proposal.** MMSE result for the tracker that combines the color and motion localization cues with the motion based proposal on the sequence in Fig. 17.

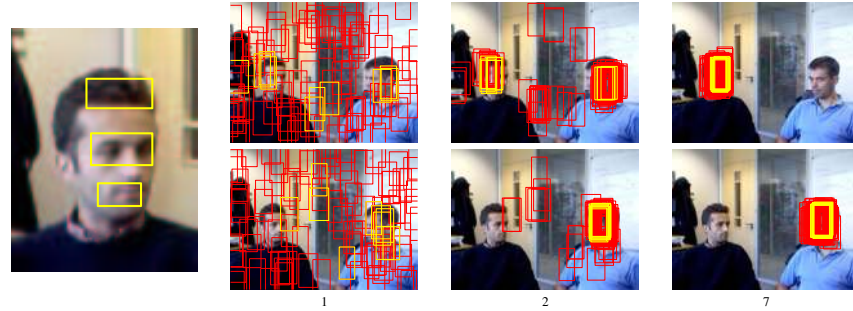


Fig. 19. **Two different runs of the color tracker.** The reference color model is defined on the three-fold selection on the left in frame 1. After a uniform initialization the tracker rapidly locks on to one of the subjects at random, and maintains lock on this subject throughout the sequence. The rectangles indicate the bounding boxes for the particles, with the yellow rectangles depicting the ten best particles.



Fig. 20. **Fusing color and sound.** The tracking result for the fusion of color and sound cues on the same frames as in Fig. 12. The sound based proposal allows the tracker to jump between the subjects as they alternate in conversation (frames 18, 62, 191, 214, 219, 252). In the absence of sound cues lock is maintained on the last active speaker due to the information in the color cues. The yellow rectangle in each of the frames depicts the MMSE estimate of the particle bounding box.

modality facilitates the adaptation of the model parameters associated with complementary modalities, while simultaneously minimizing the risk of learning a wrong appearance model, such as the background [8], [24], [55], [59]. Such adaptive systems are most useful in multi-object tracking systems where it may be desirable to individualize a generic model to each of the objects in the scene.

ACKNOWLEDGMENT

The authors would like to thank Michel Gangnet for his contribution in the experimental phase of the investigation.

REFERENCES

- [1] B. Anderson and J. Moore. *Optimal Filtering*. Prentice-Hall, Englewood Cliffs, 1979.
- [2] A. Blake, R. Curwen, and A. Zisserman. A framework for spatiotemporal control in the tracking of visual contours. *Int. J. Computer Vision*, 11(2):127–145, 1993.
- [3] A. Blake and M. Isard. *Active contours*. Springer, London, 1998.
- [4] G. Bradski. Computer vision face tracking as a component of a perceptual user interface. In *Workshop on Applications of Computer Vision*, pages 214–219, Princeton, NJ, Oct. 1998.
- [5] G. Carter. Coherence and time delay estimation. *Proceedings of the IEEE*, 75(2):236–255, 1987.
- [6] H. Chen and T. Liu. Trust-region methods for real-time tracking. In *Proc. Int. Conf. Computer Vision*, pages II: 717–722, Vancouver, Canada, July 2001.
- [7] K. Choo and D.J. Fleet. People tracking using hybrid Monte Carlo filtering. In *Proc. Int. Conf. Computer Vision*, pages II: 321–328, Vancouver, Canada, July 2001.
- [8] R. Collins and Y. Liu. On-line selection of discriminative tracking features. In *Proc. Int. Conf. Computer Vision*, pages 346–352, Nice, France, October 2003.
- [9] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages II:142–149, Hilton Head, SC, June 2000.
- [10] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Trans. Pattern Anal. Machine Intell.*, 25(5):564–577, 2003.
- [11] J. Crowley, J. Coutaz, and F. Brard. Things that see: Machine perception for human computer interaction. *Communications of the ACM*, 43(3):54–64, 2000.
- [12] S. Das Poddada and R. McDevitt. Lesat average residual algorithm (LARA) for tracking the motion of arctic sea ice. *IEEE. Trans. Geoscience and Remote Sensing*, 34(4):915–926, 1996.
- [13] F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the Condensation algorithm for robust, vision-based mobile robot localization. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages II: 588–594, Fort Collins, CO, June 1999.

- [14] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New-York, 2001.
- [15] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.
- [16] J. Geweke. Bayesian inference in econometrics models using Monte Carlo integration. *Econometrica*, 57:1317–1339, 1989.
- [17] N. Gordon. A hybrid bootstrap filter for target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1):353–358, 1997.
- [18] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, 1993.
- [19] I. Haritaoglu, D. Harwood, and L. Davis. W4: Real-time surveillance of people and their activities. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(8):809–830, 2000.
- [20] M. Isard and A. Blake. CONDENSATION—conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):5–28, 1998.
- [21] M. Isard and A. Blake. ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework. In *Proc. Europ. Conf. Computer Vision*, pages 893–908, 1998.
- [22] M. Isard and A. Blake. A smoothing filter for condensation. In *Proc. Europ. Conf. Computer Vision*, pages 1:767–781, 1998.
- [23] M. Isard and J. MacCormick. BraMBLE: a Bayesian multiple-blob tracker. In *Proc. Int. Conf. Computer Vision*, pages II: 34–41, Vancouver, Canada, July 2001.
- [24] A.D. Jepson, D.J. Fleet, and T.F. El-Maraghi. Robust online appearance models for visual tracking. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages I:415–422, Kauai, Hawaii, December 2001.
- [25] C. Kervrann and F. Heitz. A hierarchical Markov modeling approach for the segmentation and tracking of deformable shape. *Graph. Mod. Image Proc.*, 60(3):173–195, 1998.
- [26] G. Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Computational and Graphical Stat.*, 5(1):1–25, 1996.
- [27] C. Knapp and G. Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-24(4):320–327, 1976.
- [28] E. Koller-Meier and F. Ade. Tracking multiple objects using the Condensation algorithm. *Journal of Robotics and Autonomous Systems*, 34(2-3):93–105, 2001.
- [29] J. Konrad. Motion detection and estimation. In A. Bovik, editor, *Handbook of Image and Video Processing*, pages 207–225. Academic Press, 2000.
- [30] F. Leymarie and M. Levine. Tracking deformable objects in the plane using an active contour model. *IEEE Trans. Pattern Anal. Machine Intell.*, 15(6):617–634, 1993.
- [31] J. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93:1032–1044, 1998.
- [32] J. MacCormick and A. Blake. Probabilistic exclusion and partitioned sampling for multiple object tracking. *International Journal of Computer Vision*, 39(1):57–71, 2000.
- [33] J. MacCormick and M. Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *Proc. Europ. Conf. Computer Vision*, pages 34–41, Dublin, Ireland, June 2000.
- [34] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 D visual servoing. *IEEE Trans. on Robotics and Automation*, 15(2):238–250, 1999.
- [35] E. Marchand and F. Chaumette. Virtual visual servoing: a framework for real-time augmented reality. In *Proc. Eurographics*, pages 289–298, Saarbrücken, Germany, September 2002.
- [36] T. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001.
- [37] C. Papin, P. Boutheymy, and G. Rochard. Unsupervised segmentation of low clouds from infrared meteosat images based on a contextual spatio-temporal labeling approach. *IEEE Trans. on Geoscience and Remote Sensing*, 40(1):104–114, 2002.
- [38] A. Pentland. Looking at people: sensing for ubiquitous and wearable computing. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(1):107–119, 2000.
- [39] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *Proc. Europ. Conf. Computer Vision*, pages I: 661–675, Copenhagen, Denmark, May 2002.
- [40] N. Peterfreund. Robust tracking of position and velocity with Kalman snakes. *IEEE Trans. Pattern Anal. Machine Intell.*, 21(6):564–569, 1999.
- [41] V. Philomin, R. Duraiswami, and L.S. Davis. Quasi-random sampling for Condensation. In *Proc. Europ. Conf. Computer Vision*, pages 134–149, Dublin, Ireland, June 2000.
- [42] D. Rabinkin, R. Renomeron, A. Dahl, J. French, J. Flanagan, and M. Bianchi. A DSP implementation of source location using microphone arrays. In *Proceedings of the SPIE*, volume 2846, pages 88–99, 1996.
- [43] Y. Rui and Y. Chen. Better proposal distributions: Object tracking using unscented particle filter. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages II:786–794, Kauai, Hawaii, December 2001.
- [44] H. Sidenbladh and M. Black. Learning the statistics of people in images and videos. *Int. J. Computer Vision*, 54(1/2/3):183–209, 2003.
- [45] H. Sidenbladh, M.J. Black, and D.J. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In *Proc. Europ. Conf. Computer Vision*, pages II: 702–718, Dublin, Ireland, June 2000.
- [46] H. Sidenbladh, M.J. Black, and L. Sigal. Implicit probabilistic models of human motion for synthesis and tracking. In *Proc. Europ. Conf. Computer Vision*, pages I: 784–800, Copenhagen, Denmark, May 2002.
- [47] H. Silverman and E. Kirtman. A two-stage algorithm for determining talker location from linear microphone array data. *Computer Speech and Language*, 6:129–152, 1992.
- [48] C. Sminchisescu and B. Triggs. Covariance scaled sampling for monocular 3D body tracking. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages I:447–454, Kauai, Hawaii, December 2001.
- [49] C. Sminchisescu and B. Triggs. Hyperdynamics importance sampling. In *Proc. Europ. Conf. Computer Vision*, pages I: 769–783, Copenhagen, Denmark, May 2002.
- [50] M. Spengler and B. Schiele. Towards robust multi-cue integration for visual tracking. In *Int. Workshop on Computer Vision Systems*, Vancouver, Canada, July 2001.
- [51] P. Svaizer, M. Matassoni, and M. Omologo. Acoustic source location in a three-dimensional space using crosspower spectrum phase. In *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, pages 231–234, 1997.
- [52] A. Thayananthan, B. Stenger, Ph. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages 127–133, Madison, WI, June 2003.
- [53] K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *Proc. Int. Conf. Computer Vision*, pages II: 50–57, Vancouver, Canada, July 2001.
- [54] J. Vermaak, A. Blake, M. Gangnet, and P. Pérez. Sequential Monte Carlo fusion of sound and vision for speaker tracking. In *Proc. IEEE Int. Conf. Computer Vision*, pages I:741–746, Vancouver, Canada, July 2001.
- [55] J. Vermaak, P. Pérez, M. Gangnet, and A. Blake. Towards improved observation models for visual tracking: selective adaptation. In *Proc. Europ. Conf. Computer Vision*, pages I: 645–660, Copenhagen, Denmark, May 2002.
- [56] W. Vieux, Schwerdt. K., and J. Crowley. Face-tracking and coding for video compression. In *Proc. Int. Conf. on Computer Vision Systems*, pages 151–160, Las Palmas, Spain, January 1999.
- [57] H. Wang and P. Chu. Voice source localization for automatic camera pointing system in videoconferencing. In *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, pages 187–190, 1997.
- [58] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfunder: Real-time tracking of the human body. *IEEE Trans. Pattern Anal. Machine Intell.*, 19(7):780–785, 1997.
- [59] Y. Wu and T.S. Huang. A co-inference approach to robust visual tracking. In *Proc. Int. Conf. Computer Vision*, pages II: 26–33, Vancouver, Canada, July 2001.



Patrick Pérez was born in 1968. He graduated from École Centrale Paris, France, in 1990 and received the Ph.D. degree at the University of Rennes, France, in 1993. After one year as an Inria post-doctoral researcher in the Dpt of Applied Mathematics at Brown University (Providence, USA), he was appointed at Inria in 1994 as a full time researcher. In 2000, he joined Microsoft Research, Cambridge, U.K. His research interests include probabilistic models for understanding, analysing, and manipulating still and moving images.



multi-media manipulation, statistical signal processing methods and machine learning.

Jaco Vermaak was born in South Africa in 1969. He received the B.Eng. and M.Eng. degrees from the University of Pretoria, South Africa in 1993 and 1996, respectively, and the Ph.D. degree at the University of Cambridge, U.K. in 2000. He has worked as a post-doctoral researcher at Microsoft Research, Cambridge, U.K. between 2000 and 2002. He is currently employed as a senior research associate in the Signal Processing Group of the Cambridge University Engineering Department. His research interests include audio-visual tracking techniques,



of papers in vision, a book with A.Zisserman ("Visual Reconstruction", MIT press), edited "Active Vision" with Alan Yuille (MIT Press) and a book ("Active Contours", Springer-Verlag) with Michael Isard. He was elected Fellow of the Royal Academy of Engineering in 1998.

Andrew Blake was on the faculty of Computer Science at the University of Edinburgh, and also a Royal Society Research Fellow, until 1987, then at the Department of Engineering Science in the University of Oxford, where he became a Professor in 1996, and Royal Society Senior Research Fellow in 1998. In 1999 he was appointed Senior Research Scientist at Microsoft Research, Cambridge, while continuing as visiting Professor at Oxford. His research interests are in computer vision, signal processing and learning. He has published a number