Visual Tracking by Fusing Multiple Cues with Context-Sensitive Reliabilities

Erkurt Erdem[1], Séverine Dubuisson[2] and Isabelle Bloch[3]

[1] *Hacettepe University, Ankara, Turkey.*[1]
[2] *Université Pierre et Marie Curie, Laboratoire d'Informatique de Paris 6, France.*
[3] *Institut TELECOM, Télécom ParisTech, CNRS LTCI, Paris, France*

Abstract

Many researchers argue that fusing multiple cues increases the reliability and robustness of visual tracking. However, how the multi-cue integration is realized during tracking is still an open issue. In this work, we present a novel data fusion approach for multi-cue tracking using particle filter. Our method differs from previous approaches in a number of ways. First, we carry out the integration of cues both in making predictions about the target object and in verifying them through observations. Our second and more significant contribution is that both stages of integration directly depend on the dynamically-changing reliabilities of visual cues. These two aspects of our method allow the tracker to easily adapt itself to the changes in the context, and accordingly improve the tracking accuracy by resolving the ambiguities.

Keywords: Visual tracking, data fusion, multiple cues, particle filter

[1] This work was performed while Erkut Erdem was a post-doctoral researcher at Université Pierre et Marie Curie and at Télécom ParisTech

1. Introduction

Visual tracking is a widely studied topic in computer vision for a wide range of application areas. These include visual surveillance, activity analysis, man-machine interaction, augmented reality, etc. Here we consider the task of locating an object of interest on each frame of a given video sequence. This object of interest can be an actual object in the scene, e.g. a person, or a specific image region of prime importance, e.g. a face. For real-world applications, it is generally accepted that tracking based on a single visual feature would be likely to fail due to the complex nature of the data and the tracking process. Thus, it has been argued in many works that considering multi-modal data leads to an improvement in tracking. It increases the robustness by letting complementary observations from different sources work together. These sources are either the visual features extracted from the same image sequence, such as color and motion cues, or the visual cues coming from different physical sensors, such as from a CCD or from an infrared camera. However, how the information extracted from these sources is combined in tracking is still an open problem.

1.1. Related Work

Tracking methods generally involve two key processes: generating hypotheses through a prediction step and then verifying these hypotheses through some measurements. Considering the vast number of studies in tracking literature, the most general way of performing data fusion is in the measurement step. For example, in an early work [4], Birchfield suggested to combine two orthogonal visual cues (*color* and *intensity gradients*) within a hypothesize-and-test procedure. In these studies, each cue provides a likelihood or a matching score for the possible positions of the object, and the final output is determined by taking into account the product of individual likelihoods or the summation of the matching scores. The main problem with this approach is that all the modalities are given an equal reliability, which is a very unrealistic assumption. Thus, if one of visual cues becomes unreliable, it may result in a wrong estimate.

In tracking literature, different definitions of cue reliability have been proposed. For example, in [2, 19], the authors defined the reliability of a single cue by means of the covariance or the spread of the samples suggested by the cue at each tracking step, measuring its uncertainty. On the other hand, in [10], the cue reliability is considered as a measure specifying the success of the cue in discriminating the object from the surrounding background.

Tracking approaches can be grouped according to the way they employ the cue reliabilities. The first group of works [7, 19, 23, 24, 25] assigns different reliability values to different visual cues, and takes them into consideration in the measurement step. In [24, 25], the authors formulate the fusion as the weighted average of saliency maps extracted for each cue with the weights corresponding to the cues' reliabilities. Hence, the reliabilities are determined by considering the correlation among the visual cues. In other words, cue reliability is defined *relative to the success of the other cues in tracking the target object*. During tracking, different cues try to reach an agreement on a joint result and they adapt themselves considering the result currently agreed on. Similarly, the Sequential Monte Carlo based framework proposed in [7, 19, 23] use adaptive weights for the cues utilized in estimating the combined likelihoods. In this approach, the overall likelihood is more precise since the reliabilities of cues are now taken into account in the computations. On the other hand, the weakness of these studies is that the fusion is carried out only in verifying object hypotheses against observations. The utilized multiple cues are involved in neither making predictions nor generating hypotheses in any way. In terms of robustness, however, this is an important direction that should be pursued as well.

The second line of works [9, 18, 22, 28], indeed, concentrates on this issue and lets the multi-modal data interact with each other more explicitly throughout the tracking process. The common characteristics of these works is that the integration is also carried out in the prediction step. For instance, the ICONDENSATION algorithm [18] uses a fixed color model specific to the object of interest to detect blobs in the current frame and uses them in the prediction step of a shape-based particle filter tracker. In [28], the authors suggested an approximate co-inference among the modalities by decoupling the object state and the measurements according to color and shape and by letting each visual cue provide hypotheses for the other one. Thus, in their formulation, the shape samples are drawn according to the color measurements, and the color samples are drawn according to the shape measurements. The tracker in [22], on the other hand, uses a partitioned sampling structure which consists of two layers. The first layer constructed considering either motion or sound provides a coarse information on the target object, which is then refined by the second layer by using color. The work in [9] also suggests a two-level, but more centralized, particle filter architecture. At the lower level, the individual trackers based on different cues perform tracking independently. At the upper level, a fuser integrates the trackers' outputs to construct more reliable hypotheses, and in return provides a feedback to the individual trackers. Although the studies that can be categorized within this latter group introduce explicit interactions between multiple cues, the way these interactions occur in each study is mainly predetermined by the global scheme/architecture considered. Furthermore, the reliabilities of the visual cues are not taken into account in any way. In this respect, the dynamic partitioned sampling approach in [13] is interesting as it proposes to dynamically change the order of cues used in sampling depending on the cue reliabilities.

1.2. Proposed Framework

In this paper, we present a Sequential Monte Carlo based tracking algorithm that combines multi-modal data in an original way. Our main motivation is to develop a tracking algorithm that has the properties of the two groups of works mentioned previously. That is to say, *we suggest to carry out the integration of the multiple cues in both the prediction step and in the measurement step, in estimating the likelihoods.* In [20], Nickel and Stiefelhagen suggested a work in a line similar to ours by combining *Democratic Integration* [25] with two-staged layered sampling [22]. They used a predetermined layer structure with each layer being adaptive in its own. For instance, the first layer is composed of stereo cues each describing a part of the target object. However, compared to theirs, our system architecture allows interactions between multiple cues to be more dynamic and flexible.

For the prediction step, we associate each particle with a specific cue and accordingly with a specific proposal function. The crucial point is that this process is defined as an adaptive process which is governed by the dynamically-changing reliabilities of the visual cues. Thus, if one cue becomes unreliable, the tendency is to reduce the total number of particles associated with it and to increase the total number of particles associated with other visual cue(s). This dynamic process improves the accuracy of the predictions since less reliable proposal functions are utilized less in the sequential importance sampling. During the prediction step no cue is given a preference over another, and the interactions between the cues are directly determined by the current context in an adaptive manner. As mentioned above, we take into account the reliabilities of the visual cues in estimating the confidence measures of the particles as well. We define the overall likelihood function so that the measurements from each cue contribute the overall likelihood according to its reliability. In return, we obtain more precise likelihood values in the measurement step as the misleading effects of the unreliable cues are reduced.

The remainder of the paper is organized as follows: Section 2 recalls the Sequential Monte Carlo method with a focus on multi-modal tracking. Section 3 gives the basis of our object model and the corresponding state dynamics. Section 4 introduces the visual cues and the proposal functions that we consider in our experiments. Section 5 gives the outline of our multi-modal tracking algorithm and our main contributions. Section 6 presents some illustrative tracking experiments in which we analyze the performance of the proposed algorithm. Finally, Section 7 makes a brief summary of our work, and points out the future directions.

2. Sequential Monte Carlo and Multi-modal Tracking

In a classical filtering framework, the main aim is to estimate the posterior distribution $p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$ of the state vector $\mathbf{x}_k$ through a set of measurements $\mathbf{y}_{1:k}$ up to the current time step k. The Bayesian sequential estimation approach computes this distribution according to a two-step recursion: a *prediction* step $p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) d\mathbf{x}_{k-1}$ followed by a *filtering* step $p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \propto p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})$.

This formulation requires two models to be defined: an evolution (transition) model for the state dynamics $p(\mathbf{x}_k \mid \mathbf{x}_{k-1})$ and a likelihood model for the observations $p(\mathbf{y}_k \mid \mathbf{x}_k)$. Sequential Monte Carlo based filtering (*also known as* particle filter) [1, 12, 15, 17] has proved to be an effective method, and provides a simple yet flexible solution to many optimal state estimation problems, such as tracking [8, 16, 27] and sensor fault detection [26].

The main idea behind particle filter is to approximate the posterior distribution $p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$ by a weighted set of N particles $\{\mathbf{x}_k^{(i)}, w_k^{(i)}\}_{i=1}^N$ as $p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \approx \sum_{i=1}^N w_k^{(i)} \delta_{\mathbf{x}_k^{(i)}}(\mathbf{x}_k)$, with $\delta_{\mathbf{x}_0}$ denoting the Dirac delta mass centered on x_0, and each particle representing a possible state $\mathbf{x}_k$ and its weight $w_k^{(i)} \in [0, 1]$ describing its confidence measure.

The recursive estimation is, then, characterized by two main steps: with an approximation of $p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1})$ at hand, new particles are generated from the old particle set $\{\mathbf{x}_{k-1}^{(i)}, w_{k-1}^{(i)}\}_{i=1}^{N}$ by using a known proposal function, $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})$. This prediction step is followed by an update step where the weights of the new particles $w_k^{(i)}$ are determined from the new observations $\mathbf{y}_k$ using $w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)})}{q(\mathbf{x}_k|\mathbf{x}_{0:k-1}^{(i)},\mathbf{y}_{1:k})}$ with $\sum_{i=1}^{N} w_k^{(i)} = 1$. As a further step, a resampling phase, which removes the particles with low weights and accumulates the particles with high weights, can be employed to avoid the degeneracy of the particles [15]. Generally, the final tracking decision is made by taking into account the conditional mean, the weighted average of the particles $\{\mathbf{x}_k^{(i)}\}$, or the particles with the highest weights.

For multi-modal tracking, the simplicity and the flexibility of the particle filter offer a wide variety of solutions. One direction is to perform data fusion in the likelihood estimation step. In this regard, the most straightforward way of integrating multiple measurement sources is to assume that these measurements are conditionally independent given the state and subsequently factorize the overall likelihood as $p(\mathbf{y} \mid \mathbf{x}) = \prod_{m=1}^{M} p(\mathbf{y}^m \mid \mathbf{x})$, with M being the total number of sources. As we stated in the introduction, it is possible to increase the accuracy of the joint likelihood by further considering the reliabilities of the measurement sources in the integration phase [7, 19, 24]. The studies [9, 18, 22, 28] consider another direction and suggest explicit interactions between different modalities. In these works, the main emphasis is on the proposal functions utilized in the prediction step, and how the candidate state hypothesis proposed by different modalities can be integrated.

3. Object Model and State Dynamics

The tracking framework that we propose in this work does not depend on a specific object model, and any model suggested in literature can be utilized. In this paper, we prefer to use a simple model and represent the target object by a fixed reference rectangular region parameterized as $\Omega = (x^c, y^c, w, h)$, where (x^c, y^c) denote the coordinates of the center of the rectangular region having a width w and a height h.

We define the object state as $\mathbf{x}_k = (x_k, y_k, s_k, t_k) \in \mathcal{X}$. It describes a new region $\Omega_{\mathbf{x}_k} = (x_k, y_k, s_k w, t_k h)$ with s_k and t_k denoting the scaling factors for the width and the height of the reference region, respectively.

For the state evolution model, we assume mutually independent Gaussian random walk models along with a small uniform component as in [22]. This uniform component is used to compensate the irregular motion behavior of the target object and provides a kind of re-initialization. Accordingly, the state evolution model can be written as:

$$p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) \sim \beta_U \mathcal{U}(\mathbf{0}, \mathbf{x}_{max}) + (1 - \beta_U)\mathcal{N}(\mathbf{x}_{k-1}, \Lambda) \tag{1}$$

where $\mathcal{U}(0, \mathbf{x}_{max})$ denotes the uniform distribution in $[0, \mathbf{x}_{max}]$, with the vector $\mathbf{x}_{max}$ representing the maximum allowed values over the set $\mathcal{X}$, $\mathcal{N}(\mathbf{x}_{k-1}, \Lambda)$ the Gaussian distribution with mean $\mathbf{x}_{k-1}$ and covariance matrix $\Lambda = \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_s^2, \sigma_t^2)$, and β_U is the weight of the uniform component. The initial state of the object is assumed to be described by a uniform distribution $p(\mathbf{x}_0) = \mathcal{U}(\mathbf{0}, \mathbf{x}_{max})$.

4. Visual Cues and Proposal Functions

This section describes the visual cues that we utilize in tracking an object of interest. These are simply *color*, *motion* and *infrared brightness*, and are discussed in the following subsections.

In our work, while extracting these visual cues from an image frame, we follow a conventional approach and use measurements based on histograms. We compute the likelihoods and construct the individual proposal functions by making use of reference histograms which are defined for each visual cue. We manually construct our reference histograms, and use these histograms throughout the whole tracking sequence without updating them.

Mainly, the construction of the proposal functions and the estimation of the likelihoods depend on the comparison between the histograms extracted from the candidate regions and the reference histogram. For that, we utilize the Bhattacharyya histogram similarity measure [3].

It is important to note that, as in [22], the proposal functions described in the subsequent subsections are defined only for suggesting the new values for the location component of the object state. For the scaling factors, the proposal functions are taken as the corresponding component of the state evolution model described in Equation (1).

4.1. Color Cue

Following [21], we adopt an observation model that is based on Hue-Saturation-Value (HSV) color histograms with $B_C = B_h B_s + B_v$ bins. and define our color likelihood as

$$p(\mathbf{y}^C \mid \mathbf{x}) \propto \exp\left(-\frac{D^2(\mathbf{h}^C_{\mathbf{x}}, \mathbf{h}^C_{ref})}{2\sigma^2_C}\right) \tag{2}$$

with $\mathbf{h}^C_{ref}$ denoting the B_C-bin normalized reference histogram, $\mathbf{h}^C_{\mathbf{x}}$ representing the normalized color histogram which is obtained from a candidate object region specified by the object state $\mathbf{x}$, and $D^2(\mathbf{h}^C_{\mathbf{x}}, \mathbf{h}^C_{ref})$ being the Bhattacharyya histogram similarity measure between them.

The construction of the proposal function also depends on the color likelihood model described above. Typically, we first estimate the color likelihoods on a subset of image locations over the current frame. For this, we use a predefined step size of 5 pixels through the current frame, and keep the scale factors fixed as $s = t = 1$. The likelihoods estimated in this way define an approximate probability distribution map for the target object. Once these likelihoods are estimated, we define our proposal function as follows:

$$\begin{aligned} q^C(x_k, y_k \mid x_{k-1}, y_{k-1}, \mathbf{y}^C_k) &= \beta_{RW}\mathcal{N}\left((x_{k-1}, y_{k-1}), (\sigma^2_x, \sigma^2_y)\right) \\ &+ \frac{(1-\beta_{RW})}{N_C}\sum_{i=1}^{N_C}\mathcal{N}\left(\mathbf{p}^C_i, (\sigma^2_x, \sigma^2_y)\right). \end{aligned} \tag{3}$$

In Equation (3), the first component is the Gaussian random walk component for the object location that we previously introduced in our state evolution model given in Equation (1). The points $\mathbf{p}^C_i = (x_i, y_i)$, $i = 1, \cdots, N_C$ denote the image locations having a likelihood greater than a threshold (i.e. $p(\mathbf{y}^C \mid \mathbf{x}) > \tau^C$), and define the centers of Gaussians in the mixture model utilized in the second component, respectively. We fixed $\beta_{RW} = 0.75$ in our experiments, and thus the main tendency is to preserve the smoothness of the tracking trajectory. On the other hand, the second component allows jumps in the state space to the image regions that likely contain the target object.

4.2. Motion Cue

The image locations having a motion activity at the frame k can be determined from the absolute difference of the intensity images at the frames k and $k-1$. In the frame difference, the pixels with large values indicate the motion activity. If there is no motion, the frame difference is either zero or has a very small value due to the noise and/or due to the slight changes in the intensity.

To estimate the motion likelihood, we follow the approach suggested in [22]. For a region of interest specified by the state $\mathbf{x}$, we associate a motion histogram $\mathbf{h}_{\mathbf{x}}^M = (h_{1,\mathbf{x}}^M, \cdots, h_{B_M,\mathbf{x}}^M)$ with B_M denoting the number of bins. The reference histogram $\mathbf{h}_{ref}^M$ is defined considering a uniform distribution, i.e. $h_{i,ref}^M = 1/B_M, \quad i = 1, \cdots, B_M$. In the case of no motion activity, the Bhattacharyya histogram similarity measure yields $D^2_{no\_mot.} = 1 - \sqrt{1/B_M}$. Considering this, we define the motion likelihood as

$$p(\mathbf{y}^M \mid \mathbf{x}) \propto 1 - \exp\left(- \frac{D^2_{no\_mot.} - D^2(\mathbf{h}_{\mathbf{x}}^M, \mathbf{h}_{ref}^M)}{2\sigma_M^2} \right). \qquad (4)$$

As in Section 4.1, the proposal function is constructed by estimating the likelihoods on a subset of image locations over the current frame. The locations having a likelihood greater than a threshold τ^M are then used, as in [22], to define the proposal function as

$$\begin{aligned} q^M(x_k, y_k \mid x_{k-1}, y_{k-1}, \mathbf{y}_k^M) &= \beta_{RW}\mathcal{N}\left((x_{k-1}, y_{k-1}), (\sigma_x^2, \sigma_y^2)\right) \\ &+ \frac{(1-\beta_{RW})}{N_M}\sum_{i=1}^{N_M}\mathcal{N}\left(\mathbf{p}_i^M, (\sigma_x^2, \sigma_y^2)\right). \end{aligned} \quad (5)$$

4.3. Infrared Brightness Cue

Besides color and motion, we employ infrared brightness cue in some of our experiments. This cue requires the tracking sequence to be imaged from an infrared camera, and allows us to consider different thermal characteristics of an object of interest during tracking. In estimating the likelihoods and constructing the corresponding proposal function, we follow an approach similar to the ones explained in the previous subsections. Then, we define the infrared brightness likelihood as

$$p(\mathbf{y}^I \mid \mathbf{x}) \propto \exp\left(-\frac{D^2(\mathbf{h}_{\mathbf{x}}^I, \mathbf{h}_{ref}^I)}{2\sigma_I^2}\right) \quad (6)$$

where $\mathbf{h}_{ref}^I = (h_{1,ref}^I, \cdots, h_{B_I,ref}^I)$ is the B_I-bin normalized reference histogram, and $\mathbf{h}_{\mathbf{x}}^I = (h_{1,\mathbf{x}}^I, \cdots, h_{B_I,\mathbf{x}}^I)$ is the normalized brightness histogram obtained from the candidate object region. The proposal is as follows:

$$q^I(x_k, y_k \mid x_{k-1}, y_{k-1}, \mathbf{y}_k^I) = \beta_{RW}\mathcal{N}\left((x_{k-1}, y_{k-1}), (\sigma_x^2, \sigma_y^2)\right)$$

$$+ \quad \frac{(1-\beta_{RW})}{N_I}\sum_{i=1}^{N_I}\mathcal{N}\left(\mathbf{p}_i^I,(\sigma_x^2,\sigma_y^2)\right) \qquad (7)$$

where $\mathbf{p}_i^I=(x_i,y_i), i=1,\cdots,N_I$ denote the image locations where the target object is likely to be according to the threshold τ^I.

5. Tracking Algorithm

We propose a novel approach for integrating different visual cues during tracking. Unlike the previous works summarized in Section 1.1, we do not give preference to any cue, or use a global scheme with a predetermined structure. We mainly let the current visual context determine how the interactions between multiple cues are carried out. In all phases of tracking, we emphasize the information derived from the reliable cues and ignore the information provided by the unreliable cues. This view certainly involves discovering and using the reliabilities of the visual cues. We summarize the basic outline of our tracking algorithm in Algorithm 1. As it illustrates, we nearly follow the classic flow of a particle filter-based framework. The proposed tracker consists of *prediction*, *measurement*, *resampling* phases with an additional *reliability-update* step.

5.1. Updating the reliabilities of cues

Adaptive reliabilities assigned to visual cues are key to our formulation. In this paper, we adopt the cue reliability definition of the *Democratic Integration* method [25] and follow the instructions given in Algorithm 2 to adjust them depending on the current context. In the first frame, the cue

Algorithm 1 General algorithm

In the initialization step, $p(\mathbf{x}_0) = \mathcal{U}_{\mathcal{X}}(\mathbf{x}_0)$. Then, from the particle set $\{\mathbf{x}_{k-1}^{(i)}, w_{k-1}^{(i)}\}_{i=1}^{N}$ at the time step $k-1$, determine the new particle set $\{\mathbf{x}_k^{(i)}, w_k^{(i)}\}_{i=1}^{N}$ as follows:

1. **Adjust** cue reliabilities $\{r_k^\ell\}$ considering current observations $\mathbf{y}_k$ (Algorithm 2).
2. **Generate** new hypotheses $\{\mathbf{x}_k^{(i)}\}_{i=1}^{N}$ through a prediction step (Algorithm 3).
3. **Update** weights of the particles $\{w_k^{(i)}\}_{i=1}^{N}$ (Equation 13).
4. **Estimate** the conditional mean as the solution (Equation 14) and perform resampling for the next time step.

reliabilities are initialized with equal weights with their sum equal to 1. In the subsequent frames, each reliability value is dynamically updated by using Equation (11). The new reliability value of a cue is determined by considering both the overall success of that cue in the past, which corresponds to the old reliability value, and its individual success in predicting the current joint result, which corresponds to its quality (Equation (10)). The quality of a cue simply quantifies the degree of agreement between the joint result and the result the cue individually suggests. Thus, the reliabilities can be interpreted as the qualities smoothed over time. Each quality measure compares the importance of a cue at an approximate target position $\hat{\mathbf{x}}_k$ determined by Equation (8) with its response averaged over the corresponding approximate cue likelihood. Then, a cue having a quality higher than its current reliability will be given a higher influence in the future by increasing its reliability. In a similar manner, a cue having a quality lower than its current reliability will be suppressed by decreasing its reliability.

Note that since the initial reliabilities and the quality values are normalized, the reliabilities are also normalized and their sum is always one.

Algorithm 2 Updating the reliabilities of the visual cues

- **Approximate** target position $\hat{\mathbf{x}}_k$ using previous reliabilities and current observations:

$$\hat{\mathbf{x}}_k = \arg\max_x \left(\hat{p}(\mathbf{y}_k \mid \mathbf{x})\right) = \arg\max_x \left(\prod_{\ell \in \{C,I,M\}} \hat{p}(\mathbf{y}_k^\ell \mid \mathbf{x})^{r_{k-1}^\ell} \right) \tag{8}$$

  with $\hat{p}(\mathbf{y}_k^\ell \mid \mathbf{x})$ the approximate probability distribution map estimated for the modality ℓ

- **Estimate** the quality measures for each cue as follows:

$$\bar{s}_k^\ell = \begin{cases} 0 & \textbf{if } \hat{p}(\mathbf{y}_k^\ell \mid \hat{\mathbf{x}}_k) \leq \langle \hat{p}(\mathbf{y}_k^\ell \mid \mathbf{x}) \rangle \\ \hat{p}(\mathbf{y}_k^\ell \mid \hat{\mathbf{x}}_k) - \langle \hat{p}(\mathbf{y}_k^\ell \mid \mathbf{x}) \rangle & \textbf{if } \hat{p}(\mathbf{y}_k^\ell \mid \hat{\mathbf{x}}_k) > \langle \hat{p}(\mathbf{y}_k^\ell \mid \mathbf{x}) \rangle \end{cases} \tag{9}$$

  where $\langle \cdots \rangle$ denotes the average over the approximate probability distribution map

- **Determine** the normalized qualities s_k^ℓ:

$$s_k^\ell = \frac{\bar{s}_k^\ell}{\sum_j \bar{s}_k^j} \tag{10}$$

- **Update** reliabilities considering the current quality measures as follows:

$$r_k^\ell = r_{k-1}^\ell + \eta(s_k^\ell - r_{k-1}^\ell) \tag{11}$$

  with η denoting a time constant which we set to 0.1 in our experiments.

Moreover, the cue reliabilities are defined through quality values which are defined over the whole image domain. By this way, the reliabilities are determined by considering a global picture of the tracking scene, and thus the tracking inaccuracies do not affect the reliability computations.

5.2. Predicting the new locations of particles

Once the updated cue reliabilities are determined, they are used to guide the hypothesis generation phase, providing premises regarding the new locations of particles. This process is summarized in Algorithm 3. As can be seen, in our framework, each particle is assigned to a modality denoted by ℓ

Algorithm 3 Generating the new hypotheses through prediction

```
• Simulate $\ell_k^{(i)}$:
    – Generate a random number $\alpha \in [0,1)$, uniformly distributed.
    – Set $\ell_k^{(i)} = \begin{cases} C & \textbf{if} \quad \alpha < r_k^C \\ I & \textbf{if} \quad r_k^C \leq \alpha < r_k^C + r_k^I \\ M & \textbf{if} \quad \alpha \geq r_k^C + r_k^I \end{cases}$    (12)
• Simulate $\mathbf{x}_k^{(i)} \sim q^{\ell_k^{(i)}}(\mathbf{x}_k \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{y}_k^{\ell_k^{(i)}})$
```

with $\ell \in \{C, I, M\}$ (C for color, I for infrared brightness, M for motion) and accordingly to a specific proposal function $q^{\ell_k}(\mathbf{x}_k \mid \mathbf{x}_{k-1}, \mathbf{y}_k^{\ell_k})$ (Equation (12)). This process performs sampling from a mixture model, relying on the principle of generation of non-uniformly random samples [5]. As the reliabilities determine the assignments, if one cue becomes unreliable relative to other visual cues, the tendency is to reduce the total number of particles associated with it and to increase the total number of particles associated with more reliable visual cue(s). As a result, the tracking accuracy increases as less reliable proposal functions are utilized less in the sequential importance sampling in predicting the position of the target object.

For example, consider a video sequence where all the cues equivalently give questionable observations for some of the tracking frames (e.g. during the time the target object gets completely occluded and becomes visible again). In the suggested scheme, the recovery of the lost target object can be carried out quickly since the reliabilities can quickly adapt themselves to the current context using the information acquired from the whole image, and the tracker can accordingly utilize the proposals which give more accurate predictions than the unreliable proposals.

5.3. Updating the weights of particles and estimating the joint result

The next step of our algorithm includes a measurement step which adjusts the weights of new particles according to new observations. This is performed by using the formula:

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k \mid \mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)} \mid \mathbf{x}_{k-1}^{(i)})}{q^{\ell_k^{(i)}}(\mathbf{x}_k^{(i)} \mid \mathbf{x}_{k-1}^{(i)}, \mathbf{y}_k^{\ell_k^{(i)}})} \quad \text{with} \quad \sum_{i=1}^{N} w_k^{(i)} = 1 \; . \tag{13}$$

The key point is that the updated cue reliabilities play central roles here as well. The overall likelihood function $p(\mathbf{y}_k \mid \mathbf{x}_k)$ is defined in a way that the cue likelihoods are integrated in an adaptive manner as follows:

$$p(\mathbf{y}_k \mid \mathbf{x}_k) = \prod_{\ell \in \{C,I,M\}} p(\mathbf{y}_k^{\ell} \mid \mathbf{x}_k)^{r_k^{\ell}} \tag{14}$$

with $\sum_{\ell \in \{C,I,M\}} r^{\ell} = 1$. As a result, each cue contributes to the joint tracking result according to its current reliability, and the ones having low values have little effect on the outcome. The individual likelihoods having a value estimated as zero make the overall likelihood zero as we take the product, whether its reliability score is low or not. Thus, in our experiments, we adjust all such likelihoods values and explicitly set them to a small value like $p(\mathbf{y}^{\ell} \mid \mathbf{x}) = 0.001$.

Finally, the decision about the tracking process for the current time step k is obtained from the particle set by estimating the weighted average of the hypothesized states:

$$\widehat{\mathbf{x}_k} = \sum_{i=1}^{N} w_k^{(i)} \mathbf{x}_k^{(i)} \ . \qquad (15)$$

5.4. Implementation details

We have implemented the proposed algorithm in MATLAB on a PC with a 3.16 GHz Intel Core2 Duo processor. In all the experiments, we fixed $\sigma_x = \sigma_y = 3$, $\sigma_s = \sigma_t = 0.01$, $\beta_U = 0.01$, $\sigma_C = 0.2$, $\sigma_M = 0.4$, $\sigma_I = 0.25$, $B_h = B_s = B_v = 10$, $B_M = 20$, $B_I = 30$, and used detection thresholds $\tau^C = \tau^I = 0.65$, $\tau^M = 0.2$. In Equations (3), (5) and (7), if respectively N_C, N_I or N_M equals to zero, we use only the first Gaussian random walk component for the related proposal function.

Among these parameters, the most critical ones are the detection thresholds τ^C, τ^M, and τ^I which are used to construct the proposal functions. As the experimental analysis performed in the next section indicates, the proposed work is robust in terms of false positives given the current context with respect to the values chosen for these parameters, and it generally provides better results than those of other cue integration strategies.

As for the computational cost, the main bottleneck of the suggested approach is the construction of the approximate probability distribution maps, which is carried out for each cue at each frame. The important factor here is the value of the pre-defined step size which defines the subset of image locations over the current frame where the likelihoods are estimated. For a video sequence containing 144×192 color image frames, our tracker runs at approximately 2 frames per second with a step size of 5 pixels being used. It should be added that the run-time performance could be further improved by including some MEX C++ subroutines, or parallelizing the code.

6. Experimental Results

In this section, we demonstrate the performance of the proposed framework (Algorithm 3) on illustrative video sequences. We performed two groups of experiments. The first set is mainly about the qualitative analysis of the proposed method in which we consider different tracking scenarios. Following that, in the second set of experiments, we carry out a thorough quantitative analysis in terms of tracking accuracy by using some sequences in which the ground truth is available.

We typically compare our results obtained considering multiple cues with context-sensitive reliabilities with those obtained using a single cue or multiple cues with fixed reliabilities. We also provide the tracking outcomes of the two-layered partitioned sampling (PS) and the dynamic partitioned sampling approaches (DPS), because these approaches are known to be robust and well known for the tracking based on multiple cues. Our implementation of these methods follows the architecture suggested in [22] – in the first level, the object locations are sampled from the proposal functions introduced in Sec. 4 and in the second level, the state evolution model described in Sec. 3 is used for the scaling factors with a resampling phase in between. While the order of cues is fixed for the PS [22] (from motion to color), for the DPS, following the idea suggested in [13], we change the order of cues dynamically depending on the cue reliabilities.

In our experiments, we use a fairly small number of particles, $N = 100$. The reference color models are manually constructed in the first frame of the sequences. For qualitative analysis, we employ the conditional mean and the particles with the five highest weights to depict the outcomes. We associate different colors for the particles, and the rectangular regions they represent, depending on the cue they are attached to: *green* for color, *blue* for motion, and *red* for infrared brightness. Additionally, we draw the rectangle represented by the conditional mean in *white*. This color distribution among the particles does visually represent the cue reliabilities. In the second set of experiments, we present the results by using only the corresponding conditional means. The videos showing the results of these experiments are provided as supplementary material.

6.1. Qualitative Analysis

We first consider a sequence from the BEHAVE Interactions Test Case Scenarios [6] where we try to track a person with a white shirt using color and motion information. Throughout the sequence, first, a group of people goes after the person of interest and attacks him. During this time, he is completely occluded. Next, at some point, the person of interest kneels down and stops moving. These different phenomena observed throughout the video sequence exemplifies the contextual changes that we exploit in our tracking framework.

As Figures 1.(a) and 1.(b) respectively demonstrate, the color-based tracking and the motion-based tracking may lead to inaccurate results due to the ambiguities inherent to the processing of the video sequence considering single modalities. There are objects in the background which have similar appearances to the object of interest. Therefore, soon after the initialization, the framework based on color starts tracking the wrong object and remains at this local minimum point during nearly half of the video sequence. However, it is eventually able to recover the actual object of interest with the utility of the color-based proposal. The outcomes of the motion-based tracker is much worse since the video sequence involves several persons in motion. That is, the motion likelihood function becomes non-discriminative with respect to the target object and the samples are distributed all around the moving objects. As one expects, considering color and motion cues all together with fixed values for reliabilities gives better tracking results than using only one modality (Figure 1.(c)). Yet, such a scheme has some drawbacks. Since equal weights are given for color and motion cues, if one of the sources becomes unreliable, it directly affects the results. In the video sequence, the person entering the scene during which the actual person of interest is at rest distracts tracking.

As illustrated in Figure 1.(d), considering a scheme with context-sensitive reliabilities eliminates most of the ambiguities mentioned and results in an improvement in the outcomes. For instance, when the target person is occluded by the group of people following him, the reliability of the color cue decreases, and thus the motion cue particularly guides the tracking process during this time interval. Similarly, when the person of interest becomes idle, the reliability of motion decreases, making the color cue the dominant cue.

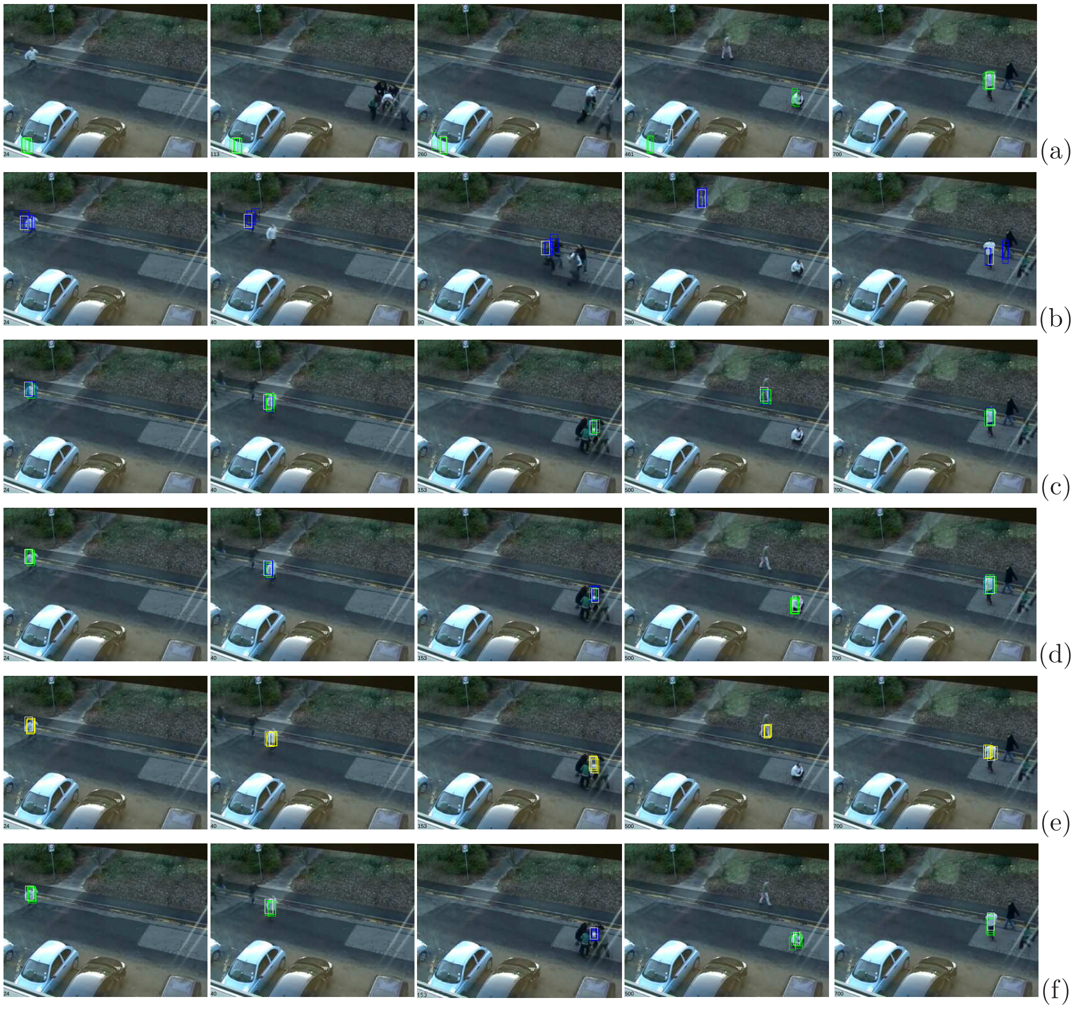
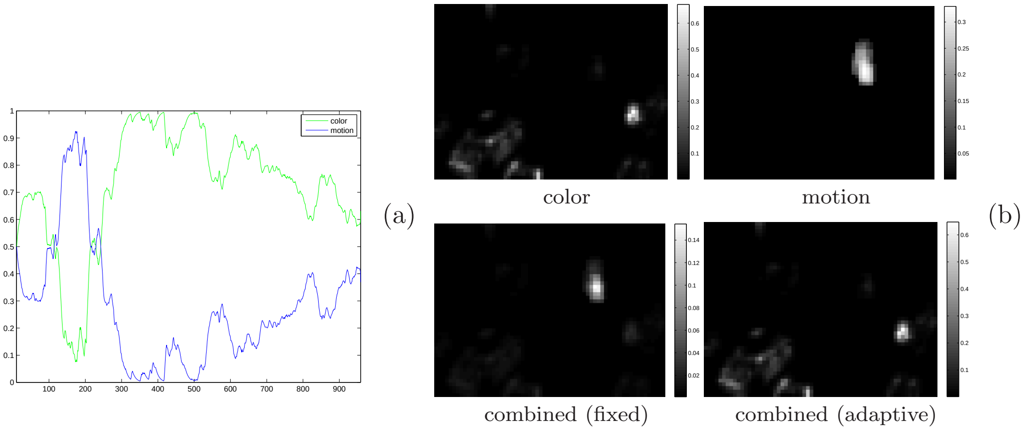


Figure 1: **seq. 1** Sample tracking results using: (a) Color. (b) Motion. (c) Both color and motion with fixed reliabilities. (d) Both color and motion with context-sensitive reliabilities. Modifying the reliabilities of the visual cues according to the context and accordingly using them eliminate most of the ambiguities that the previous cases cannot easily cope with. (e) PS. (f) DPS.

Thus, the tracking process does not get distracted by the person entering the scene unlike in the case with fixed reliabilities. Figure 2.(a) illustrates these changes in the reliabilities of the cues. In Figure 2.(b), we provide color and



color

motion

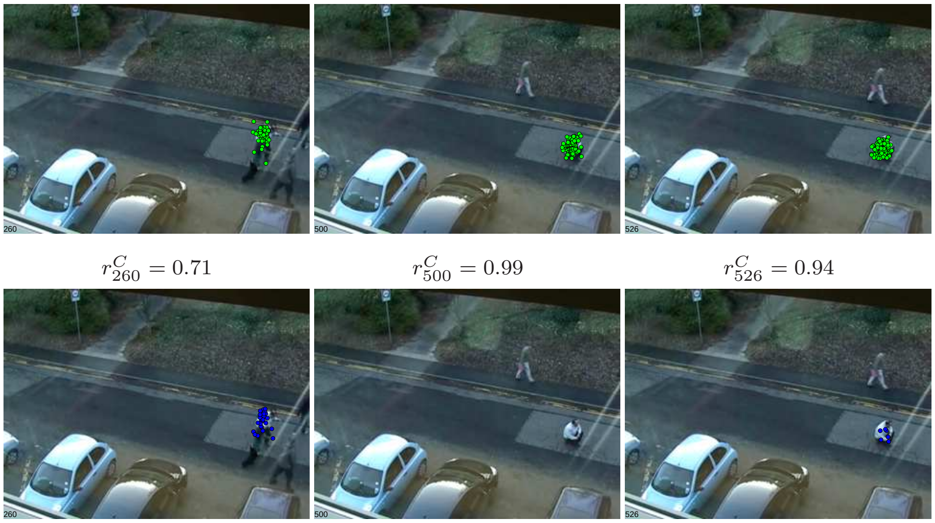
combined (fixed)

combined (adaptive)

Figure 2: (a) Reliabilities throughout **seq. 1**. (b) Likelihoods for a sample frame. A more accurate estimate is achieved using adaptive weights for the reliabilities.

motion likelihoods as well as their combinations with two different strategies for the frame where the person of interest is at rest. As mentioned at the beginning of this section, our color encoding scheme can be used to visually represent the cue reliabilities through the distribution of the colored samples. In Figure 3, we provide such a representation for three sample frames.

In Figure 1.(e), we demonstrate the disadvantage of using PS that results in inaccurate tracking. The tracking process relies primarily on the motion information in the prediction step, and thus the person entering the scene during the time the actual person of interest is at rest distracts the tracking process as in the case with fixed reliabilities (Figure 1.(c)). Since this approach does not attach the particles to any particular modality, we use a different color (yellow) for the particles representing the tracking outcomes. The tracker based on DPS, on the other hand, successfully tracks the target like ours as the order of cues in the partitioned sampling is updated accord-



$r^C_{260} = 0.71$ $r^C_{500} = 0.99$ $r^C_{526} = 0.94$

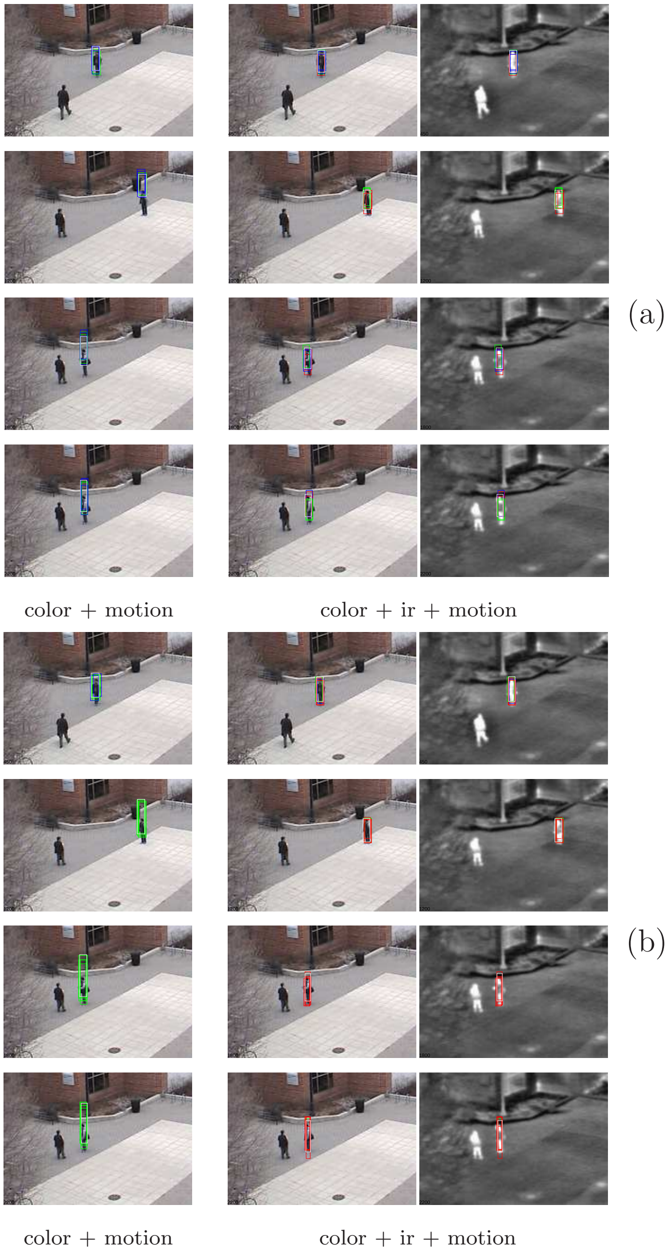
$r^M_{260} = 0.3$ $r^M_{500} = 0.01$ $r^M_{526} = 0.06$

Figure 3: **seq. 1** Visual representation of the cue reliabilities at three sample frames (*green* for color and *blue* for motion).

ing to the cue reliabilities (Figure 1.(f)). Note that increasing the value of τ^M to a convenient value makes both the framework that uses fixed reliabilities for color and motion, and PS approach accurately track the person of interest. This highlights that our proposed work is more robust against the values chosen for the detection parameters in terms of false positives given the current context.

In the second experiment, we consider a tracking sequence captured from an infrared camera along with a CCD camera (taken from the OSU Color-Thermal Database [11]). We test our framework under four scenarios. The first set of experiments involves employing fixed reliabilities, and considering color and motion cues together and additionally using infrared brightness along with them. The second set of experiments uses the same two different cue combinations, but with adaptive reliabilities for the cues. We show the results of these experiments in Figure 4. In each figure, we provide the outcomes based on color and motion, and color, motion and infrared brightness side by side. It can be seen from these figures that the results of the framework built upon color and motion are not good, whether fixed values for the reliabilities are used or not. These cues both fail to account for the uncertainties in the tracking sequence. Specifically, the reference color model quickly becomes inadequate for describing the appearance of the person of interest, leading to enlarged and inaccurate object regions. This is mainly due to the changes in the person's view throughout the sequence and the nearby objects with a similar color. The problem with the motion cue is more severe since the sequence contains another person walking in the scene, and more importantly, the person of interest does not move much most of the time.

Introducing infrared brightness as a complementary cue, in this respect, improves the performance and provides more accurate tracking. It is important to note that most of the time, refining the reliabilities with respect to the contextual information gives more accurate results than using fixed values for the reliabilities. As illustrated in Figure 5, with adaptive reliabilities, the motion cue remains being the least reliable cue throughout the sequence due to the aforementioned points. Infrared brightness and color cues competes with each other to describe the person of interest, and since infrared brightness values do not change much when the tracked person changes its pose, the infrared brightness cue is given a higher weight or importance than



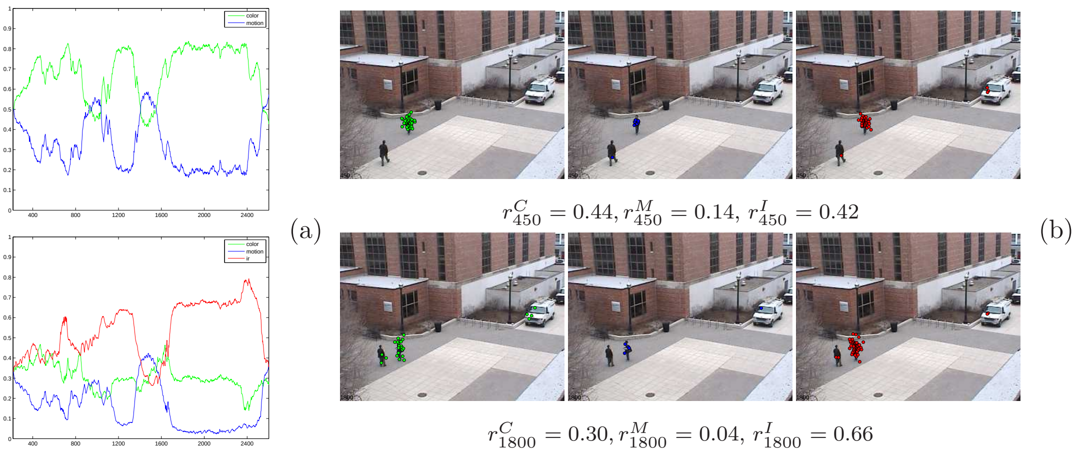
color + motion

color + ir + motion

color + motion

color + ir + motion

Figure 4: **seq. 2** Sample tracking results. (a) With fixed reliabilities. (b) With adaptive reliabilities. It results in more accurate tracking of the person of interest for the framework in which infrared brightness is introduced as a complementary cue. Infrared brightness cue is more reliable and is given a higher importance than the other cues during tracking.



$r^C_{450} = 0.44, r^M_{450} = 0.14, r^I_{450} = 0.42$

$r^C_{1800} = 0.30, r^M_{1800} = 0.04, r^I_{1800} = 0.66$

Figure 5: (a) Reliabilities throughout **seq. 2**. (b) Visual representation of the cue reliabilities at two sample frames (*green* for color, *blue* for motion and *red* for infrared brightness).

the color cue most of the time. This results in a significant change for the reliability values of color (cf. the plots in Figure 5.(a)).

Lastly, we consider the image sequence `OneShopOneWait2cor` from the CAVIAR project [14]. We again compare the tracking outcomes obtained by using single visual cues, color and motion, with that of obtained by combining these two. As illustrated in Figure 6.(b), using motion data alone leads to inaccurate tracking. The sequence contains several persons moving across the hallway. The tracking process cannot distinguish the actual person of interest from the others, and the particles are distributed all over the moving persons. On the other hand, the color-based tracking and our framework provide nearly similar tracking results (Figures 6.(a) and 6.(c)). They succeed in tracking the object for most part of the sequence, but they lose the track whenever a person having a similar appearance enters the scene.

The reason behind the similar performance is that with respect to the contextual information, color is determined to be the main cue and is given a much higher weight than motion during tracking (Figure 7). This experiment shows that combining several visual cues does not always mean robustness. It improves the tracking results only when at least one of the cues considered in tracking is effective in describing the target object. For instance, in this example, color and motion both fail to account for the uncertainties. The PS approach produces much worse results since it uses a fixed order in the sampling, from motion to color. As shown in Figure 6.(d), the tracker tracks four different persons throughout the sequence. The flexibility of the DPS approach, due to the order of visual cues changing dynamically in accordance with their reliabilities, mostly eliminates these false detections and tracking as illustrated in Figure 6.(e).

6.2. Quantitative Analysis

In this section, we quantitatively evaluate our tracking algorithm on two sets of video sequences. The first set involves the sequence from the BEHAVE Interactions Test Case Scenarios [6] that we previously presented in Section 6.1 and that consists of 949 frames. In the second set of sequences, we use several video sequences from the CAVIAR project [14]. All these video sequences exhibit a wide variety of challenges including changes in the pose and scale of the target object, varying illumination conditions, and partial occlusions. We tested the trackers by running them 5 times and by taking the average for each video sequence since they are all particle-filter based

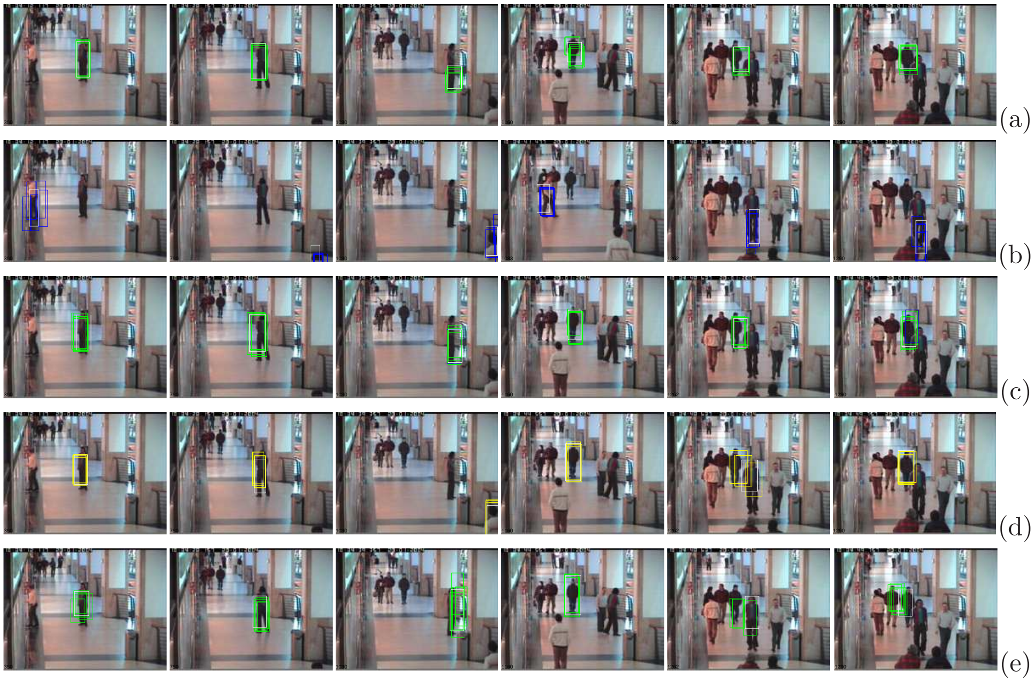


Figure 6: **seq. 3** Sample tracking results using:(a) Color. (b) Motion. (c) Both color and motion with context-sensitive reliabilities. The proposed tracking framework succeeds in tracking the person of interest until a person with a similar appearance appears in the video sequence. (d) PS. (e) DPS.

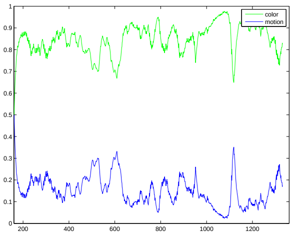


Figure 7: Reliabilities throughout **seq. 3**.

formulations and thus involve some randomness.

For quantitative analysis, we use two measures. We compute the average F-measures, given by $F = \frac{2pr}{p+r}$ where p is the precision $p = \frac{|\mathcal{E} \cap \mathcal{GT}|}{|\mathcal{E}|}$ and r the recall $r = \frac{|\mathcal{E} \cap \mathcal{GT}|}{|\mathcal{GT}|}$ with $\mathcal{E}$ the rectangular region estimated by the conditional

Tracker	F-measure	Success rate
Color	0.32 ± 0.18	70.58 ± 40.42
Motion	0.15 ± 0.04	46.35 ± 10.07
Fixed reliabilities	0.46 ± 0.07	94.42 ± 7.11
Proposed method	0.46 ± 0.03	**99.57 ± 0.17**
DPS	0.46 ± 0.02	98.53 ± 0.84
PS	0.39 ± 0.03	86.78 ± 1.77

Table 1: Average F-measures and success rates (percentage of frames in which the target object is successfully tracked) for the sequence from the BEHAVE dataset.

mean and $\mathcal{GT}$ the ground truth, and the percentage of frames where the target object was successfully tracked among the frames for which the ground truth is available. The tracking is considered to be successful if $\mathcal{E}$ overlaps with $\mathcal{GT}$.

Table 1 provides the quantitative tracking results for the sequence from the BEHAVE dataset, obtained by comparing the trackers' outcomes to the manually labeled ground truth data. As it can be seen, the outcomes are in line with the qualitative results presented before. The trackers based on single cues have the worst performances; and due to the ambiguities inherent to these cues the standard deviations of the measures are higher than those of others. In general, the proposed method and the dynamic partitioned sampling approach are competitive and give better results than the others.

We have performed the second set of our experiments on nine different video sequences from the CAVIAR project [14]. We use three of these sequences twice; in each we track two different persons, respectively. This makes twelve experiments in total. Table 2 shows the summary of these ex-

Sequence	ObjectId	Total # of Frames
`OneLeaveShop2cor`	0	546
`OneShopOneWait1cor`	2	734
`OneShopOneWait2cor`	7	1171
`OneStopEnter1cor`	1	581
`OneStopEnter1cor`	2	1324
`OneStopEnter2cor`	3	316
`OneStopEnter2cor`	4	834
`OneStopMoveEnter1cor`	7	664
`OneStopMoveNoEnter2cor`	0	639
`OneStopNoEnter1cor`	0	395
`ThreePastShop2cor`	2	331
`ThreePastShop2cor`	7	459

Table 2: The sequences from the CAVIAR project used in the experiments.

periments. The sequences used in the experiments involve different scenarios with varying complexities (changes in the appearance due to pose and illumination variations, occlusions of the target, crowdedness in the background, etc.).

Table 3 and 4 summarize the quantitative performance of the tested tracking methods[2]. It can be seen from these results that in general the proposed tracker outperforms the other trackers. Mostly, it gives either the best or the second best results with respect to the manually labeled ground truth. In terms of the quantitative measures averaged over all experiments (Table 3), it has the best F-measure and success rate performances and the smallest average rank. Among the trackers that fuse multiple cues, the PS approach [22]

[2] The qualitative comparisons (videos showing the results of these experiments) can be downloaded following the url `http://perso.telecom-paristech.fr/~bloch/PR-Submission`

Method	Avg. F-measure	Avg. Succ. Rate	Avg. Rank
Color	0.49	86.13	2.25
Motion	0.24	52.00	4.50
Proposed	**0.53**	**89.07**	**1.83**
DPS	0.50	85.16	2.50
PS	0.40	67.40	3.58

Table 3: Quantitative results averaged over all experiments from the CAVIAR project.

provides the worst performance. The reason for this mainly stems from the fixed order (from motion to color) that is used in [22] in the sampling. It can be also observed that for nearly half of the experiments the color-based tracker perform especially well. Since our method adaptively estimates the reliabilities of color and motion cues with respect to the contextual information (color is given a much higher weight than motion during tracking) and uses them both in the prediction and the likelihood estimation steps, our performance is competitive to the color-based tracker in these sequences. From all these experiments, we can conclude that for situations where fusion is actually useful, our method outperforms the other methods.

7. Summary and Future Work

We have presented a particle filter-based tracking algorithm which integrates multiple cues in a novel way. Unlike previous approaches, our method performs the multi-cue integration both in making predictions about the object of interest and in verifying them through observations. Both stages of the integration depend on the reliabilities of the visual cues, which are adapted in a dynamic way. Particularly, in the prediction step, the reliabilities determine to which cue and the proposal function the particles are attached, forcing reliable proposal functions to be employed more in the sequential importance sampling. Moreover, in the measurement step, they specify the level of contribution of each visual cue to the compound likelihood, resulting in more precise weights for the particles.

We have demonstrated the potential of the proposed approach on various illustrative video sequences with different tracking scenarios. As the experimental results reveal, dynamic structure of our formulation makes tracking process easily adapt itself to changes in the context. The proposed framework is general enough to easily include other sources of information. Even though in our experiments we use color, motion and infrared brightness cues as the main sources of information for tracking an object, we can extend this list with further visual cues (such as feature spatial cue or histogram of gradients) and integrate them in our framework without any difficulty. The conditional independence of observations should then be reconsidered, depending on the chosen cues. Moreover, the suggested approach allows introducing new modalities, whenever available, throughout tracking. However, it is important to note that combining several visual cues does not always increase the tracking accuracy as our last experiment illustrates. Intuitively, integrating various visual cues does improve the outcomes by eliminating the ambiguities only when at least one of the cues considered in tracking is effective in describing the object of interest.

In updating the reliabilities of the visual cues, we adopt the approach suggested in [25]. As a future work, it could be interesting to develop new quality measures in updating the cues' reliabilities. For example, in a recent work [27], the dynamics parameters in the particle filter are estimated via a fuzzy model. Considering fuzzy measures instead of the hard decision utilized in [25] may result in more accurate estimation of cue reliabilities. Moreover, in our formulation, we fixed the weight for the state dynamics in the proposals $\beta_{RW} = 0.75$ for all cues in tracking the target object. In the case where all the visual cues suggest likely target points (i.e., N_C, N_I and N_M all > 0), the overall filter proposal can be interpreted as a mixture containing four different proposals (one including the state dynamics with weight and one for each cue). An interesting future work could be defining the weight of the state dynamics in the mixture in an adaptive way instead of fixing it to a specific value β_{RW}. Of course, this requires defining a reliability score for this component as well. For this purpose the Democratic Integration is not suitable, and a new approach should be devised.

References

[1] Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002. A tutorial on particle filters for online nonlinear/non-Gaussian bayesian tracking. IEEE Transactions on Signal Processing 50 (2), 174–188.

[2] Badrinarayanan, V., Perez, P., Clerc, F. L., Oisel, L., 2007. Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues. In: Proc. Int. Conf. Computer Vision. pp. 1–8.

[3] Bhattacharyya, A., 1943. On a measure of divergence between two sta-

tistical populations defined by their probability distributions. Bull. Calcutta Math. Soc. 35, 99–109.

[4] Birchfield, S., 1998. Elliptical head tracking using intensity gradients and color histograms. In: Proc. Int. Conf. Computer Vision. pp. 232–237.

[5] Bishop, C. M., 2007. Pattern Recognition and Machine Learning (Information Science and Statistics), 1st Edition. Springer.

[6] Blunsden, S., Fisher, R. B., 2007. BEHAVE interactions test case scenarios. `http://groups.inf.ed.ac.uk/vision/BEHAVEDATA/INTERACTIONS`.

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

[8] Chang, W., Chen, C., Hung, Y., 2009. Tracking by parts: A Bayesian approach with component collaboration. IEEE Trans. Systems, Man, and Cybernetics, Part B 39 (2), 375–388.

[9] Chen, Y., Rui, Y., 2004. Real-time speaker tracking using particle filter sensor fusion. Proceedings of the IEEE 92 (3), 485–494.

[10] Collins, R. T., Liu, Y., Leordeanu, M., 2005. Online selection of discriminative tracking features. IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (10), 1631–1643.

[11] Davis, J., Sharma, V., 2007. Background-subtraction using contour-based fusion of thermal and visible imagery. Comput. Vis. Image Understand. 106 (2-3), 162–182.

[12] Doucet, A., Johansen, A. M., 2008. A tutorial on particle filtering and smoothing: Fifteen years later. Tech. rep., Department of Statistics, University of British Columbia.

[13] Duffner, S., Odobez, J.-M., Ricci, E., 2009. Dynamic partitioned sampling for tracking with discriminative features. In: Proc. British Machine Vision Conf.

[14] Fisher, R., October 2004. CAVIAR Test Case Scenarios. Online Book.

[15] Gordon, N., Salmond, D., Smith, A., 1993. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proc. F 140 (2), 107–113.

[16] Hu, W., Tan, T., Wang, L., Maybank, S., 2004. A survey on visual surveillance of object motion and behaviors. IEEE Trans. Systems, Man, and Cybernetics, Part C 34 (3), 334–352.

[17] Isard, M., Blake, A., 1998. CONDENSATION-conditional density propagation for visual tracking. Int. J. Comput. Vis. 29 (1), 5–28.

[18] Isard, M., Blake, A., 1998. ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework. In: Proc. Eur. Conf. Computer Vision. pp. 893–908.

[19] Maggio, E., Smeraldi, F., Cavallaro, A., 2005. Combining colour and orientation for adaptive particle filter-based tracking. In: Proc. British Machine Vision Conf. pp. 659–668.

[20] Nickel, K., Stiefelhagen, R., 2008. Dynamic integration of generalized cues for person tracking. In: Proc. Eur. Conf. Computer Vision. pp. 514–526.

[21] Pérez, P., Hue, C., Vermaak, J., Gangnet, M., 2002. Color-based probabilistic tracking. In: Proc. Eur. Conf. Computer Vision. pp. 661–675.

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

[23] Shen, C., van den Hengel, A., Dick, A., 2003. Probabilistic multiple cue integration for particle filter based tracking. In: Proc. Int. Conf. Digital Image Computing: Techniques and Applications. pp. 399–408.

[24] Spengler, M., Schiele, B., 2003. Towards robust multi-cue integration for visual tracking. Mach. Vision Appl. 14 (1), 50–58.

[25] Triesch, J., von der Malsburg, C., 2001. Democratic integration: Self-organized integration of adaptive cues. Neural Computation 13 (9), 2049–2074.

[26] Wei, T., Huang, Y., Chen, C. L. P., 2009. Adaptive sensor fault detection and identification using particle filter algorithms. IEEE Trans. Systems, Man, and Cybernetics, Part C 39 (1), 201–213.

[27] Widynski, N., Dubuisson, S., Bloch, I., 2011. Integration of fuzzy spatial information in tracking based on particle filtering. IEEE Trans. Systems, Man, and Cybernetics, Part B 41 (3), 635–649.

[28] Wu, Y., Huang, T. S., 2001. A co-inference approach to robust visual tracking. In: Proc. Int. Conf. Computer Vision. pp. 26–33.

Table 4: Individual average F-measures and success rates. Algorithms compared are: color, motion, proposed (color and motion with context-sensitive reliabilities), DPS, and PS. The best and the second best performances are indicated in red and blue, respectively.

Sequence(ObjectId)		Color	Motion	Proposed	DPS	PS
`OneLeaveShop2cor(0)`	succ.	79.30 ± 44.24	71.85 ± 6.23	**95.05 ± 1.29**	**93.94 ± 0.80**	52.15 ± 32.07
	F-meas.	0.47 ± 0.26	0.41 ± 0.04	0.59 ± 0.02	0.57 ± 0.02	0.33 ± 0.21
`OneShopOneWait1cor(2)`	succ.	**100.00 ± 0.00**	67.29 ± 3.59	**99.95 ± 0.07**	99.70 ± 0.67	85.81 ± 3.14
	F-meas.	0.65 ± 0.01	0.16 ± 0.02	0.62 ± 0.02	0.53 ± 0.02	0.45 ± 0.02
`OneShopOneWait2cor(7)`	succ.	**96.59 ± 3.15**	74.61 ± 0.63	**96.18 ± 5.24**	95.09 ± 0.05	87.16 ± 4.15
	F-meas.	0.58 ± 0.02	0.43 ± 0.07	0.58 ± 0.02	0.60 ± 0.01	0.57 ± 0.02
`OneStopEnter1cor(1)`	succ.	81.00 ± 42.20	91.69 ± 6.00	**98.72 ± 1.15**	93.97 ± 12.82	**99.62 ± 0.39**
	F-meas.	0.47 ± 0.25	0.44 ± 0.04	0.62 ± 0.02	0.62 ± 0.08	0.64 ± 0.03
`OneStopEnter1cor(2)`	succ.	74.61 ± 0.54	49.03 ± 13.15	**96.11 ± 7.99**	**80.88 ± 17.84**	60.31 ± 15.82
	F-meas.	0.50 ± 0.01	0.25 ± 0.07	0.61 ± 0.06	0.49 ± 0.06	0.43 ± 0.10
`OneStopEnter2cor(3)`	succ.	62.92 ± 50.88	99.43 ± 0.27	**99.49 ± 0.28**	**99.75 ± 0.14**	**99.75 ± 0.14**
	F-meas.	0.25 ± 0.20	0.45 ± 0.03	0.42 ± 0.02	0.41 ± 0.03	0.40 ± 0.02
`OneStopEnter2cor(4)`	succ.	**99.52 ± 0.42**	60.72 ± 1.82	**65.19 ± 1.22**	57.84 ± 15.45	57.72 ± 14.68
	F-meas.	0.59 ± 0.02	0.34 ± 0.02	0.41 ± 0.01	0.37 ± 0.10	0.36 ± 0.10
`OneStopMoveEnter1cor(7)`	succ.	**69.11 ± 1.36**	21.84 ± 4.94	**70.23 ± 1.61**	58.64 ± 18.22	44.86 ± 21.31
	F-meas.	0.50 ± 0.01	0.03 ± 0.01	0.47 ± 0.03	0.38 ± 0.15	0.28 ± 0.15
`OneStopMoveNoEnter2cor(0)`	success rate	**99.84 ± 0.11**	45.20 ± 9.34	95.61 ± 9.11	**95.99 ± 8.10**	79.28 ± 0.45
	F-measure	0.56 ± 0.02	0.23 ± 0.06	0.58 ± 0.05	0.61 ± 0.03	0.52 ± 0.03
`OneStopNoEnter1cor(0)`	success rate	**97.41 ± 2.44**	0.00 ± 0.00	**54.26 ± 50.23**	47.82 ± 45.31	15.33 ± 8.97
	F-measure	0.49 ± 0.02	0.00 ± 0.00	0.30 ± 0.28	0.29 ± 0.27	0.09 ± 0.05
`ThreePastShop2cor(2)`	success rate	**99.58 ± 0.17**	2.12 ± 4.74	99.15 ± 0.72	**99.45 ± 0.40**	35.03 ± 16.09
	F-measure	0.49 ± 0.01	0.00 ± 0.01	0.57 ± 0.03	0.61 ± 0.02	0.20 ± 0.09
`ThreePastShop2cor(7)`	succ.	73.67 ± 35.97	40.17 ± 8.30	**98.95 ± 2.10**	**98.82 ± 1.27**	91.79 ± 18.36
	F-meas.	0.36 ± 0.18	0.16 ± 0.05	0.56 ± 0.04	0.58 ± 0.04	0.57 ± 0.12