An Efficient Particle Filter for Color-Based Tracking in Complex Scenes

Jesús Martínez-del-Rincón  Carlos Orrite-Uruñuela  J. Elías Herrero-Jaraba

Computer Vision Lab
Aragon Institute for Engineering Research
University of Zaragoza - Spain

Abstract

In this paper, we introduce an efficient method for particle selection in tracking objects in complex scenes. Firstly, we improve the proposal distribution function of the tracking algorithm, including current observation, reducing the cost of evaluating particles with a very low likelihood. In addition, we use a partitioned sampling approach to decompose the dynamic state in several stages. It enables to deal with high-dimensional states without an excessive computational cost. To represent the color distribution, the appearance of the tracked object is modelled by sampled pixels. Based on this representation, the probability of any observation is estimated using non-parametric techniques in color space. As a result, we obtain a Probability color Density Image (PDI) where each pixel points its membership to the target color model. In this way, the evaluation of all particles is accelerated by computing the likelihood \( p(z|x) \) using the Integral Image of the PDI.

1. Introduction

Tracking moving objects is one of the basic tasks performed by surveillance systems. So far, many tracking algorithms have been implemented for surveillance applications. One of the most popular methods for object tracking is the Particle Filter, since it achieves good performances in all cases including those where the target is partially occluded or several distracters appear in the scene [1, 5, 9, 11]. Particle filter can be regarded as a hypothesis tracker that approximates the filtered posterior distribution by a set of weighted hypotheses called particles. This technique weights particles based on a likelihood function and distributes them according to a motion model.

However when the scene/situation is complex the performance decreases significantly unless a large number of particles are used. This increases computational cost and time performance and affects further surveillance functionalities.

In this paper we propose a new particle filter algorithm based on two sampling techniques, which improves substantially the efficiency of the filter. The first sampling technique considered consists in the improvement of the proposal distribution, reducing the cost of evaluating particles with a very low likelihood. These particles occur due to poor prior density estimation, particularly when the motion of the object is badly modelled by the dynamic model. By including the current observation in the proposal distribution, we avoid this undesirable effect.

The second technique is the Partitioned sampling [11]. Originally, it was developed to track various objects utilising the same filter. This approach reduces the number of particles which grows with the state dimensionality. In this paper we use partitioned sampling to decompose the required parameters of the target (location, scale, shape, etc.). The a priori estimation of the first parameters facilitates to easily estimate the remaining ones, requiring thus a lower number of particles.

Since the present approach is based on a priori probability, i.e., a proposal distribution which introduces information about the current observation, we need to obtain the probability density function describing an observed target. The tracked object appearance is modelled by acquiring sample pixels from the object to represent the color distribution. Given this representation, the probability of any observation is estimated using non-parametric density methods in the color space. In this way, we obtain a Probability color Density Image (PDI) where every pixel points its membership to the target color image. The calculation of the likelihood \( p(z|x) \) of a certain observation \( z \) given a hypothesis \( x \) of the state of the system involves the integration over a set of observation points. To accomplish this task efficiently, the Integral Image of the PDI is calculated in advance. So, the integration is reduced to a simple combination of sums, accelerating the likelihood evaluation of all particles.

2 Importance Sampling Particle Filter

Monte Carlo methods approximate the target density by number of samples \( \{x^i_{t-1}, \omega^i_{t-1}\}_{i=1}^N \) that are distributed on
are computed iteratively as: the prior set of samples is generated and the importance weights to evaluate the likelihood and the transition probability. A proposal distribution is used to perform the sampling and to evaluate the likelihood and the transition probability. A proposal distribution controls the layout of the hypothesis in the dimensional space. Therefore, the distance density. Proposal distribution controls the layout of the probability of appearance, which is obtained using the proposal distribution, makes the algorithm prone to be distracted by background clutters.

An alternative approach consists in sampling the observation to improve the efficiency of particle filter. A function \( q(x_t) \), which introduces information about the current observation, is then applied as proposal distribution. Nevertheless, since particles are distributed using this proposal instead of the transition probability, they are not generated from a previous one and therefore, the particles cannot be paired with previous ones to compute the probability \( p(x_t|x_{t-1}) \). So an additional function \( f_d(x_t) \) is applied to maintain the temporal coherence. This term represents the probability of appearance, which is obtained using the weighted mean over all possible transitions.

\[
\omega_i^t \propto \omega_{i-1}^t \frac{p(z_t|x_t^i) g_i(x_t^i)}{f_d(x_t^i)} \cdot f_d(x_t^i) = \sum_{j=1}^N \omega_{i-1}^j \cdot p(x_t^j|x_{t-1}^j)
\]

This approach makes up for particle distribution and ensures that the importance function does not distort the calculation of the posterior probability \( p(x_t|z_t) \).

In practice, the proposal distribution derives from a rough observation process and might produce errors and imperfect estimations. For this reason, it is recommended to add a percentage of particles by conventional sampling.

This approach is similar to [13], where an auxiliary tracker generates a more accurate proposal distribution using secondary observations. On the other hand, our proposal uses the same features as the likelihood function to improve the sampling process.

3 Partitioned Sampling

Partitioned sampling is a sampling method developed by Mac Cormick and Blake [11], which allows dealing with high-dimensional states without an excessive computational cost. This sampling methodology will reduce the computational load from \( O(N^2) \) to nearly \( O(2N) \) mitigating the curse of dimensionality.

This technique is especially useful when the previous knowledge of some dimensions simplifies the search in the additional ones. Examples can be observed tracking several targets, in which it is more efficient to search first for the target that occludes another; or locating a target in the image before obtaining its shape.

Partitioned sampling decomposes the dynamic into two stages, between which an estimation of the first dimensions is calculated. It is crucial that this intermediate stage does not modify the distribution of the particle set. For this purpose, weighted resampling is applied. If this requirement is fulfilled, the dynamic can be decomposed into two steps, as follows:

\[
p(x_t|x_{t-1}) = \int_{\mathbf{X}_t} p(x_t|\hat{x}_t)p(\hat{x}_t|x_{t-1}) d\hat{x}_t
\]

Weighted resampling produces a new particle set by resampling proportionally to the importance function \( h_t(x_t^i) \). The weights of the resulting particle set are inversely proportional to the values of the importance function, which have been used to make the resampling. Thus, the overall distribution represented by the new particle set is the same as the old one.

\[
\omega_i^t \propto \omega_{i-1}^t p(z_t|x_t^i)/h_t(x_t^i)
\]

Weighted resampling is based on the same principles as importance sampling previously described. Both approaches share similar goals and results, but the methodology is entirely dissimilar. Importance sampling generates new particles from the importance function, adding later a correction weight. On the other hand, partition sampling evolves the new particles from the old particles through the importance function.
4 Efficient Partitioned Sampling PF

The resampling techniques presented before are complementary. Partitioned sampling enables to use a priori estimation of some dimensions, which can be extracted easily, for estimating the remaining ones. As an example we can mention the position coordinates, which are easily obtained with regards to the shape parameters. Once the location of the target is known, extracting its shape is a simpler task. Because these dimensions are more evident, estimating them with a rough measurement process as first approximation is a feasible task. Therefore, an importance function $g_t(x_t^i)$ can be obtained to be used as proposal distribution only for these main dimensions. Since the partitioned sampling can be repeated several times, we can decompose the estimation procedure for large dimensional spaces or to increase even more the efficiency. The compensation factor due to the better proposal distribution is only included at the first stage.

Algorithm 1 Efficient Particle Filter Proposal

Given a particle set $\{x_{t-1}^{i}, \omega_{t-1}^{i}\}_{i=1}^{N=N_1+N_2}$ which represents the posterior probability $p(x_{t-1}|z_{t-1})$ at time $t-1$

1. Generate $N_1$ new samples from the importance function in the main partitioned dimensions. $\hat{x}_t^i \sim g(x_t)$.
2. Propagate $N_2$ samples from the old samples applying dynamic $\hat{x}_t^i \sim p(\hat{x}_t|x_{t-1})$.
3. Weighted resampling: Particles $N_1$ are resampled using the importance function $h_t^i(x_t) = p(z_t|x_t^i) \cdot f_t(\hat{x}_t^i)/g_t(\hat{x}_t^i)$, and a compensation factor is applied to the weights $\hat{\omega}_t^i = \omega_{t-1}^i/h_t^i(\hat{x}_t)$.
4. Weighted resampling: Particles $N_2$ are resampled using the importance function $h_t^2(x_t) = p(z_t|x_t^2)$, and a compensation factor is applied to the weights $\hat{\omega}_t^2 = \omega_{t-1}^2/h_t^2(\hat{x}_t)$.
5. Propagate $N = N_1 + N_2$ samples from the generated samples applying dynamic in the remaining dimensions $x_t^i \sim p(x_t|x_{t-1})$.
6. Multiply by likelihood: weigh the particles $\omega_t^i = \hat{\omega}_t^i \cdot p(z_t|x_t^i) = \omega_{t-1}^i \cdot p(z_t|x_t^i)/h_t(\hat{x}_t) = \begin{cases} \omega_{t-1}^i \cdot \frac{p(z_t|x_t^i) \cdot g_t(\hat{x}_t^i)}{p(z_t|x_t^i)} & i = 1, \ldots, N_1 \\ \omega_{t-1}^i \cdot \frac{p(z_t|x_t^2)}{p(z_t|x_t^2)} & i = N_1 + 1, \ldots, N \end{cases}$ and normalize them $\sum_{i=1}^{N} \omega_t^i = 1$.
7. Estimate the new position of the state $\hat{E}[x_t] = \sum_{i=1}^{N} \omega_t^i \cdot x_t^i$.

Algorithm 1 describes the complete algorithm combining better proposal distribution and partitioned sampling. Note that a percentage of conventional particles $N_2$ have been added to compensate imperfect proposal distribution, as we mentioned in section 2.

5 Color density estimation

The observation process is the stage used to measure the likelihood of each particle and assign a corresponding weight, usually by comparison to an observation model. In this paper we consider how to use the efficient particle filter described earlier with color-based image features. Thus, the tracked object appearance is modelled by sampled pixels from the object to represent the color distribution.

Color density functions can be modelled by parametric [7] and non-parametric methods [8]. The major advantage of non-parametric approaches is the flexibility to represent complicated densities effectively since it does not assume any specific shape for the density function. Histogram is the simplest non-parametric density estimator. A classical and widespread approach uses color histogram as model and Bhattacharyya distance as similarity measurement [3, 4, 5, 8, 9]. However, this approach exhibits some drawbacks: spatial layout is lost, cluttered backgrounds can confuse the tracker and the computational cost may become excessive for large regions or high color resolutions.

By using histograms as models, the present approach applies a likelihood function which transforms the input image to the corresponding Probability color Density Image (PDI). Each pixel of this image points its membership to the target color model. The PDI is used in two ways: to generate the proposal distribution which introduces information about the current observation (prior probability), and to compute the likelihood of each particle (posterior probability).

A priori: Once the PDI has been calculated, the particles are distributed in those parts of the image with high PDI value. This is achieved following the proposal distribution function $g_t(x_t^i)$ described earlier.

A posteriori: The likelihood $p(z|x)$ of a certain observation $z$ given a hypothesis $x$ of the state (position, size, etc.) involves the integration over a set of observation points (PDI pixels). By calculating previously the integral image of the likelihood image, it becomes in a simple combination of sums, reducing the time to evaluate all particles.

5.1 Integral Image

The integral image was used by Viola and Jones [6], to evaluate a large number of rectangular masks on a gray scale image. By converting the gray scale to the integral image, where each pixel is the sum of all pixels above and left from
its current position, the estimation of each rectangular mask can be computed in four array reference.

In [14] authors describe a particle filter whose observation process combines a measurement of the similarity between color regions with an edge orientation histogram, in order to assign weights to the particles. Each color channel and each feasible gradient orientation give rise to an integral image. Then, likelihood functions (in this case, Euclidean distance) measure the similarity between target model and the value computed from the integral images. Our observation process is based on a very similar methodology. While [14] generates one integral image for each observed feature, we propose to summarize all them in a unique integral image. Integral image, traditionally a feature counter, becomes a membership counter by being generated from a PDI. Summarizing all features in an image emphasizes the importance depends on the number of rectangles used to rebuild the original shape. Although an infinite number of rectangles are needed, we show how adequate results can be obtained with a reasonable number of rectangles in Table 1.b.

The content of the rotated mask is computed as the content of the bounding-box minus the contents of several rectangles which are inside the triangles (Fig. 1.b). The larger number of rectangles, the lower error of the covered area. However, rotation angle is correctly estimated even with few rectangles (although few rectangles produce a lower resolution in the angle estimation).

Table 1: (a) Comparison between partitioned and traditional rotated rectangle estimation. (b) Error vs number of rectangles.

<table>
<thead>
<tr>
<th></th>
<th>Trad</th>
<th>Part</th>
<th>Rectan.</th>
<th>Area Er.</th>
<th>Ang. Er.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE x</td>
<td>0.48</td>
<td>0.58</td>
<td>5</td>
<td>25%</td>
<td>11.97</td>
</tr>
<tr>
<td>MSE y</td>
<td>0.73</td>
<td>0.57</td>
<td>13</td>
<td>12.5%</td>
<td>8.96</td>
</tr>
<tr>
<td>MSE ang.</td>
<td>2.47</td>
<td>3.52</td>
<td>33</td>
<td>6.25%</td>
<td>5.54</td>
</tr>
<tr>
<td>Time[s]</td>
<td>459.1</td>
<td>4.92</td>
<td>61</td>
<td>3.125%</td>
<td>3.79</td>
</tr>
</tbody>
</table>

Table I.a shows the differences in time (seconds) and in error (Mean Square Error) between estimating location and angle simultaneously (traditional), or extracting a first location estimation to help the final one (partitioned). We obtain a similar error with a much lower time cost because of the lower number of particles.

6 Experimental results

The proposal has been tested with different test sequences that illustrate different situations: location/angle/size tracking, static or dynamic camera, color or gray level model, single target or multi target.

The initial color or gray-level model corresponding to the target must be given in advance. The target appearance model, in this case based on color histogram, is updated in each frame to compensate lighting changes, following the approach shown in [5].

Particle propagation is made by a first order model: an object moving with constant velocity plus a stochastic component consisting in acceleration noise.

We have applied the present approach to the zodiac sequences taken from PETS 2005. In this domain of application, the zodiac tracking presents a high complexity due to multiple aspects like outdoor functioning with a high number of distracting moving targets (waves), dynamic non-rectangular or more complex shapes, we can select a simple strategy: decompose the shape into the smaller rectangles that it contains. Of course we always make an error, whose importance depends on the number of rectangles used to rebuild the original shape. Although an infinite number of rectangles are needed, we show how adequate results can be obtained with a reasonable number of rectangles in Table 1.b.
stationary occlusions, complex backgrounds and sudden changes in scale.

Given the first frame, the initial histogram of the moving object to track (the zodiac) is obtained. For this sequence, the PDI is obtaining following the next equation.

\[ p_v(h_{\text{target}}(u)|I(x, y)) = \exp(-\min_u(|I(x, y) - u|)), \]

where \( u \) is the gray level of the target histogram with some positive value. This probability gives a value between 1, perfect match and 0, the worst.

Particle distribution evolution throughout the different stages is depicted in Fig. 2. Once PDI (2.b) and its integral image have been calculated, particles are distributed in accordance with the proposal distribution (2.c). Evaluating these particles in the first dimensions (location), we obtain the weights (2.c) needed to make the weighted resampling. Resulting particles (2.d) are finally evaluated in the remaining dimensions (size/angle) (2.e) using the likelihood function to estimate the final state (2.f). Colors like yellow and magenta represent low weights while cyan and dark blue represent high values.

Fig. 3 shows the state target estimation (position and size) highlighted by a red rectangle. We have chosen these frames because the zodiac suffers from a sudden change in scale, so we can show the robustness of our method to cope with this situation.

We have compared the proposed approach with other similar color-based particle filters found in the literature [5, 9, 10]. Table 2 shows the results in mean square error and computational cost. As we can see, using Integral Image allows a relevant reduction factor for each particle estimation. Although an initial cost is added in order to create the integral matrix, it is compensated by the large number of particles usually required for the evaluation. The algorithm has been tested using MATLAB running on a Pentium 2.4 GHz so that the time costs obtained are only useful to compare one another.

The algorithm has been tested for multi-target tracking, applying an independent particle filter to each target, and solving the occlusion and conflicts using a multiple hypothesis tracking algorithm (MHT) [2] based on trajectories analysis. The results with this sequence are highly satisfactory as shown in Fig. 4 where target is highlighted in different colors. Finally, Fig. 5 and 6 show the results for the multi-face tracking AVSS sequences [16]. Fig. 7 compares traditional PF with our approach.

7 Conclusions

In this paper, we have introduced an efficient method for particle selection in tracking objects in complex scenes. The main contribution is the use of successive partitioned es-
timations to drive the particle set towards high likelihood regions avoiding local maxima in the posterior probability distribution. We have focused our approach on location, size and orientation estimation. Once the target probability density based on color is detected, the new set of particles is refined step-by-step through the partitioned dimensions. In order to speed up the observation process we have generated an integral image from the color probability image, allowing the use of rectangular masks for efficient weight computation. Thus, the computational cost is reduced enabling the simultaneous tracking of a large number of trackers.

The presented tracker is based on color, but it can be extended to other features (gradients, texture, movement, etc.) by applying the adequate likelihood function. Moreover, following this partitioned philosophy we could extend the methodology to estimate feature-based jointly with structure/shape-based parameters.

**Acknowledgments**

This work is partially supported by grants TIN2006-11044 and FEDER from MEyC. J. Martínez is supported by a FPI grant BES-2004-3741 from MEyC.

**References**


**Table 2**: Results obtained comparing our algorithm with conventional particle filter (zodiac sequence).

<table>
<thead>
<tr>
<th>N=150</th>
<th>N=500</th>
<th>N=1000</th>
<th>Our</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE [pix^2]</td>
<td>190.2</td>
<td>165.3</td>
<td>171.9</td>
</tr>
<tr>
<td>Mean N</td>
<td>150</td>
<td>500</td>
<td>1000</td>
</tr>
<tr>
<td>Time PDI [s]</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Time particles [s]</td>
<td>4.4</td>
<td>23.9</td>
<td>44.3</td>
</tr>
<tr>
<td>Total Time [s]</td>
<td>4.4</td>
<td>23.9</td>
<td>44.3</td>
</tr>
</tbody>
</table>

**Figure 6**: Webcam sequence. Frames: 2, 35, 50, 60, 88, 109, 133, 261, 284, 286. The PF fails because of a change of the illumination conditions which distorts the color model.

**Figure 7**: Comparison between traditional and our partitioned PF. Up: Time comparison. Down: Error in location, size and angle in AVSS sequences.