RAO-BLACKWELLIZED PARTICLE FILTER FOR GAUSSIAN MIXTURE MODELS AND APPLICATION TO VISUAL TRACKING

Jungho Kim and In So Kweon

Korea Advanced Institute of Science and Technology
373-1 Guseong-dong Yuseong-gu Daejeon, Korea
jhkim@rcv.kaist.ac.kr, iskweon@kaist.ac.kr

1. INTRODUCTION

Many approaches have been proposed to solve visual tracking problems that are defined as to estimate sequentially the state of a dynamic object using a sequence of noisy measurements. Particle filter-based visual tracking [1] has been extensively studied for this purpose. However, particle filter-based tracking algorithms have suffered from occlusion, large deformation of the target objects and illumination changes that result in the large difference between the current observations and the appearance model. Thus the strategy for updating the appearance model by using new observations is required to solve these problems. The RBPF [2] has been also applied to tracking problems to effectively estimate the high-dimensional joint distributions [3]. In [3], the authors addressed a RBPF-based tracking method consisting of estimation of the target location and estimation of subspace coefficients for eigenbases computed from a training set. It uses a Kalman filter to estimate a single Gaussian for coefficients. However, in the real data, new observations different from the training images are obtained due to deformation or varying illumination conditions and a single Gaussian has limitations in modeling the target appearance.

Comaniciu et al. [4] proposed a method for real-time tracking of non-rigid objects by using the mean-shift algorithm. Adam et al. [5] proposed an efficient scheme, in which an object is represented by multiple image fragments or patches, to handle partial occlusion or pose changes. Recently, Ross et al. [6] presented an online algorithm that incrementally learns and adapts a low dimensional eigenspace representation to reflect appearance changes of the target object. Zhou et al. [7] proposed a tracking approach that incorporates adaptive appearance models in a particle-filter framework. Babenko et al. [8] addressed the problem of learning an adaptive appearance model by training a discriminative classifier in an online manner.

In our approach, we estimate the joint distribution of both the target state and the appearance model. In general, estimation for this joint distribution is a high-dimensional problem, and it makes the particle filter infeasible. Thus we present an efficient tracking algorithm that adopts a Rao-Blackwellized particle filter, an effective method of reducing the dimensionality of a problem. Our adaptive appearance model is represented by a mixture of Gaussians (MoG) because a single Gaussian suffers from significant limitations when it comes to modeling real data set even though the Gaussian distribution has some important analytical properties. What is worse, if a single Gaussian is updated through the corrupted observations by occlusion or the tracking error, tracking is prone to failure in the subsequent frames. To solve these problems, we address a Rao-Blackwellized particle filter in which the appearance model is represented by a mixture of Gaussians.

2. RAO-BLACKWELLIZED PARTICLE FILTER

The system state vector consisting of the target state $x_t$ and the appearance model $a_t$ is first repartitioned as referred in Eq. (1). It decomposes the problem of estimating the joint distribution into an estimation problem of the target state and an estimation problem of the appearance model that is conditioned on the target state.

$$p(x_t, a_t | z_{1:t}) = p(x_t | z_{1:t}) p(a_t | x_t, z_{1:t}),$$

where $z_{1:t}$ are all observations obtained until time $t$. 

Index Terms— Visual Tracking, Rao-Blackwellized Particle Filter, Gaussian Mixture Model
In the Rao-Blackwellized particle filter, the distribution of the target state is estimated by marginalizing out the appearance model as

\[ p(x_t | z_{1:t}) = \int_{a_t} p(x_t, a_t | z_{1:t}) \, da_t = \kappa \int_{a_t} p(z_t | x_t, a_t) \, p(x_t, a_t | z_{1:t-1}) \, da_t, \tag{2} \]

where \( \kappa \) is a normalizing constant.

Under the Markov assumption, the prior distribution in Eq. (2) can be rewritten as

\[ p(x_t, a_t | z_{1:t-1}) = \int_{x_{t-1}} \int_{a_{t-1}} p(x_t, a_t, x_{t-1}, a_{t-1} | z_{1:t-1}) \, da_{t-1} \, dx_{t-1}. \tag{3} \]

Finally, the distribution of the target state can be decomposed into prior distributions for both the target state and the appearance model, and the likelihood distribution as

\[ p(x_t | z_{1:t}) \propto \kappa \int_{a_t} \left[ \int_{x_{t-1}} p(x_t, a_t, x_{t-1} | z_{1:t-1}) \, da_{t-1} \right] \, dx_{t-1}. \tag{4} \]

In our approach, the distribution of the target state \( p(x_t | z_{1:t}) \) is represented by a set of weighted particles and the distribution of the appearance model \( p(a_t | x_t, z_{1:t}) \) is represented by a mixture of Gaussians. Each particle consists of one target state \( x_{t}^{[i]} \) and its corresponding appearance model \( a_{t}^{[i]} \), and the distributions, \( p(x_{t}^{[i]} | z_{1:t}) \) and \( p(a_{t}^{[i]} | x_{t}^{[i]}, z_{1:t}) \) are individually estimated.

3. TARGET STATE ESTIMATION

3.1. Prior Distribution for Target State

Suppose that the required distribution \( p(x_{t-1} | z_{1:t-1}) \) at time \( t \) is available. Then the prior distribution is determined by the motion model \( p(x_t | x_{t-1}) \) which evolves the target candidates between time steps. In our approach, we adopt a random walk that is based on a uniform density about the previous state, \( x_{t-1} \), our uncertainty for the incremental changes of the target object, \( \nu \), and the velocity, \( v_t \), updated through the image sequence as follows:

\[ p(x_t | x_{t-1}) \sim U \left[ x_{t-1} - \nu + v_t, x_{t-1} + \nu + v_t \right] \quad v_t = (1 - \alpha_p) v_{t-1} + \alpha_p (x_{t-1} - x_{t-2}), \tag{5} \]

where \( \alpha_p \) is a constant learning rate.

3.2. Prior Distribution for Appearance Model

The distribution \( p(a_{t-1} | x_{t-1}, z_{1:t-1}) \) is given by an on-line approximation of a mixture of Gaussians, which will be presented in Sec. 4, as

\[ p(a_{t-1} | x_{t-1}, z_{1:t-1}) = \sum_{i=1}^{G} w_i \eta \left( a_{t-1}; \mu_i, \Sigma_i \right) \]

where \( \eta \left( a; \mu, \Sigma \right) \) is a Gaussian distribution, \( N (0, \sigma_p^2 I) \), then each Gaussian component, the mean and the covariance, are determined by

\[ \mu_i = \mu_{i-1}, \quad \Sigma_i = \Sigma_{i-1} + \sigma_p^2 I, \tag{6} \]

where \( I \) is an identity matrix.

In this step, we estimate the distribution \( p(a_t | x_t, z_{1:t}) \) by using an on-line approximation of a mixture of Gaussians and we calculate the appearance for each sub-region, as introduced in Sec. 4, that corresponds to the maximum a posteriori as

\[ a_t = \arg \max_{a_t} p(a_t | x_t, z_{1:t}) \quad \text{where} \quad p(a_t | x_t, z_{1:t}) = \sum_{i=1}^{G} w_i \eta \left( a_t; \mu_i, \Sigma_i \right) \tag{7} \]

We can calculate \( \hat{a_t} \) by employing the mean-shift algorithm as

\[ \hat{a}_t^{k+1} = \frac{\sum_{i=1}^{G} w_i \mu_i \eta \left( \hat{a}_t^{k}; \mu_i, \Sigma_i \right)}{\sum_{i=1}^{G} w_i \eta \left( \hat{a}_t^{k}; \mu_i, \Sigma_i \right)}, \tag{8} \]

where \( k \) is the number of iterations.

Iterations are terminated when \( \sum_{i=1}^{G} w_i \eta \left( \hat{a}_t^{k+1}; \mu_i, \Sigma_i \right) \leq \sum_{i=1}^{G} w_i \eta \left( \hat{a}_t^{k}; \mu_i, \Sigma_i \right) \), or it reaches the maximum iterations value. Then we calculate the probability for the appearance model by using the appearance \( \hat{a}_t \) and the prior distribution for the appearance model, as shown in Eq. (7), to calculate the weight for each particle as

\[ \int_{a_{t-1}} p(a_t | x_t, a_{t-1}) \, p(a_{t-1} | x_{t-1}, z_{1:t-1}) \, da_{t-1} = \sum_{i=1}^{G} w_i \eta \left( \hat{a}_t; \mu_i, \Sigma_i \right) \tag{9} \]

3.3. Likelihood Distribution

Our observation model is assumed to be the Gaussian distribution whose mean is the appearance computed by Eq. (10) and covariance is \( \sigma_p^2 I \). Thus the likelihood is determined by

\[ p(z_t | x_t, a_t) \sim N \left( \hat{a}_t, \sigma_p^2 I \right) = \eta \left( z_t; \hat{a}_t, \sigma_p^2 I \right) \tag{12} \]
4. APPEARANCE MODEL ESTIMATION

4.1. Appearance Model Representation

We use the 128-dimensional orientation histogram [9] for the observations $z_t$. In addition, we construct the integral images [10] to reduce the computational cost to compute histograms. When the target objects are partially occluded, that causes the large dissimilarity in that small occluded portions influence the entire orientation histograms. What is worse, if the occluding object contains highly textured regions, occluding parts strongly affect the tracking results because we generally use the entirely normalized histograms. Thus, we maintain the histograms from uniformly divided sub regions and individually normalize them in order that the proposed algorithm is not strongly influenced by the effect of partial occlusion. Thus the appearance model is spatially divided into $B$ sub regions to ensure the robustness to occlusion.

The appearance model $a_t$ is composed of independent $B$ appearance models for sub regions i.e. $a_t = [a_{t1}, a_{t2}, \ldots, a_{tB}]$, and those are modeled by a mixture of $G$ Gaussians as

$$p(a_t|x_t, z_t) = \prod_{j=1}^{B} p\left(a_{tj}|x_{tj}, z_{tj}\right)$$

$$p\left(a_{tj}|x_{tj}, z_{tj}\right) = \sum_{k=1}^{G} w_{jk} \eta\left(a_{tj}; \mu_{jk}, \Sigma_{jk}\right), \quad \Sigma_{jk} = \left(\sigma_{jk}\right)^2 I,$$

where $\mu_{jk}$ and $\Sigma_{jk}$ are the mean and the covariance corresponding to $k$th Gaussian for the $j$th sub region.

4.2. Appearance Model Update

We update the posterior distribution of the appearance model by using an on-line approximation of a mixture of Gaussians [11].

If none of the $G$ distributions match the current observations, the least probable distribution is replaced with a new Gaussian distribution whose mean value is the current histogram $z_{tj}$ and the pre-defined initial variance. Otherwise, we increase the weight for the Gaussian distribution ($n$th Gaussian) that matches the new observations, and $\mu$ and $\sigma$ parameters of the $n$th Gaussian are updated by

$$w_j^{n+1} = (1 - \alpha_n)w_j^{n} + \alpha_n$$

$$\mu_j^{n+1} = (1 - \rho^{n+1})\mu_j^{n} + \rho^{n+1}z_{tj}$$

$$(\sigma_j^{n+1})^2 = (1 - \rho^{n+1})(\sigma_j^{n})^2 + \rho^{n+1}(z_{tj} - \mu_j^{n})^T(z_{tj} - \mu_j^{n}),$$

where $\rho^{n+1} = \alpha_n \eta\left(z_{tj}; \mu_{j-1}^{n}, \Sigma_{j-1}^{n}\right)$ and $\alpha_n$ is a constant learning rate ($\alpha_n = 0.2$ in our implementation). And after this approximation, the weights are renormalized.

In our approach, the first Gaussian distribution, which is computed from the initial model image, remains same to prevent the drift problem.

5. EXPERIMENTAL RESULTS

We carried out the experiments using 300 particles for tracking and 8 Gaussians for the appearance model. We tested the proposed method using challenging sequences from the CAVIAR database $^2$, the face database $[5]$, and the PETS database $^3$.

The first row of Fig. 1 shows the tracking results for the CAVIAR database by using a single Gaussian. It shows a drift problem when the target object is partially occluded while the proposed method successfully track two target objects, as shown in the second row of Fig. 1. We show one particle that corresponds to the maximum weight.

![Fig. 1. The tracking results for the CAVIAR database by using (first row) a RBPF with a single Gaussian and (second row) the proposed method](image)

![Fig. 2. The tracking results for the face database [5] (first row) without dividing into sub-regions and (second row) with dividing into sub regions](image)

The video of the PETS database has 7 people who are randomly wandering with changing their appearances. Some of them are occluded by static objects or other moving people. Fig. 3 shows the tracking results for a total of 119 frames by using the proposed method. For this sequence we compared


$^3$PETS datasets available at http://www.evg.cs.rdg.ac.uk/slides/pets.html
The tracking results for the PETS database by using the proposed method.

![Frame 2](image1.png) ![Frame 19](image2.png) ![Frame 36](image3.png)

![Frame 65](image4.png) ![Frame 84](image5.png) ![Frame 119](image6.png)

**Table 1.** Performance comparison: we show the success rate (%) (the number of frames). Each target is classified by color in Fig. 3. * The second target (Blue) disappeared from frame 72.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>61(73)</td>
<td>87(104)</td>
<td>82(97)</td>
<td>100(119)</td>
</tr>
<tr>
<td>Blue</td>
<td>49(35)</td>
<td>100(72)</td>
<td>100(72)</td>
<td>100(72)</td>
</tr>
<tr>
<td>Green</td>
<td>75(89)</td>
<td>57(68)</td>
<td>53(63)</td>
<td>100(119)</td>
</tr>
<tr>
<td>Yellow</td>
<td>7(8)</td>
<td>11(13)</td>
<td>20(24)</td>
<td>100(119)</td>
</tr>
<tr>
<td>Cyan</td>
<td>4(5)</td>
<td>13(15)</td>
<td>100(119)</td>
<td>100(119)</td>
</tr>
<tr>
<td>Magenta</td>
<td>28(33)</td>
<td>48(57)</td>
<td>45(54)</td>
<td>100(119)</td>
</tr>
<tr>
<td>White</td>
<td>29(35)</td>
<td>29(35)</td>
<td>46(55)</td>
<td>100(119)</td>
</tr>
</tbody>
</table>

The performance of the proposed method with other methods that are a mean-shift-based tracker [12], the MIL tracker [8] and a RBPF-based tracker with the appearance model represented by a single Gaussian (SG). For this purpose, we show the success rate for each target in Table 1.

The proposed algorithm was implemented in C++, and we used a 2.4GHz CPU and 3GB RAM. The average processing time for one target is 13.5ms for this image sequence, i.e. it takes averagely 94.5ms to simultaneously track 7 people from each image whose resolution is 768 × 576 pixels.

**6. CONCLUSION**

We presented an effective tracking method using a Rao-Blackwellized particle filter to incrementally update the appearance model and to simultaneously track the target object. We presented a method of estimating the distribution of the target state by marginalizing out the appearance model represented by a mixture of Gaussians and updating the appearance model by using new observations. We demonstrated the robustness of the proposed method for tracking problems related to occlusion, deformation and varying illumination conditions.

**7. ACKNOWLEDGEMENT**

This work was supported by Samsung Techwin and partially supported by MKE(The Ministry of Knowledge Economy), Korea, under the Human Resources Development Program for Convergence Robot Specialists support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2010-C7000-1001-0007)

**8. REFERENCES**


