Object Tracking with an Adaptive Color-Based Particle Filter

Katja Nummiaro[1], Esther Koller-Meier[2], and Luc Van Gool[1,2]

[1] Katholieke Universiteit Leuven, ESAT/VISICS, Belgium
{knummiar,vangool}@esat.kuleuven.ac.be
[2] Swiss Federal Institute of Technology (ETH), D-ITET/BIWI, Switzerland
{ebmeier,vangool}@vision.ee.ethz.ch

Abstract. Color can provide an efficient visual feature for tracking non-rigid objects in real-time. However, the color of an object can vary over time dependent on the illumination, the visual angle and the camera parameters. To handle these appearance changes a color-based target model must be adapted during temporally stable image observations.
This paper presents the integration of color distributions into particle filtering and shows how these distributions can be adapted over time. A particle filter tracks several hypotheses simultaneously and weights them according to their similarity to the target model. As similarity measure between two color distributions the popular Bhattacharyya coefficient is applied. In order to update the target model to slowly varying image conditions, frames where the object is occluded or too noisy must be discarded.

1 Introduction

Object tracking is required by many vision applications, but especially in video technology [2, 5, 7, 12]. Tracking methods can be divided into two main classes specified as *bottom-up* or *top-down* approaches. In a *bottom-up* approach the image is generally segmented into objects which are then used for the tracking. For example blob detection [12] can be used for the object extraction. In contrast, a *top-down* approach generates object hypotheses and tries to verify them using the image. Typically, model-based [5, 7] or template matching approaches [2] comprise this class. The proposed particle filter with color-based image features belongs to the *top-down* approaches as the image content is only evaluated at the hypothetical object positions.

The idea of a particle filter – to apply a recursive Bayesian filter based on sample sets – was independently proposed by several research groups [3, 5, 7, 11]. These filters provide robust tracking frameworks as they are neither limited to linear systems nor require the noise to be Gaussian. In this paper we present the integration of color distributions into particle filtering, which has typically used edge-based image features [4, 5, 7, 13]. Color histograms have many advantages for tracking non-rigid objects as they are robust to partial occlusion, are rotation and scale invariant and are calculated efficiently. In [8] color information has already been employed in particle filtering for foreground and background model using Gaussian mixtures. Our target model has the advantage of matching only objects that have a similar histogram, whereas for Gaussian mixtures objects that contain one of the colors of the mixture will already match.

The color of an object can vary over time dependent on the illumination, the visual angle and the camera parameters. To handle these appearance changes the color model must be adapted during temporally stable image observations. Particle filtering has already been used with several static target models [4], but to the best of our knowledge it has not yet been applied with an adaptive model.

A related tracking approach which also uses color histograms is the mean shift tracker [2]. In comparison, our proposal employs multiple hypotheses and a model of the system dynamics which results in a more reliably tracking in cases of clutter and occlusions. Adaptive models have been discussed in [9, 14], but both approaches employ Gaussian mixture models while we use color histograms together with multiple hypotheses.

The outline of this paper is as follows. In Section 2 we briefly describe particle filtering and in Section 3 we indicate how color distributions are used as object models. The integration of the color information into the particle filter is explained in Section 4 and Section 5 describes the model update. Finally, some experimental results are presented in Section 6.

2 Particle Filtering

Particle filtering [5, 7] was developed to track objects in clutter, in which the posterior density $p(X_t|Z_t)$ and the observation density $p(Z_t|X_t)$ are often non-Gaussian. The quantities of a tracked object are described in the state vector X_t while the vector Z_t denotes all the observations $\{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$ up to time t.

The key idea of particle filtering is to approximate the probability distribution of the object state by a weighted sample set $S = \{\mathbf{s}^{(n)}, \pi^{(n)})|n = 1 \ldots N\}$. Each sample consists of an element $\mathbf{s}$ which represents the hypothetical state of the object and a corresponding discrete sampling probability π where $\sum_{n=1}^{N} \pi^{(n)} = 1$.

The evolution of the sample set is calculated by propagating each sample according to a system model. Each element of the set is then weighted in terms of the observations and N samples are drawn with replacement, by choosing a particular sample with probability $\pi^{(n)} = p(\mathbf{z}_t|X_t = \mathbf{s}_t^{(n)})$. The mean state of the object is estimated at each time step by

$$E[S] = \sum_{n=1}^{N} \pi^{(n)} \mathbf{s}^{(n)}. \tag{1}$$

Particle filtering provides a robust tracking framework, as it models uncertainty. It can keep its options open and consider multiple state hypotheses simultaneously. Since less likely object states have a chance to temporarily remain in the tracking process, particle filters can deal well with short-lived occlusions.

3 Color Distributions

We want to apply such a particle filter in a color-based context. To achieve robustness against non-rigidity, rotation and partial occlusion we focus on color distributions as target models. These are represented by histograms which are typically calculated in the RGB space using 8x8x8 bins.

Not all pixels in a region are equally important to describe an object. For example, pixels that are further away from the region center can be assigned smaller weights by employing a weighting function

$$k(r) = \begin{cases} 1 - r^2 & : \quad r < 1 \\ 0 & : \quad \text{otherwise} \end{cases} \tag{2}$$

where r is the distance from the region center. Thus, we increase the reliability of the color distribution when these boundary pixels belong to the background or get occluded. It is also possible to use a different weighting function for example the Epanechnikov kernel [2].

The color distribution $p_{\mathbf{y}} = \{p_{\mathbf{y}}^{(u)}\}_{u=1\ldots m}$ of a region R at location $\mathbf{y}$ is calculated as

$$p_{\mathbf{y}}^{(u)} = f \sum_{\mathbf{x}_i \in R} k\left(\frac{\|\mathbf{y} - \mathbf{x}_i\|}{a}\right) \delta[h(\mathbf{x}_i) - u] \tag{3}$$

where δ is the Kronecker delta function and $h(\mathbf{x}_i)$ assigns one of the m-bins of the histogram to a given color at location $\mathbf{x}_i$. The variable a provides invariance against scaling of the region and the normalization factor f ensures that $\sum_{u=1}^{m} p_{\mathbf{y}}^{(u)} = 1$.

In a tracking approach the estimated state is updated at each time step by incorporating the new observations. Therefore, a similarity measure is needed between the color distributions of a region in the newly observed image and the target model. A popular measure between two distributions is the Bhattacharyya coefficient [1, 10]. Considering discrete densities such as two color histograms $p = \{p^{(u)}\}_{u=1\ldots m}$ and $q = \{q^{(u)}\}_{u=1\ldots m}$ the coefficient is defined as

$$\rho[p, q] = \sum_{u=1}^{m} \sqrt{p^{(u)} q^{(u)}}. \tag{4}$$

The larger ρ is, the more similar the distributions are. For two identical histograms we obtain $\rho = 1$, indicating a perfect match. As distance between two distributions we define the measure

$$d = \sqrt{1 - \rho[p, q]} \tag{5}$$

which is called the Bhattacharyya distance.

4 Color-based Particle Filtering

The proposed tracker employs the Bhattacharyya distance to update the a priori distribution calculated by the particle filter. The target regions are represented by ellipses, so that a sample is given as

$$\mathbf{s} = \{x, y, \dot{x}, \dot{y}, H_x, H_y, \dot{H}_x, \dot{H}_y\} \tag{6}$$

where x, y represent the location of the ellipse, $\dot{x}$, $\dot{y}$ the motion, H_x, H_y the length of the half axes and $\dot{H}_x$, $\dot{H}_y$ the corresponding scale changes.

The sample set is propagated through the application of a dynamic model

$$\mathbf{s}_t = A\,\mathbf{s}_{t-1} + \mathbf{w}_{t-1} \tag{7}$$

where A defines the deterministic system model and $\mathbf{w}_{t-1}$ a random vector drawn from the noise distribution of the system. In our application we currently use a first order model for A describing an object moving with constant velocity for x, y, H_x and H_y. Expanding this model to second order is straightforward.

To weigh the sample set, the Bhattacharyya coefficients are computed between the target distribution and the distributions at the locations of the hypotheses. Each hypothetical region is specified by its state vector $\mathbf{s}^{(n)}$. Both the target q and the candidate histogram $p_{\mathbf{s}^{(n)}}$ are calculated from Eq. 3 where the target is centered in the origin and $a = \sqrt{H_x^2 + H_y^2}$.

The observation probability of each sample

$$\pi^{(n)} = \frac{1}{\sqrt{2\pi}\sigma}\, e^{-\frac{d^2}{2\,\sigma^2}} = \frac{1}{\sqrt{2\pi}\sigma}\, e^{-\frac{(1-\rho[p_{\mathbf{s}^{(n)}},q])}{2\,\sigma^2}} \tag{8}$$

is specified by a Gaussian with variance σ. During filtering, samples with a high weight may be chosen several times, leading to identical copies, while others with relatively low weights may not be chosen at all.

A comparison of this approach to the mean shift tracker [2] and different methods for the initialization respectively re-initialization of the color-based particle filter are described in [15]. An extension for multiple objects and the usage of more than one histogram per sample can be found in [16].

5 Model Update

Illumination conditions, the visual angle as well as the camera parameters can influence the quality of the color-based particle filter. To overcome these appearance changes we update the target model during slowly changing image observations. By discarding image outliers — where the object is occluded or too noisy — the tracker can be protected against updating the model when the object has been lost. So, we use the update rule

$$\pi_{E[S]} > \pi_T \tag{9}$$

where $\pi_{E[S]}$ is the observation probability in terms of the mean state $E[S]$ and π_T is a threshold.

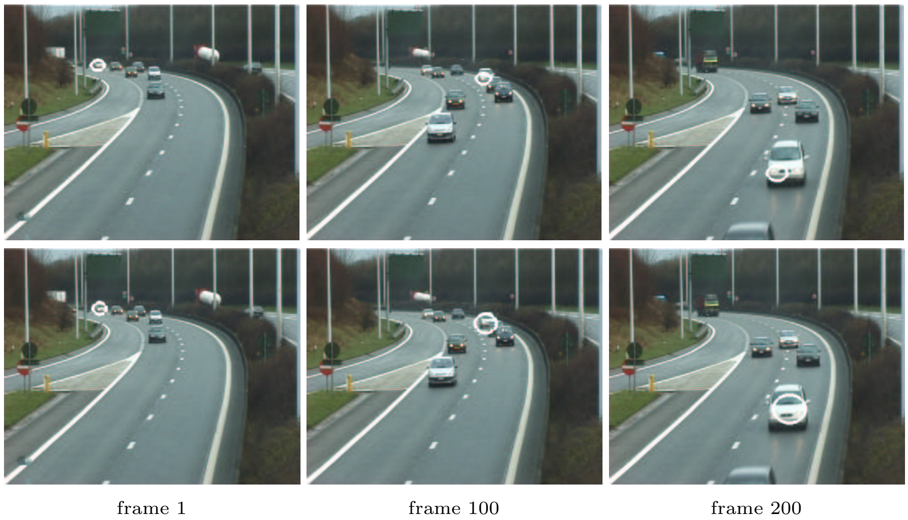

Fig. 1. The *traffic* sequence illustrates the importance of an adaptive target model in cases of occlusions and large scale changes. The white ellipses represent the mean states of the underlying sample distribution of $N = 100$ elements. In the top row the tracking results without a model update are presented while in the bottom row an update is applied.

The update of the target model is implemented by the equation

$$q_t^{(u)} = (1 - \alpha)\, q_{t-1}^{(u)} + \alpha\, p_{E[S_t]}^{(u)} \tag{10}$$

for each bin u where α weighs the contribution of the mean state histogram $p_{E[S_t]}$ to the target model q_{t-1}. Thus, we evoke a forgetting process in the sense that the contribution of a specific frame decreases exponentially the further it lies in the past.

6 Results

To illustrate the adaptive color-based particle filter and its behavior, we applied the proposed method to a *traffic* and a *face* sequence and show the tracking results. The experiments have been processed with a 800 MHz Pentium3 PC under Linux, using the RGB color space with 8x8x8 bins and images of size 360x288 pixels. The goal of the experiments has been to track a manually initialized object region (car, human) during the sequence until it has disappeared.

To show the importance of the model update we regard the *traffic* sequence of 234 frames recorded by a highway monitoring system. There is an evident scale change during this sequence as the camera was placed towards the traffic flow. Furthermore, different viewing angles of the car and partial occlusions make the experiment more difficult. In the top row of Figure 1 no model update is

Fig. 2. The *face* sequence shows the tracking performance with an adaptive model. The tracker handles a large object motion and illumination changes using $N = 100$ samples.

performed and the resulting region gets stuck on the left front side of the car. In contrast, the bottom row shows the effectiveness of the model update.

In Figure 2 the *face* sequence of 600 frames is shown which has been captured with a strong sun/shadow effect. At the beginning of the sequence the face is situated in the shadow for the initialization and at the end of the sequence it is moved into the sun. The tracked face is affected by changing illumination conditions and facial expressions as well as a full turn of the person and large scale changes. In frame 400, the tracked position is not very exact as the model does not match the back of the head very well. Nevertheless, the person can still be tracked and the positions improves rapidly once the person has turned around.

The target model of our tracker is only updated according to Eq. 9 as outliers must be discarded, for example when the person is turning. A related update rule is given by the maximization of the log-likelihood [9] over the last M frames: $L = \sum_{t=1}^{M} log\, \pi_{E[S]}^{(t)}$. In Figure 3 both update possibilities are plotted for the *face* sequence. The two update approaches behave similarly in the sense that a model update is only performed under slowly varying image conditions. As the history of samples through the log-likelihood does not significantly improve the results, we use our more efficient method.

Finally, Figure 4 illustrates the running times of the adaptive color-based particle filter for the *face* sequence for two different image sizes. The proposed approach has real-time capability but the processing time is dependent on the region size and the number of samples.

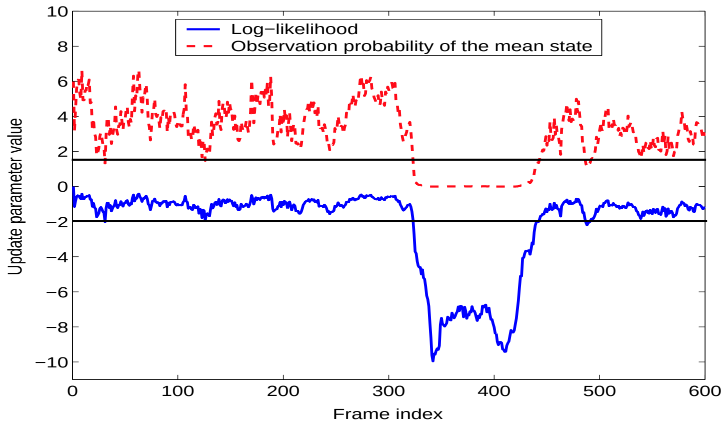

Fig. 3. The log-likelihood L and the observation probability $\pi_{E[S]}$ (here scaled with factor 10) can be both applied as an update rule with an appropriate threshold.

7 Conclusion

The proposed tracking method adds an adaptive appearance model based on color distributions to particle filtering. The color-based tracker can efficiently and successfully handle non-rigid and fast moving objects under different appearance changes. Moreover, as multiple hypotheses are processed, objects can be well tracked in cases of occlusion or clutter. As a limitation of the proposed approach the tracker might loose an object when it changes appearance quickly — for example through occlusion — and makes a rapid movement at the same time.

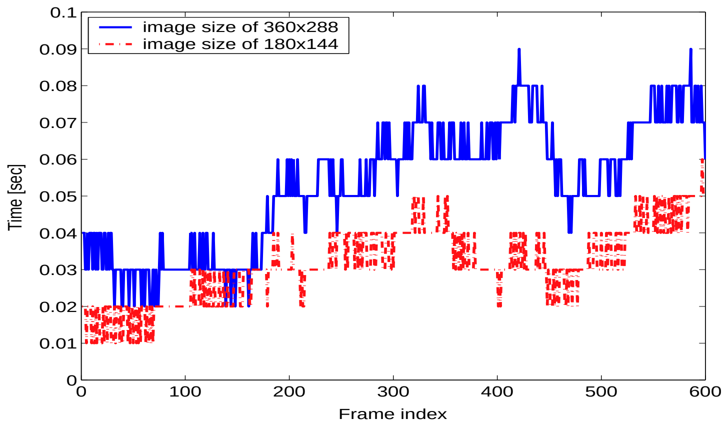

Fig. 4. Running times for the *face* sequence for two different image sizes using $N = 100$ samples.

The application of an adaptive model comprises always a trade-off between an increasing sensitivity to extended occlusions and a more reliable tracking under appearance changes.

Acknowledgment

The authors acknowledge the support by STAR (IST-2000-28764) which is a project of the European Commission and by the GOA/VHS+ project financed by the Research Fund KUL.

References

1. F. Aherne, N. Thacker and P. Rockett, *The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data*, Kybernetika, pp. 1-7, Vol. 32(4), 1997.
2. D. Comaniciu, V. Ramesh and P. Meer, *Real-Time Tracking of Non-Rigid Objects using Mean Shift*, CVPR, pp. 142-149, Vol. 2, 2000.
3. N. Gordon and D. Salmond, *Bayesian State Estimation for Tracking and Guidance Using the Bootstrap Filter*, Journal of Guidance, Control and Dynamics, pp. 1434-1443, Vol. 18(6), November-December, 1995.
4. T. Heap and D. Hogg, *Wormholes in Shape Space: Tracking through Discontinuous Changes in Shape*, ICCV, pp. 344-349, 1998.
5. M. Isard and A. Blake, *Contour Tracking by Stochastic Propagation of Conditional Density*, ECCV, pp. 343-356, Vol. 1, 1996.
6. M. Isard and A. Blake, *ICONDENSATION: Unifying Low-Level and High-Level Tracking in a Stochastic Framework*, ECCV, pp. 893-908, Vol. 1, 1998.
7. M. Isard and A. Blake, *CONDENSATION – Conditional Density Propagation for Visual Tracking*, International Journal on Computer Vision, pp. 5-28, Vol. 1(29), 1998.
8. M. Isard and J. MacCormick, *BraMBLe: A Bayesian Multiple-Blob Tracker*, ICCV, pp. 34-41, 2001.
9. A. Jepson, D. Fleet and T. El-Maraghi, *Robust Online Appearance Models for Visual Tracking*, CVPR, pp. 415-422, Vol. 1, 2001.
10. T. Kailath, *The Divergence and Bhattacharyya Distance Measures in Signal Selection*, IEEE Transactions on Communication Technology, pp. 52-60, Vol. COM-15(1), 1967.
11. G. Kitagawa, *Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models*, Journal of Computational and Graphical Statistics, pp. 1-25, Vol. 5(1), 1996.
12. D. Koller, J. Weber and J. Malik, *Robust Multiple Car Tracking with Occlusion Reasoning*, ECCV, pp. 189-196, 1994.
13. J. MacCormick and A. Blake, *A Probabilistic Exclusion Principle for Tracking Multiple Objects*, ICCV, pp. 572-587, Vol. 1, 1999.
14. S. McKenna, Y. Raja and S. Gong, *Tracking Colour Objects Using Adaptive Mixture Models*, Journal of Image and Vision Computing, pp. 225-231, Vol. 17, 1999.
15. K. Nummiaro, E. Koller-Meier and L. Van Gool, *A Color-Based Particle Filter*, First International Workshop on Generative-Model-Based Vision, in conjunction with ECCV'02, pp. 53-60, 2002.
16. P. Pérez, C. Hue, J. Vermaak and M. Gangnet, *Color-Based Probabilistic Tracking*, ECCV, pp. 661-675, Vol. 1, 2002.