A study on Object Tracking using Color-based Particle Filter

Mai Thanh Nhat Truong*, Sanghoon Kim*
*Department of Electrical, Electronic, and Control Engineering, Hankyong National University

Abstract

Object tracking in video sequences is a challenging task and has various applications. Particle filtering has been proven very successful for non-Gaussian and non-linear estimation problems. In this study, we first try to develop a color-based particle filter. In this approach, the color distributions of video frames are integrated into particle filtering. Color distributions are applied because of their robustness and computational efficiency. The model of the particle filter is defined by the color information of the tracked object. The model is compared with the current hypotheses of the particle filter using the Bhattacharyya coefficient. The proposed tracking method directly incorporates the scale and motion changes of the objects. Experimental results have been presented to show the effectiveness of our proposed system.

1. Introduction

Object tracking is required in many vision-based applications such as human-computer interfaces, video communication, road traffic control, security and surveillance systems. The main goal of object tracking is to obtain a record of the trajectory of the moving single or multiple targets over time and space, by processing information from distributed sensors. Object tracking in video sequences requires on-line processing of a large amount of data. Additionally, most of the problems encountered in visual tracking are nonlinear, non-Gaussian, multi-modal or any combination of these. Therefore, tracking objects can be extremely complex and time-consuming, especially when it is done in outdoor environments.

Recently, the particle filter method, a numerical method that allows finding an approximate solution to the sequential estimation, has been shown to be very successful for nonlinear and non-Gaussian estimation problems. This approximates a posterior probability density of the state, such as the object's position, by using samples which are called particles. The particle filter based tracking algorithms usually use contours, color features, or appearance models [1, 2, 3, 4, 5]. The color histogram is robust against noise and partial occlusion, but suffers from illumination changes, or the presence of the confusing colors in the background.

Although particle filters have been widely used in recent years, they have important drawbacks. One is sampling impoverishment, i.e., samples are spread around several modes pointing out the different hypotheses in the state space, but most of these may be spurious. In addition, the objects with a higher likelihood may monopolize the sample set, and objects whose samples exhibit a lower likelihood have a higher probability of being lost. On the other hand, the computation is expensive if the tracked region and the number of samples are large. The contour-based methods are invariant against the illumination variation but computationally expensive which restricts the number of samples (particles). Unfortunately when the dimensionality of the state space increases, the number of samples required for the sampling increases exponentially.

2. Particle filter

2.1 Classical particle filter

Particle filtering [1] was developed to track objects in clutter, in which the posterior density $p(X_t \mid Z_t)$ and the observation density $p(Z_t \mid X_t)$ 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 $\{z_1, \dots, z_t\}$ up to time t.
The key idea of particle filtering is to approximate the probability distribution by a weighted sample set $S = \{(s_n, \pi_n) \mid n = 1 \dots N\}$. Each sample consists of an element s which represents the hypothetical state of an object and a corresponding discrete sampling probability π. The evolution of the sample set is described by propagating each sample according to a system model. Then, each element of the set is weighted in terms of the observations and N samples are drawn with replacement, by choosing a particular sample with probability $\pi_n = p(z_t \mid X_t = s_t(n))$. The mean state of an object is estimated at each time step by

$$E[S] = \sum_{n=1}^{N} \pi_n s_n \tag{1}$$

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 with short lived occlusions.

2.2 Color Model

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 produced with the function $h(x_i)$, that assigns one of the m-bins to a given color at location x_i. The histograms are typically calculated in the RGB space using 8x8x8 bins. To make the algorithm less sensitive to lighting conditions, the HSV color space could be used instead with less sensitivity to V (e.g. 8x8x4 bins).

Not all pixels in the region are equally important to describe the objects. 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.

The color distribution $p(y) = \{p_u(y)\}_{u=1\ldots m}$ at location y is calculated as

$$p_u(y) = f \sum_{i=1}^{I} k\left(\frac{\|y - x_i\|}{a}\right) \delta[h(x_i) - u] \tag{3}$$

where δ is the Kronecker delta function and a is used to adapt the size of the region. The normalization factor is

$$f = \frac{1}{\sum_{i=1}^{I} k\left(\frac{\|y - x_i\|}{a}\right)} \tag{4}$$

In a tracking approach the estimated state is updated at each time step by incorporating the new observations. Therefore, we need a similarity measure which is based on color distributions. A popular measure between two distributions $p(u)$ and $q(u)$ is Bhattacharyya coefficient [6],

$$\rho[p,q] = \int \sqrt{p(u)q(u)}du \tag{5}$$

Considering discrete densities such as our color histograms $p = \{p_u\}_{u=1\ldots m}$ and $q = \{q_u\}_{u=1\ldots m}$ the coefficient is defined as

$$\rho[p,q] = \sum_{u=1}^{m} \sqrt{p_u q_u} \tag{6}$$

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

$$d = \sqrt{1 - \rho[p,q]} \tag{7}$$

which is called the Bhattacharyya distance.

3. Experiments

In this section, we present the performance of object tracking method. The algorithm is implemented in C++, using OpenCV library. The tracking process is shown in Figure 1. It shows the result of the color-based particle filter using N = 2000 samples, in this case the histograms are calculated in the RGB color space using 8x8x8 bins. The figure shows that the color-based particle filter yields good performance when applied in object tracking.

4. Conclusions and future work

In this study we developed a color-based particle filter for object tracking. The experimental results show that our method capable of tracking object presented in color video, yielding high accuracy and reliable results. For further development, we will develop a multiple object tracking for visual inspection system, which uses color-based particle filter. The aim of future tracking method is making it robust in dealing with different problems in visual object tracking. This multiple object tracking will also have low computational complexity in order to be deployed on embedded systems.

5. Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A1A01057518).

References

[1] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. Int'l Journal of Computer Vision, 29(1):5–28, Aug. 1998.

[2] M. Isard and A. Blake. Condensation: Unifying low-level and high-level tracking in a stochastic framework. In Proc. European Conf. Computer Vision, volume 1, pages 893–908, 1998.

[3] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In Proc. Int'l Conf. Computer Vision, pages 572–578, 1999.

[4] Y. Wu. Robust visual tracking by integrating multiple cues based on co-inference learning. Int'l Journal of Computer Vision, 58(1):55–71, June 2004.

[5] Shun-Sheng Ko, Chien-Sheng Liu, Yang-Cheng Lin, Optical Z. Khan, T. Balch, and F. Dellaert. A rao-blackwellized particle filter for eigentracking. In Proc. IEEE Conf. Comp. Vision Pattern Recognition, volume 2, pages 980–986, Washington DC, 2004.

[6] F. Aherne, N. Thacker, and P. Rockett. The Bhattacharyya metric as an absolute similarity measure for frequency coded data. Kybernetika, 32(4):1–7, 1997.

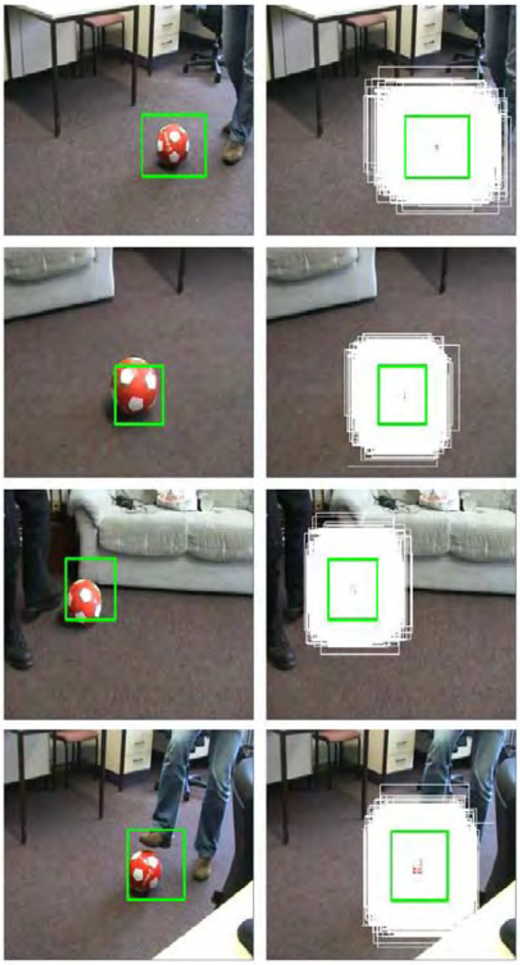

Figure 1. The object tracking method is used to track a ball in video. Each row shows tracked object (left) and the locations of corresponding particles (right).