MEAN SHIFT AND OPTIMAL PREDICTION FOR EFFICIENT OBJECT TRACKING

Dorin Comaniciu and Visvanathan Ramesh

Imaging and Visualization Department, Siemens Corporate Research
755 College Road East, Princeton, NJ 08540
{comanici, vramesh}@scr.siemens.com

Abstract

A new paradigm for the efficient color-based tracking of objects seen from a moving camera is presented. The proposed technique employs the mean shift analysis to derive the target candidate that is the most similar to a given target model, while the prediction of the next target location is computed with a Kalman filter. The dissimilarity between the target model and the target candidates is expressed by a metric based on the Bhattacharyya coefficient. The implementation of the new method achieves real-time performance, being appropriate for a large variety of objects with different color patterns. The resulting tracking, tested on various sequences, is robust to partial occlusion, significant clutter, target scale variations, rotations in depth, and changes in camera position.

1. INTRODUCTION

Object tracking is a task required by different computer vision applications, such as perceptual user interfaces [3], intelligent video compression [8], and surveillance [12]. To achieve robustness to out-of-plane rotations of the target, the color distribution of the target model is employed instead of raw image pixels. The location of the target in the new frame is predicted based on the past trajectory, and a search is performed in its neighborhood for image regions (target candidates) whose distribution is similar to that of the model. In single hypothesis tracking the best match determines the new location estimate, however, more complex strategies also exist to form multiple hypothesis [1].

The exhaustive search in the neighborhood of the predicted target location for the best target candidate is, however, a computationally intensive process. As a solution to this problem we propose a color-based tracking method based on the mean shift iterations [4, 5] which works in real time, being based on a gradient ascent optimization rather than exhaustive search. The measurement vector is derived based on mean shifts, while the prediction of the next target location is computed by a Kalman filter (Figure 1).

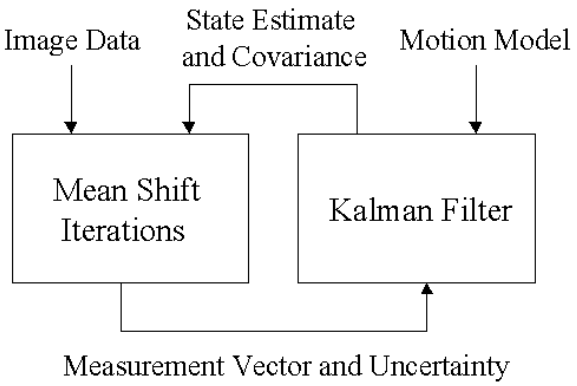

Fig. 1. Block diagram showing the main computational modules of the proposed tracking: the fast target localization based on mean shift iterations and the state prediction using Kalman filtering. The motion of the target is assumed to have a velocity that undergoes slight changes, modeled by a zero mean white noise that affects the acceleration.

It is assumed next the support of two modules which should provide (a) detection and localization in the initial frame of the objects to track (targets) [12], and (b) periodic analysis of each object to account for possible updates of the target models due to significant changes in color [13].

The organization of the paper is as follows. Section 2 presents the employed similarity measure. The mean shift based localization of the target is described in Section 3. Section 4 discusses the Kalman filter, while the scale adaptation is presented in Section 5. Experimental results are given in Section 6.

2. COLOR-BASED SIMILARITY MEASURE

Given the predicted location of the target in the current frame and its uncertainty, the measurement task assumes the search of a confidence region for the target candidate that is the most similar to the target model. The similarity measure we develop is based on color information. The feature $\mathbf{z}$ representing the color of the target model is assumed to have a density function $q_{\mathbf{z}}$, while the target candidate centered

at location $\mathbf{y}$ has the feature distributed according to $p_{\mathbf{z}}(\mathbf{y})$. The problem is to find the discrete location $\mathbf{y}$ whose associated density $p_{\mathbf{z}}(\mathbf{y})$ is the closest to the target density $q_{\mathbf{z}}$.

Our measure of the distance between the two densities is based on the Bhattacharyya coefficient, whose general form is defined by [11]

$$\rho(\mathbf{y}) \equiv \rho\left[p(\mathbf{y}), q\right] = \int \sqrt{p_{\mathbf{z}}(\mathbf{y}) q_{\mathbf{z}}}\, d\mathbf{z} \ . \tag{1}$$

Properties of the Bhattacharyya coefficient such as its relation to the Fisher measure of information, quality of the sample estimate, and explicit forms for various distributions are discussed in [7, 11].

The derivation of the Bhattacharyya coefficient from sample data involves the estimation of the densities p and q, for which we employ the histogram formulation. The discrete density $\hat{\mathbf{q}} = \{\hat{q}_u\}_{u=1\ldots m}$ (with $\sum_{u=1}^{m} \hat{q}_u = 1$) is estimated from the m-bin histogram of the target model, while $\hat{\mathbf{p}}(\mathbf{y}) = \{\hat{p}_u(\mathbf{y})\}_{u=1\ldots m}$ (with $\sum_{u=1}^{m} \hat{p}_u = 1$) is estimated at a given location $\mathbf{y}$ from the m-bin histogram of the target candidate. Therefore, the sample estimate of the Bhattacharyya coefficient is given by

$$\hat{\rho}(\mathbf{y}) \equiv \rho\left[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}\right] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\mathbf{y}) \hat{q}_u}. \tag{2}$$

Based on equation (2) we define the distance between two distributions as

$$d(\mathbf{y}) = \sqrt{1 - \rho\left[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}\right]} \ . \tag{3}$$

The statistical measure (3) is a metric valid for arbitrary distributions, being nearly optimal (due to its link to the Bayes error [11]) and invariant to the scale of the target. It is therefore superior to other measures such as histogram intersection [14], Bhattacharyya distance, Fisher linear discriminant [10], or Kullback divergence.

3. TARGET LOCALIZATION

This section shows how to efficiently minimize (3) as a function of $\mathbf{y}$ in the neighborhood of a predicted location. By contrast to object tracking based on exhaustive search in a confidence region [2, 9, 12], our optimization through mean shift iterations is faster since it exploits the spatial gradient of the measure (3).

3.1. Weighted Histogram Computation

Target Model We denote by $\{\mathbf{x}_i^\star\}_{i=1\ldots n}$ the pixel locations of the target model, centered at $\mathbf{0}$. Let $b : R^2 \rightarrow \{1 \ldots m\}$ be function which associates to the pixel at location $\mathbf{x}_i^\star$ the index $b(\mathbf{x}_i^\star)$ of the histogram bin corresponding to the color of that pixel. The probability of the color u in the target model is derived by employing a convex and monotonic decreasing function $k : [0, \infty) \rightarrow R$ which assigns a smaller weight to the locations that are farther from the center of the target. The weighting increases the robustness of the estimation, since the peripheral pixels are the least reliable, being often affected by occlusions (clutter) or background. By assuming that the generic coordinates x and y are normalized with h_x and h_y, respectively, we can write

$$\hat{q}_u = C \sum_{i=1}^{n} k(\|\mathbf{x}_i^\star\|^2) \delta\left[b(\mathbf{x}_i^\star) - u\right] \ , \tag{4}$$

where δ is the Kronecker delta function. The normalization constant C is derived by imposing the condition $\sum_{u=1}^{m} \hat{q}_u = 1$, from where

$$C = \frac{1}{\sum_{i=1}^{n} k(\|\mathbf{x}_i^\star\|^2)} \ , \tag{5}$$

the summation of delta functions for $u = 1 \ldots m$ being equal to one.

Target Candidates Let us denote by $\{\mathbf{x}_i\}_{i=1\ldots n_h}$ the pixel locations of the target candidate, centered at $\mathbf{y}$ in the current frame. Employing the same weighting function k, the probability of the color u in the target candidate is given by

$$\hat{p}_u(\mathbf{y}) = C_h \sum_{i=1}^{n_h} k\left(\left\| \frac{\mathbf{y} - \mathbf{x}_i}{h} \right\|^2\right) \delta\left[b(\mathbf{x}_i) - u\right] \ . \tag{6}$$

The scale of the target candidate (i.e., the number of pixels) is determined by the constant h which plays the same role as the bandwidth (radius) in the case of kernel density estimation [5]. By imposing the condition that $\sum_{u=1}^{m} \hat{p}_u = 1$ we obtain the normalization constant

$$C_h = \frac{1}{\sum_{i=1}^{n_h} k(\|\frac{\mathbf{y} - \mathbf{x}_i}{h}\|^2)}. \tag{7}$$

Note that C_h does not depend on $\mathbf{y}$, since the pixel locations $\mathbf{x}_i$ are organized in a regular lattice, $\mathbf{y}$ being one of the lattice nodes. Therefore, C_h can be precalculated for a given kernel and different values of h.

3.2. Distance Minimization

The search for the new target location in the current frame starts at the predicted location $\hat{\mathbf{y}}_0$ of the target computed by the Kalman filter (Figure 1). Thus, the color probabilities $\{\hat{p}_u(\hat{\mathbf{y}}_0)\}_{u=1\ldots m}$ of the target candidate at location $\hat{\mathbf{y}}_0$ in the current frame have to be computed first.

The minimization of the distance (3) being equivalent to the maximization of the Bhattacharyya coefficient (2), we start with the Taylor expansion of $\rho\left[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}\right]$ around the values $\hat{p}_u(\hat{\mathbf{y}}_0)$, which yields

$$\rho\left[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}\right] \approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{\hat{p}_u(\hat{\mathbf{y}}_0) \hat{q}_u} + \frac{1}{2} \sum_{u=1}^{m} \hat{p}_u(\mathbf{y}) \sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}} \tag{8}$$

Introducing now (6) in (8) we obtain

$$\rho\left[\hat{\mathbf{p}}(\mathbf{y}),\hat{\mathbf{q}}\right] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{\mathbf{y}}_0)\hat{q}_u} + \frac{C_h}{2}\sum_{i=1}^{n_h} w_i k\left(\left\|\frac{\mathbf{y}-\mathbf{x}_i}{h}\right\|^2\right) \quad (9)$$

where

$$w_i = \sum_{u=1}^{m}\delta\left[b(\mathbf{x}_i)-u\right]\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}}. \quad (10)$$

Hence, to minimize the distance (3), the second term in equation (9) has to be maximized, the first term being independent of $\mathbf{y}$. The second term represents the density estimate computed with kernel profile k at $\mathbf{y}$ in the current frame, with the data being weighted by w_i (10). The maximization can be efficiently achieved based on the mean shift iterations (see [5]), using the following algorithm.

Maximization of Bhattacharyya Coefficient $\rho\left[\hat{\mathbf{p}}(\mathbf{y}),\hat{\mathbf{q}}\right]$

Given the distribution $\{\hat{q}_u\}_{u=1\ldots m}$ of the target model and the predicted location $\hat{\mathbf{y}}_0$ of the target:

1. Compute the distribution $\{\hat{p}_u(\hat{\mathbf{y}}_0)\}_{u=1\ldots m}$, and evaluate
$$\rho\left[\hat{\mathbf{p}}(\hat{\mathbf{y}}_0),\hat{\mathbf{q}}\right] = \textstyle\sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathbf{y}_0)\hat{q}_u}\,.$$
2. Derive the weights $\{w_i\}_{i=1\ldots n_h}$ according to (10).
3. Derive the new location of the target [5]
$$\hat{\mathbf{y}}_1 = \frac{\sum_{i=1}^{n_h}\mathbf{x}_i w_i g\left(\left\|\frac{\hat{\mathbf{y}}_0-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{\hat{\mathbf{y}}_0-\mathbf{x}_i}{h}\right\|^2\right)}\,.$$
Update $\{\hat{p}_u(\hat{\mathbf{y}}_1)\}_{u=1\ldots m}$, and evaluate
$$\rho\left[\hat{\mathbf{p}}(\hat{\mathbf{y}}_1),\hat{\mathbf{q}}\right] = \textstyle\sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathbf{y}_1)\hat{q}_u}\,.$$
4. While $\rho\left[\hat{\mathbf{p}}(\hat{\mathbf{y}}_1),\hat{\mathbf{q}}\right] < \rho\left[\hat{\mathbf{p}}(\hat{\mathbf{y}}_0),\hat{\mathbf{q}}\right]$ Do $\hat{\mathbf{y}}_1 \leftarrow \frac{1}{2}\left(\hat{\mathbf{y}}_0+\hat{\mathbf{y}}_1\right)$.
5. If $\|\hat{\mathbf{y}}_1-\hat{\mathbf{y}}_0\| < \epsilon$ Stop. Otherwise Set $\hat{\mathbf{y}}_0 \leftarrow \hat{\mathbf{y}}_1$ and go to Step 1.

The above optimization employs the mean shift vector in Step 3 to increase the value of the approximated Bhattacharyya coefficient $\tilde{\rho}(\mathbf{y})$. Since this operation does not necessarily increase the value of $\hat{\rho}(\mathbf{y})$, the test included in Step 4 is needed to validate the new location of the target. However, practical experiments (tracking different objects, for long periods of time) showed that the Bhattacharyya coefficient computed at the location defined by equation (11) was almost always larger than the coefficient corresponding to $\hat{\mathbf{y}}_0$. Less than 0.1% of the performed maximizations yielded cases where the Step 4 iterations were necessary. The termination threshold ϵ used in Step 5 is derived by constraining the vectors representing $\hat{\mathbf{y}}_0$ and $\hat{\mathbf{y}}_1$ to be within the same pixel in image coordinates.

3.3. Measurement Uncertainty

The uncertainty in the localization of the target is determined by the image noise, the similarity between the target colors and background/clutter colors, and the percentage of occlusion. However, the perturbation sources also influence the maximum value of the Bhattacharyya coefficient and the curvature around the maximum. Since these two parameters (the maximum value and the curvature around maximum) can be evaluated in real time, we derived through Monte-Carlo simulations a lookup-table that relates the maximum value and the surface curvature to the uncertainty in the location estimate. As a result, after each mean shift optimization that gives the measured location of the target, the uncertainty of the estimate can also be computed.

4. KALMAN PREDICTION

The tracker employs two independent Kalman filters, one for each direction x and y. The target motion is assumed to have a slightly changing velocity ([1, p. 82]) modeled by a zero mean, low variance (0.01) white noise that affects the acceleration.

The tracking process consists in running for each frame the mean shift based optimization which determines the measurement vector and its uncertainty, followed by the Kalman iteration which gives the predicted position of the target and a confidence region. These entities are used in turn to initialize the mean shift optimization for the next frame.

5. SCALE ADAPTATION

The scale adaptation scheme exploits the property of the distance (3) to be invariant to changes in the object scale. We simply modify the bandwidth h of the kernel profile with a certain fraction (we used $\pm 10\%$), let the mean shift based algorithm to converge again, and choose the radius yielding the largest decrease in the distance (3). An IIR filter is used to derive the new radius based on the current measurements and old radius.

6. EXPERIMENTS

The proposed tracking has been applied to various test sequences with superior performance and low computational complexity. Figure 2 shows the successful tracking in the presence of a complete occlusion of the hand-drawn ellipsoidal region of size $(h_x, h_y) = (55, 39)$ marked in the first image. Note that the target histogram has been derived in the RGB space with $32 \times 32 \times 32$ bins. The algorithm runs comfortably at 30 fps on a 600 MHz PC, Java implementation.

Figure 3 shows samples from a sequence taken with a moving camera, demonstrating the tracking of an electronic device whose colors are close to those of the background. One can observe the scale adaptation provided by the algorithm.

Fig. 2. *Tennis* sequence: The frames 21, 47, and 52 are shown (left-right).

Fig. 3. *Device* sequence: The frames 1, 100, 200, and 300 are shown.

7. REFERENCES

[1] Y. Bar-Shalom, T. Fortmann, *Tracking and Data Association*, Academic Press, London, 1988.

[2] S. Birchfield, "Elliptical Head Tracking using intensity Gradients and Color Histograms," *IEEE Conf. on Comp. Vis. and Pat. Rec.*, Santa Barbara, 232–237, 1998.

[3] G.R. Bradski, "Computer Vision Face Tracking as a Component of a Perceptual User Interface," *IEEE Work. on Applic. Comp. Vis.*, Princeton, 214–219, 1998.

[4] D. Comaniciu, V. Ramesh, P. Meer, "Real-Time Tracking of Non-Rigid Objects using Mean Shift, To appear, *IEEE Conf. on Comp. Vis. and Pat. Rec.*, Hilton Head Island, South Carolina, 2000.

[5] D. Comaniciu, P. Meer, "Mean Shift Analysis and Applications," *IEEE Int'l Conf. Comp. Vis.*, Kerkyra, Greece, 1197–1203, 1999.

[6] D. Comaniciu, P. Meer, "Distribution Free Decomposition of Multivariate Data", *Pattern Anal. and Applic.*, 2:22–30, 1999.

[7] A. Djouadi, O. Snorrason, F.D. Garber, "The Quality of Training-Sample Estimates of the Bhattacharyya Coefficient," *IEEE Trans. Pattern Analysis Machine Intell.*, 12:92–97, 1990.

[8] A. Eleftheriadis, A. Jacquin, "Automatic Face Location Detection and Tracking for Model-Assisted Coding of Video Teleconference Sequences at Low Bit Rates," *Signal Processing - Image Communication*, 7(3): 231–248, 1995.

[9] P. Fieguth, D. Terzopoulos, "Color-Based Tracking of Heads and Other Mobile Objects at Video Frame Rates," *IEEE Conf. on Comp. Vis. and Pat. Rec*, Puerto Rico, 21–27, 1997.

[10] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Second Ed., Academic Press, Boston, 1990.

[11] T. Kailath, "The Divergence and Bhattacharyya Distance Measures in Signal Selection," *IEEE Trans. Commun. Tech.*, COM-15:52–60, 1967.

[12] A.J. Lipton, H. Fujiyoshi, R.S. Patil, "Moving Target Classification and Tracking from Real-Time Video," *IEEE Workshop on Applications of Computer Vision*, Princeton, 8–14, 1998.

[13] S.J. McKenna, Y. Raja, S. Gong, "Tracking Colour Objects using Adaptive Mixture Models," *Image and Vision Computing*, 17:223–229, 1999.

[14] M.J. Swain, D.H. Ballard, "Color Indexing," *Intern. J. Comp. Vis.*, 7(1):11–32, 1991.

[15] "Real-Time Tracking of Non-Rigid Objects using Mean Shift," US patent pending.