REAL-TIME TRACKING OF NON-RIGID OBJECTS USING MODIFIED KERNEL-BASED MEAN SHIFT AND OPTIMAL PREDICTOIN

Amin Merati

Sharif University of Technology
Department of Electrical Engineering
P.O.Box 15875-4412, Tehran, Iran
merati@mehr.sharif.edu

Shohreh Kasaei

Sharif University of Technology
Department of Computer Engineering
P.O.Box 11365-9517, Tehran, Iran
skasaei@sharif.edu

ABSTRACT

An efficient scheme for real-time color-based tracking of non-rigid objects is proposed. The central computational module is based on mean shift iterations. It computes the most probable target position in the current frame, while the prediction of the next target location is computed using a Kalman filter. The dissimilarity between the target model and the target candidates is expressed by a metric based on the Bhattacharyya coefficient. In this work, we have adapted the kernel profile (used in calculating the feature histogram) with a binary mask generated by proposed adaptive background subtraction scheme. The modified kernel calculates the feature histogram only for foreground pixels and prevents background pixels from causing the estimation process to deviate. The adaptive background subtraction algorithm may fail under varying illumination and shadow conditions. To overcome this problem, we have decomposed the incoming image into its intrinsic components (illuminance and reflectance), and have designed an adaptive background subtraction scheme using the reflectance image. The experimental results show the capability of the proposed tracker to handle real-time partial occlusions, significant clutter, and also target scale variations.

1. INTRODUCTION

Object tracking is a task required by different computer vision applications, such as perceptual user interface [3], intelligent video compression [7], and surveillance [11]. To achieve robustness to out-of-plane rotations of the target, the color distribution of the target model is employed instead of the raw image pixels. The location of the target in the new frame is predicted based on the past trajectory and then a search is performed in its neighborhood to determine the image regions (target candidates) whose distribution is similar to that of the model. In single hypothesis tracking schemes the best match determines the new location estimation; however, more complex strategies also exist to form multiple hypotheses [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, a color-based tracking method based on mean shift iterations [4, 5] is proposed. This method works in real time; as it is based on the gradient ascent optimization rather than the 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 shows the block diagram of the main computational modules of the proposed tracking algorithm. The fast target localization is based on the 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.

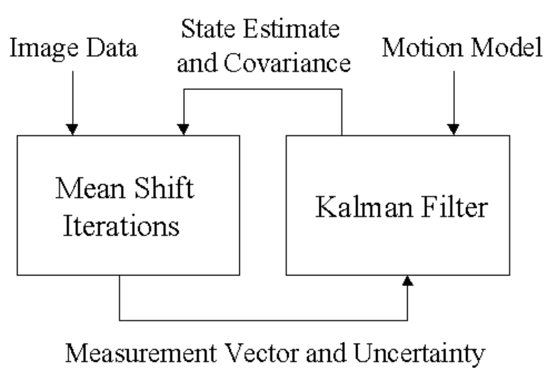


Figure 1. *Block diagram of the main computational modules of the proposed tracking algorithm*.

In many situations, due to the similarity between the target and background colors, the mean shift vector deviates to a background region. Thus, here we use a binary mask to modify the kernel used in [4] (calculating the weighted histogram of the regions) to prevent the mean shift vector to deviate to the background regions.

The organization of paper is as follows. Section 2 presents the employed similarity measure, the mean shift-based target localization without kernel modification and the Kalman filter. Section 3 discusses the proposed algorithm which consists of the kernel modification and the illumination invariance background subtraction processes. Experimental results are described in Section 4. The paper is concluded in Section 5.

2. KERNEL-BASED MEAN SHIFT TRACKING

2.1. COLOR –BASED SIMILARITY MEASURE

Given the predicted location of the target in 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 developed similarity measure is based on color information. The feature **z** representing the color of the target model and is assumed to have a density function $q_{\mathbf{z}}$, while the target candidate centered at location **y** has the feature distributed according to $p_z(\mathbf{y})$. Now, the problem is to find the discrete location **y** whose associated density $p_z(\mathbf{y})$ is the closest to the target density q_z .

Our measure of the distance between the two densities is based on the Bhattacharyya coefficient, whose general form is defined by [10]:

$$\rho(\mathbf{y}) \equiv \rho[p(\mathbf{y}), q] = \int \sqrt{p_z(\mathbf{y}) q_z} \, dz. \tag{1}$$

Properties of the Bhattacharyya coefficient such as its relation to the Fisher measure of information, quality of the sample estimate, and its explicit forms for different distributions are discussed in [6, 10].

The derivation of the Bhattacharyya coefficient from sample data involves the estimation of the densities p and q, for which the histogram has been employed. 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 **y** from the *m*-bin histogram of the target candidate. Therefore, the sample estimate of the Bhattacharyya coefficient is given by:

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

Based on equation (2) the distance between two distributions is defined as:

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

The statistical measure (3) is a metric valid for arbitrary distributions, being nearly optimal (due to its link to the Bayes error [10]) and invariant to scale of the target. It is therefore superior to other measures such as histogram intersection [12], Fisher linear discriminant [9], or Kullback divergence.

2.2. TARGET LOCALIZATION

This section explains how to efficiently minimize (3) as a function of **y** in the neighborhood of a predicted location. In contrast to object tracking based on exhaustive search in a confidence region [2, 8, 11], the optimization through mean shift iterations is faster since it exploits the spatial gradient measure (3).

2.2.1. Weighted Histogram Computation

I) Target Model: The pixel locations of the target model centered at 0 are denoted by $\{\mathbf{x}_i^*\}_{i=1\ldots n}$ .Let $b: R^2 \rightarrow \{1\ldots m\}$ be the function that associates to the pixel at location $\mathbf{x}_i^*$ and the index $b(\mathbf{x}_i^*)$ of the histogram bin correspond 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 target center. This weight increases the robustness of the estimation; since the peripheral pixels are the least reliable, being often affected by occlusions (clutter) or background areas. By assuming that the generic coordinates *x* and *y* are normalized by h_x and h_y , respectively, we get:

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

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:

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

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

II) Target Candidates: Let denote the pixel locations of the target candidate, centered at **y** in the current frame, by $\{\mathbf{x}_i\}_{i=1\ldots n_h}$. Employing the same weighting function k, the probability of occurrence 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 the kernel density estimation [5]. By imposing the condition that $\sum_{u=1}^{m} \hat{p}_u = 1$, we obtain the normalization constant as:

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

Note that C_h does not depend on **y**; since the pixel locations $\mathbf{x}_i$ are organized in a regular lattice (**y** denotes the lattice node). Therefore, the C_h can be precalculated for a given kernel with different values of h.

2.2.2. 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 current frame have to be computed first.

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

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

By substituting (6) in (8) we get:

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

where,

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

Hence, to minimize the distance (3), the second term in equation (9) has to be maximized (the first term is independent to **y**). The second term represents the density estimate computed with kernel profile *k* at ***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.

To maximize the Bhattacharyya coefficient $\rho[\hat{\mathbf{p}}(\mathbf{y}),\hat{\mathbf{q}}]$, 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:

   $$\rho[\mathbf{p}(\hat{\mathbf{y}}_0),\hat{\mathbf{q}}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathbf{y}_0)\hat{q}_u}.$$

2. Derive the weights $\{w_i\}_{i=1\ldots nh}$ according to (10).
3. Derive the new location of the target [5]:

   $$\hat{\mathbf{y}}_1 = \frac{\sum_{i=1}^{nh}\mathbf{x}_i w_i g\left(\left\|\frac{\hat{\mathbf{y}}_0-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{nh} w_i g\left(\left\|\frac{\hat{\mathbf{y}}_0-\mathbf{x}_i}{h}\right\|^2\right)}.$$

   Update $\{\hat{p}_u(\hat{\mathbf{y}}_1)\}_{u=1\ldots m}$, and evaluate:

   $$\rho[\hat{\mathbf{p}}(\hat{\mathbf{y}}_1),\hat{\mathbf{q}}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathbf{y}_1)\hat{q}_u}.$$

4. While $\rho[\hat{\mathbf{p}}(\hat{\mathbf{y}}_1),\hat{\mathbf{q}}] < \rho[\hat{\mathbf{p}}(\hat{\mathbf{y}}_0),\hat{\mathbf{q}}]$,

   Do: $\hat{\mathbf{y}}_1 \leftarrow \frac{1}{2}(\hat{\mathbf{y}}_0+\hat{\mathbf{y}}_1).$
5. If $\|\hat{\mathbf{y}}_1-\hat{\mathbf{y}}_0\| < \varepsilon$, 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 location defined by equation (11) was almost always lager than the coefficient corresponding to $\hat{\mathbf{y}}_0$. Less than 0.1% of the performed maximizations yielded cases where Step 4 was 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.

2.2.3. Measurement Uncertainty

The uncertainty in the target localization mainly caused by the image noise, the similarity between the target colors and the 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, a lookup-table that relates the maximum value and the surface curvature to uncertainty in location estimate through the Monte-Carlo simulations has been derived. As a result, after each mean shift optimization that gives the target measured location, the uncertainty of the estimation can be computed.

2.3. KALMAN PREDICTION

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

The tracking process for each frame consists of running 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.

3. PROPOSED ALGORITHM

3.1. KERNEL MODIFICATION

The above mentioned method for real time tracking assumes that the Bhattacharyya surface formed from the Bhattacharyya coefficients in pixels around the predicted location is smooth and has one maximum which corresponds to the real location of the target. But it is seen in the experiments that due to the similarity of the background colors (histogram) to the target colors, the Bhattacharyya surface has more than one maximum and sometimes the value of the other maximums is larger than the maximum in real location. This causes the mean shift to converge to a local maximum. In this work, we have modified the kernel used in Section 2 to calculate the histogram of the model and the candidate, so that it prevents background pixels to be computed in the histogram.

In order to modify the kernel, a binary mask that is obtained by an adaptive background subtraction is used. This binary mask specifies whether a pixel belongs to the background or the foreground. By using this binary mask, the candidate region histogram is only computed in foreground pixels preventing background pixels to be computed in histogram and also the deviation of mean shift iterations to a local maximum in the Bhattacharyya surface. The binary mask in the n*th* frame is given by:

$$Binary\_Mask_n : \Re^2 \rightarrow \{0,1\} \tag{11}$$

$$Binary\_Mask_n(\mathbf{x}_i) = \begin{cases} 1 & if\ \mathbf{x}_i \in foreground \\ 0 & if\ \mathbf{x}_i \in background \end{cases} \tag{12}$$

In order to use this binary mask, the former kernel is multiplied by binary mask defined above. The new modified kernel weights the pixels in the candidate region not only due to the distance to the region center, but also due to the likelihood of the foreground pixels. By this multiplication, the target model histogram will be:

$$\hat{q}_u = C\sum_{i=1}^{n} k(\|\mathbf{x}_i^*\|^2)\delta\left[b(\mathbf{x}_i^*)-u\right]Binary\_Mask_n(\mathbf{x}_i) \tag{13}$$

in which the normalization constant (C) is given by:

$$C = \frac{1}{\sum_{i=1}^{n} k(\|\mathbf{x}_i^*\|^2)Binary\_Mask_n(\mathbf{x}_i)}, \tag{14}$$

The histogram of the candidate region is computed by:

$$\hat{p}_u(\mathbf{y}) = C_h \sum_{i=1}^{nh} k\left(\left\|\frac{\mathbf{y}-\mathbf{x}_i}{h}\right\|^2\right)\delta[b(\mathbf{x}_i)-u]Binary\_Mask_n(\mathbf{x}_i), \quad (15)$$

where,

$$C_h = \frac{1}{\sum_{i=1}^{nh} k\left(\left\|\frac{\mathbf{y}-\mathbf{x}_i}{h}\right\|^2\right)Binary\_Mask_n(\mathbf{x}_i)}, \quad (16)$$

In computing the new location, the weights w_i are given by:

$$w_i = \sum_{u=1}^{m} \delta[b(\mathbf{x}_i)-u]Binary\_Mask_n(\mathbf{x}_i)\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}} \quad (17)$$

Other formulas will be the same as in Section 2. The iterations are performed using the new formulas.

3.2. ILLUMINATION INVARIANCE BACKGROUND SUBTRACTION

The biggest problem with background subtraction tends to be its failure under conditions of varying illumination and shadows. To overcome this problem, we propose to decompose an incoming image into its intrinsic components; illuminance and reflectance [12]. A background subtraction routine that is very easily fooled by weather/illumination changes would be brittle if it were applied on an intrinsic image as opposed to the original image. However, recovering the intrinsic components of a single color image is an ill-defined problem [13]. In most practical surveillance applications, it is justifiable to assume that the scene illuminance varies smoothly. Since the new image model is required to be invariant not only to the global change lighting but also to the smooth variations of the distribution, the illuminance component should be acquired as local as possible. Therefore, a local Gaussian low-pass filter is employed in our framework described below. This process is known as the homomorphic filtering [14] and results in separating the reflection components from the image. Denoting the input image by $I(x, y)$, the reflectance image by $R(x, y)$ and the illumination image by $L(x, y)$, we have:

$$I(x, y) = L(x, y)R(x, y). \quad (18)$$

Then:

$$\ln I(x, y) = \ln L(x, y) + \ln R(x, y) \quad (19)$$

and,

$$\ln L = G_{3*3} \otimes \ln I \quad (20)$$

also,

$$r' = \exp(\ln I - \ln L) \quad (21)$$

where G_{3*3} is the 3x3 Gaussian mask, $\boldsymbol{I}$ is the image, and $\otimes$ denotes the convolution operation. Applying (21) to each color channel yields:

$$r'_R = \exp(\ln I_R - \ln L) \quad (22)$$

$$r'_G = \exp(\ln I_G - \ln L) \quad (23)$$

$$r'_B = \exp(\ln I_B - \ln L) \quad (24)$$

Combining the reflectance components of all these channels, we get the reflectance image corresponding to the original color image. After this preprocessing step, we apply the regular adaptive background subtraction technique.

4. EXPERIMENTAL RESULTS

The proposed algorithm was used to track the objects contained in the test sequences of our database. The database consists of 50 color sequences containing more than 200 surveillance objects (humans and vehicles). The size of sequences frames is 576×720. We have run the algorithm using Visual C++ on a 1.2 GHz PC. A typical sequence is shown in Figure 2. It is worth to mention that other sequences resulted in similar results. In this sequence the pedestrians are tracked. To so this, first a rectangular patch is selected as the model of a pedestrian. The target histogram is derived in the RGB space with $32 \times 32 \times 32$ bins. The algorithm runs comfortably at 30 fps. This figure shows 3 samples from a sequence (frames 2, 21, 27) using mean shift tracking without kernel modification by binary mask (a) and with kernel modification by binary mask (b). As it is seen, due to the similarity between the background colors (histogram) and the target colors (histogram), the rectangular patch in red in left (a) is deviated to some other place in shadow. But due to the kernel modification by binary mask, this deviation does not occur in (b).

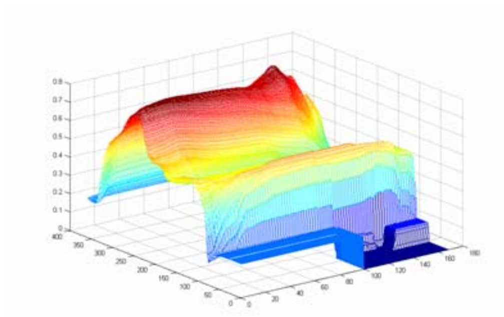
Figure 3 shows the Bhattacharyya surface for pixels around the predicted location in frame 21 for two cases with kernel modification (a) and without kernel modification (b). As seen in Figure.3, the surface for the case without kernel modification by binary mask (b) has many local maxima and the global maximum does not correspond to the real location of the target. In contrast to this case, the surface obtained for the case with the kernel modification by the binary mask has a global maximum that corresponds to the real location of the target and the surface has some small values which correspond to the background. Our algorithm performance is much better in comparison surveillance tracking algorithms [11, 15].

5. CONCLUSION

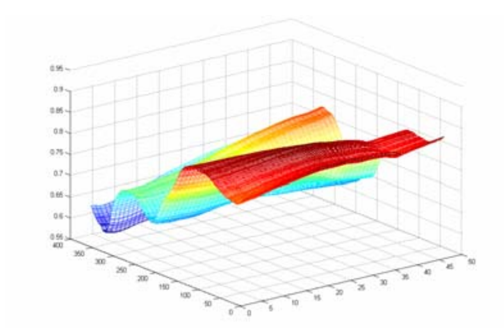
In this paper, we proposed a tracking algorithm that improves the mean shift object tracking performance by modification of the kernel using a binary mask obtained from our proposed adaptive background subtraction scheme. As a result, the probability of deviation of the algorithm to some local maxima has been decreased. Experimental results show the superior performance of the proposed algorithm for surveillance applications. In such applications due to ill illumination conditions, the histogram of some incorrect locations is very similar to that of the target histogram and thus the proposed method can efficiently track the real location of the objects. Also, in order to make the background subtraction scheme more robust under varying illumination, the homomorphic filtering for recovering the reflectance image is used that further improved the performance.

(a) (b)

Figure 2.Tracking a pedestrian without kernel modification (a) and with kernel modification (b).



(a)



(b)

Figure 3.Bhattacharyya surface for 2 cases with kernel modification (a) and without kernel modification (b).

ACKNOWLEDGMENT

This work was in part supported by a grant from ITRC.

6. 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] 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.
[7] 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.
[8] 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.
[9] K. Fukunaga, Introduction to Statistical Pattern Recognition, Second Ed., Academic Press, Boston, 1990.T. Kailath, “The Divergence and Bhattacharyya Distance Measures in Signal Selection,” *IEEE Trans. Commun. Tech.*, COM-15:52–60, 1967.
[10] T. Kailath, “The Divergence and Bhattacharyya Distance Measures in Signal Selection,” *IEEE Trans. Commun. Tech.*, COM-15:52–60, 1967.
[11] 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.
[12] H.G. Barrow and J. Tenenbaum, Recovering Intrinsic Scene Characteristics from Images, Academic press 1978.
[13] E.H. Land and J.J. McCann, Lightness and Retinex Theory, Journal of the Optical Society of America, 61 1-11, (1971).
[14] Andrzej J. Kasinski and Alaa M. Hamdy, *Segmentation based on homomorphic filtering and improved seeded region growing for mobile robots tracking in image sequences*, Machine Graphics & Vision International Journal, 10, 447-466, (2001).
[15] O. Masoud, and N. P. Papanikolopoulos, "A novel method for tracking and counting pedestrians in real-time using a single camera". *IEEE Trans. Vehicular Technology*, 50:1267-1278, 2001.