J. Basic. Appl. Sci. Res., 3(4)596-607, 2013
© 2013, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

A Robust Mean-Shift Tracking Using GMM Background Subtraction

Mohsen Atefian* **and Homayoun Mahdavi-Nasab**

Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Esfahan, Iran

ABSTRACT

Although mean-shift based tracking algorithms are robust in representing the object appearance, they face difficulties under complex conditions such as high similarity of target and background or low contrast. This paper aims at presenting a robust algorithm to track moving objects in image sequences containing such scenes. In order to achieve this goal, Gaussian Mixture Model (GMM) based background subtraction is applied in a mean-shift color histogram framework tracking. In the temporal tracking stage, the object is determined by the Bhattacharyya similarity measure and its position in consecutive frames is obtained using GMM background subtraction. To make the algorithm robust against shape and size variations a natural extension of mean-shift method is used. Experimental results show that the proposed approach has remarkably better robustness and reliability compared to the traditional mean-shift based tracking.

KEYWORDS: Object tracking, Color histogram, Background subtraction, Gaussian Mixture Model, Mean-shift.

1. INTRODUCTION

As one of the most important tasks in intelligent video surveillance, moving object tracking has been an intensive research topic recently [1]. It has a wide range of potential applications as motion based recognition, monitor activities, traffic monitoring, video indexing and human-computer interactions. Many different tracking methods exist, but yet there are not many impeccable techniques which are quick and reliable [2].The difficulty of the issue depends on how an object is defined. Mean-shift algorithm as a reliable and fast tracking technique can overcome computational complexity problems while it performs proper qualified consequences. Fukunaga and Hostetler originally proposed this algorithm for data clustering purpose [3]. It was introduced to image processing community by Cheng [4]. In [5] Comaniciu et al. introduced new segmentation and target tracking scheme using mean-shift. Mean-shift tracking is an iterative gradient based algorithm which tries to find the modes of a probability density function (pdf) for target zone. Original mean-shift procedure cannot match with the shape and size of objects; accordingly, some shape descriptor methods were defined as a solution [6-8]. In [9] an extended mean-shift was introduced, which in addition to estimating the position of a local mode, describes its approximate shape using a covariance matrix. Such a method can adapt to changes in shape and scale of objects.

Mean-shift algorithm uses a color histogram of the object region as a pdf. Although this is a simple and efficient way to find the target location in next frame, color histogram based methods generally lead to failure object tracking when a moving object has a similar color as its background or shows poor contrast. On the other hand, these methods may cause the spatial information of the target to be missed [10].

For better histogram target representation, recently some hybrid methods are developed to overcome the problems. In [11] merging the merits of region-based and contour-based methods resulted in an efficient object tracking in complex environments despite the camera motion. The method is more precise than traditional mean-shift, but is not able to track the target in the mentioned conditions and is also time-consuming. In [1] Kalman filter is employed in mean-shift tracking to estimate the target position. Although resulted in optimum use of the target motion information, it led to wrong object tracking in similar target and background color conditions. Applying particle filter in color-based tracking to solve non-linear and non-Gaussian target tracking is discussed in [12]. Due to the use of color feature, this method loses target in complex scenes. Particle filter is used in a color based algorithm in [13] to minimize the disturbance of background. Although this approach improves feature points matching, the problem of false matching in certain conditions is not solved yet. Han et. al. present a combined tracking algorithm of mean-shift and a double model filter to obtain robust results in abrupt and fast motion scenes [14]. A tracking algorithm is also introduced using mean-shift and grey prediction, which represents object with color and gradient features. This approach avoids the instability of lighting variations and somewhat background similarity [15]. In [16] a fuzzy histogram tracker is presented to reduce noisy interference of the color-based mean-shift tracking. It brings about troubles under low contrast conditions either and needs high computational processes during

Corresponding Author: Mohsen Atefian, Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Esfahan, Iran; Emails: m_atefi@sel.iaun.ac.ir , mahdavinasab@iaun.ac.ir

object tracking. Applying a joint color-texture has been proposed in [10] to present an accurate algorithm in mean-shift tracking, where the problem of missing spatial information remained as an unsolved problem.

This paper aims to intensify the robustness of mean-shift algorithm under critical conditions (e.g. target and background similarity or low contrast). We apply background subtraction which prepares spatial information for color histogram tracking. To model the background, a modified Gaussian Mixture Model (GMM) is adopted, and then an on-line approximation is used to update it. Applying some post-processing operations decreases various interferences of background and noise which leads to better results of foreground detection. Extended mean-shift (EM) is employed to solve the problem of matching shape and size variants. In comparison to traditional color histogram methods, this hybrid algorithm presents more robust tracking; especially under critical conditions.

In summary, the scientific contributions of this paper are:

1. Applying the post-processing operations to eliminate noise effects and reduce the background interferences in output image of background subtraction stage.

2. Incorporating the GMM background subtraction in mean-shift algorithm as an improvement of the tracking robustness technique.

This paper is organized as follows. Section 2 introduces mean-shift tracking and the extended mean-shift algorithm. In section 3 a brief review of GMM based background subtraction and the modified GMM is presented. Section 4 presents the proposed tracking framework. Section 5 is dedicated to experiments and simulation results and section 6 concludes the paper.

2. MEAN-SHIFT OBJECT TRACKING

2.1. Color Histogram Based Object Localizaton

One of the most useful features to demonstrate an object is color and more conventional color based tracking methods employ color histogram to represent the object [17-19]. Histogram based methods use a similarity criterion for target localization. Bhattacharyya coefficient is a similarity metric that uses correlation between target model and candidates in frame sequences.

By normalizing the histogram, a target is charectrized by a pdf which is used to describe the target in the color feature space. In histogram based target representation, the target region is determined by an ellipse or rectangle area which is used as the reference model. Normalized histogram vector is defined by [1]:

$$\vec{q} = \{q_u\}_{u=1\ldots m} \tag{1}$$

$$\sum_{u=1}^{m} q_u = 1 \tag{2}$$

Where q_u represents the uth component of the histogram vector that describes the target. Each target component q_u is demonstrated by:

$$q_u(y) = C\sum_{i=1}^{n} k(\|x_i^*\|^2)\delta[b(x_i^*) - u)] \tag{3}$$

Where $\{x_i^*\}_{i=1\ldots n}$ represents the position of the ith normalized pixel in the target region and $b(x_i^*)$ associates x_i^* to the histogram bin, δ refers to the Kronecker delta function, $k(.)$ is an isotropic kernel function [12] and C is a constant normalizing factor defined as:

$$C = 1 \Big/ \sum_{i=1}^{n} k(\|x_i^*\|^2) \tag{4}$$

For localizing the target in the next frames, the search window moves inside the images. Considering y as a window center, histogram of regions in the window can be calculated and normalized for any y . The obtained probability function is demonstrated by $p(y)$ which represents the target candidate. Assuming $\{x_i\}_{i=1\ldots n_h}$ denotes the pixel positions in a target candidate region centered at y, the following relation for target candidate holds:

$$p(y) = \{p_u(y)\}_{u=1\ldots m} \tag{5}$$

$$\sum_{u=1}^{m} p_u = 1 \tag{6}$$

$$p_u(y) = C_h \sum_{i=1}^{n_h} k(\|\frac{y - x_i}{h}\|^2)\delta[b(x_i) - u)] \tag{7}$$

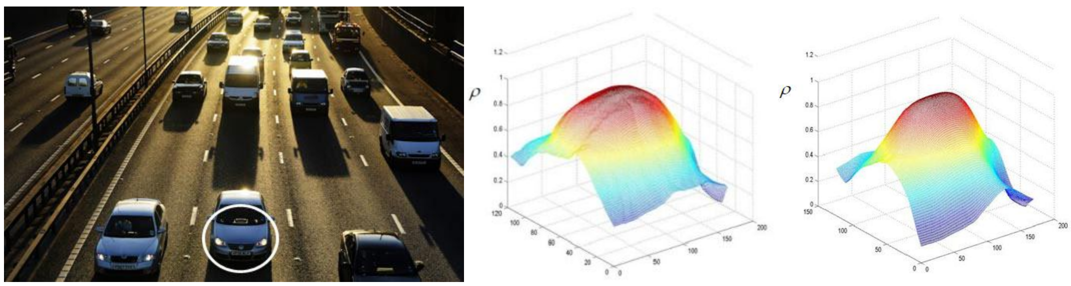
$$C_h = 1 \Big/ \sum_{i=1}^{n_h} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \tag{8}$$

Where h is the bandwidth and C_h is the normalization factor.

The goal of tracking process is to find the best target which has the most similarity to its candidate. The similarity metric based on the Bhattacharyya coefficient is denoted as:

$$\rho(p(y), q) = \sum_{u=1}^{m} \sqrt{p_u(y)} . \sqrt{q_u} \tag{9}$$

A local maximum of ρ indicates the presence of object in the next frame. To find this maximum, gradient based optimization can be applied [20]. Using the kernel mask $k(.)$ improves searching procedure and causes the similarity function to become smooth [1] (See Fig. 1).



(a) (b) (c)

Fig. 1. a) Marked region of image, b) the similarity surface (values of Bhattacharyya criteria) for the marked region, c) The similarity surface of the region after implementation of the kernel $k(.)$

2.2. Mean-shift tracking

To maximize the similarity measure the approximate Taylor expansion around $p_u(y_0)$ is used. This is [1]:

$$\rho(p(y), q) \approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{p(y_0) q_u} + \frac{1}{2} C_h \sum_{i=1}^{n_h} \omega_i k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \tag{10}$$

where

$$\omega_i = \sum_{u=1}^{m} \sqrt{\frac{q_u}{p_u(y_0)}} \delta[b(x_i) - u)] \tag{11}$$

Maximizing $\rho(p(y), q)$ according to (10) is equivalent to maximizing the second term of (10) which is shown to yield [12]:

$$y_1 = \frac{\sum_{i=1}^{n_h} x_i \omega_i g\left(\left\| \frac{y - x_i}{h} \right\|^2\right)}{\sum_{i=1}^{n_h} \omega_i g\left(\left\| \frac{y - x_i}{h} \right\|^2\right)} \tag{12}$$

Where *g*(.) is the derivative of the kernel profile *k*. Choosing an Epanechnikov kernel for $g(.)$ [21], (12) is reduced to:

$$y_1 = \frac{\sum_{i=1}^{n_h} x_i \omega_i}{\sum_{i=1}^{n_h} \omega_i} \tag{13}$$

Fig. 2 shows the brief explanation of color histogram based object localization in flow chart.

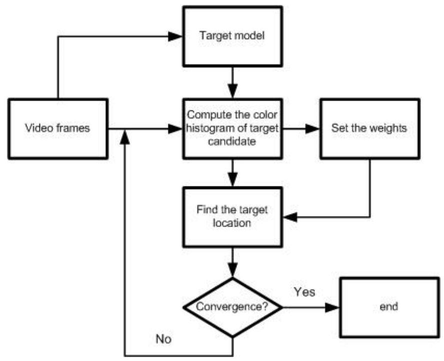


Fig. 2. Flow chart of color histogram based object localization

2.3. Extended Mean-shift

The original mean-shift discussed in section (2.2.) may mislead the tracking when the shape and size of moving object change. An extension of mean-shift is introduced in [9] which claim to be robust against changes of shape and size.

In this method the search window is supposed to be an ellipse with $\vec{\theta}$ as its center. A Gaussian window defined by $\vec{\theta}$ and V (covariance matrix) is applied to the elliptic search area. Suppose $\mathrm{N}(\vec{x}_i,\vec{\theta},V)$ represent the adopted value of window in $\vec{x}_i$ position and vectors $\vec{q},\vec{p}(\vec{\theta})$ are defined as:

$$\mathrm{N}(\vec{x}_i,\vec{\theta},V)=e^{(\vec{x}_i-\vec{\theta})^T V^{-1}(\vec{x}_i-\vec{\theta})} \tag{14}$$

$$q_u=\sum_{i=1}^{N_v}\mathrm{N}(\vec{x}_i,\vec{\theta}^*,V)\delta[b(\vec{x}_i)-u]\ 1\le u\le m \tag{15}$$

$$\vec{p}(\vec{\theta})=\{p_u(\vec{\theta},V)\}_{u=1\ldots m} \tag{16}$$

$$p_u(\vec{\theta})=\sum_{i=1}^{N_v}\mathrm{N}(\vec{x}_i,\vec{\theta},V)\delta[b(\vec{x}_i)-u] \tag{17}$$

Here N_v is the number of pixels surrounded by the ellipse, i describes each pixel number ($1\le i\le N_v$). $b(\vec{x}_i)$ associates $\vec{x}_i$ to the histogram bin and $\vec{\theta}^*$ demonstrates the user defined ellipse center of the target reference region. And

$$\delta[b(\vec{x}_i)-u]=\begin{cases}1 & if\ x_i\ is\ in\ u^{th}\ histogram\ \ bin\\ 0 & else\end{cases} \tag{18}$$

The objective of tracking process is to find the candidate which has the maximum similarity with the reference model. Considering the similarity function:

$$\rho(\vec{p}(\vec{\theta}),\vec{q})=\sum_{u=1}^{m}\sqrt{p_u(\vec{\theta})}.\sqrt{q_u} \tag{19}$$

With the approximate Taylor expansion:

$$\rho(\vec{p}(\vec{\theta}),\vec{q})\approx c_1+c_2\sum_{i=1}^{N_v}\omega_i\mathrm{N}(\vec{x}_i,\vec{\theta},V) \tag{20}$$

Where c_1 and c_2 are supposed to be constant and ω_i is:

$$\omega_i=\sum_{u=1}^{m}\sqrt{\frac{q_u}{p_u(\vec{\theta},V)}}\delta[b(\vec{x}_i)-u)] \tag{21}$$

The following function need to be maximized:

$$f(\vec{\theta},V)=\sum_{i=1}^{N_v}\omega_i\mathrm{N}(\vec{x}_i,\vec{\theta},V) \tag{22}$$

From Jensen inequality [22, pp.25] we have:

$$\log(f(\vec{\theta},V))\ge L(\vec{\theta},r_1,\ldots,r_N)=\sum_{i=1}^{N_v}\log(\frac{\omega_i\mathrm{N}(x_i,\theta,V)}{r_i})^{r_i} \tag{23}$$

Where r_i are arbitrary coefficients and:

$$\sum_{i=1}^{N_v} r_i = 1 \ , r_i \geq 0 \tag{24}$$

Let $\vec{\theta}^{(k)}$ be the obtained ellipse center in k[th] iteration, it can be proved that putting r_i in (23), leads to:

$$r_i = \frac{\omega_i \mathrm{N}(\vec{x}_i, \vec{\theta}^{(k)}, V^{(k)})}{\sum_{i=1}^{N_v} \omega_i \mathrm{N}(\vec{x}_i, \vec{\theta}^{(k)}, V^{(k)})} \tag{25}$$

In this algorithm moving towards large r_i causes the algorithm to converge; maximizing $\mathrm{N}(\vec{\theta}, r_1, ..., r_N)$.To this end it is required to maximize the following function:

$$g(\vec{\theta}, V) = \sum_{i=1}^{N_v} r_i \log \mathrm{N}(\vec{x}_i, \vec{\theta}, V) \tag{26}$$

Putting $\frac{\partial}{\partial \vec{\theta}} g(\vec{\theta}, V) = 0$ the following relation results:

$$\vec{\theta}^{(k+1)} = \sum_{i=1}^{N_v} r_i \vec{x}_i = \frac{\sum_{i=1}^{N_v} x_i \omega_i \mathrm{N}(\vec{x}_i, \vec{\theta}^{(k)}, V^{(k)})}{\sum_{i=1}^{N_v} \omega_i \mathrm{N}(\vec{x}_i, \vec{\theta}^{(k)}, V^{(k)})} \tag{27}$$

Practically an end criterion for the algorithm is required to prevent from falling in a loop.

3. BACKGROUND SUBTRACTION

A commonly used method for motion detection in digital video sequences is background subtraction. As a simple classification, this process can be divided to adaptive and non-adaptive. Manual selection techniques, voting pixel value methods and mean-shift algorithm are non-adaptive techniques which cause noise accumulation as an imperfect property. Adaptive techniques include images averaging methods, Kalman filter, adaptive Gaussian estimation, Gaussian Mixture Model and so on [2, 23]. Among adaptive techniques, use of temporal averaging of images leads to simple and fast tracking. However, these do not work well in scenes containing many moving objects [24]. Furthermore, these methods are unable to describe multi-view areas. Kalman filter based methods [25] also offer partial and incomplete responses. GMM based methods are capable multi-view background subtraction techniques which are also robust against illumination variations and small camera motions [24].

Russell and Friedman modeled each pixel value by a parametric adaptive model with a mixture of the three Gaussian distributions [26]. They also developed primary online functions to update distribution parameters. Koller et al. used Kalman filter for illumination changes compensation in each pixel value [27]. Although their model is resistant against intensity changes, there are deficiencies in evaluation of this algorithm when a new object is added or removed from the background. To tackle this problem Stauffer and Grimson proposed an adaptive multi-colored background model for each pixel value [28-30]. This model obviates repetitive short movements of background elements. The standard GMM update equations have been enhanced in [31] to increase system matching speed.

3.1. Standard GMM Based Background Subtraction

The method described here is proposed by Stauffer and Grimson [28-30].From a mathematical viewpoint, pixels of consecutive frames in a dynamic scene without moving objects have a regular behavior which can be stated by a stochastic model. Each pixel is considered as an independent statistical process which its observed intensity in previous n frames is recorded. The record is then optimally fit to a mixture of K Gaussians. For each pixel, more than one distribution can be stated over the time.

Since each pixel is modeled by K Gaussians, the probability of an observed pixel with intensity value $\ell(t)$ at time t is modeled as:

$$p(\ell(t)) = \sum_{i=1}^{k} w_{i,t} .\eta(\ell(t), \mu_{i,t}, \Sigma_{i,t}) \tag{28}$$

Where $w_{i,t}$ is the i[th] Gaussian mixture weight and $\eta(\ell(t), \mu_{i,t}, \Sigma_{i,t})$ are the component Gaussian densities expressed by:

$$\eta(\ell(t), \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{(2\pi)^{n/2} |\Sigma_{i,t}|^{n/2}} \exp(-\frac{1}{2}(\ell(t) - \mu_{i,t})^T \Sigma_{i,t}^{-1} (\ell(t) - \mu_{i,t})) \tag{29}$$

With mean $\mu_{i,t}$ and covariance matrix $\sum_{i,t} = \sigma_{i,t}^2 I$. The weight parameter $w_{i,t}$ determines the time duration that i[th] distribution exists in the background. The weights are positive and their sum is one. The K distributions are ordered based on the fitness parameter w_k / σ_k and the number of active Gaussian components is calculated assuming that the background includes B colors with the most probability. This is obtained as follows:

$$B = \arg_b \min(\sum_{i=1}^{b} w_{i,t} > T_{thr}) \tag{30}$$

T_{thr} is the minimum prior probability that the background is in the scene. Dedicating a small amount of threshold, causes Gaussian components with the most probability to be considered as the background, whereas large amounts of T_{thr} adopts more components for the background, so the leaves, flags, etc. may be modeled.

Background subtraction process is performed by marking pixels that are more than 2.5 SD away from any of the B distributions as the foreground moving objects. Let define b_i as the i[th] binary output image which contains foregrounds detached from backgrounds. Therefore it can be followed as:

$$b_i = \begin{cases} 1 & for\ pixels\ 2.5\ SD\ from\ the\ distributions \\ 0 & else \end{cases} \tag{31}$$

The first Gaussian component that matches the test value is updated by the following equation.

$$w_{i,t+1} = (1-\alpha) w_{i,t} + \alpha M_{i,t} \tag{32}$$

where

$$M_{i,t} = \begin{cases} 1 & if\ w_k\ belongs\ to\ k^{th}\ \ Gaussian\ component \\ 0 & else \end{cases} \tag{33}$$

Where α is the learning rate, which sets the time constant of the speed at which the distribution parameters change.

Other parameter updating equations at time t are:

$$\mu_{i,t+1} = (1-\rho)\mu_{i,t} + \rho \ell(t) \tag{34}$$

$$\sigma_{i,t+1}^2 = (1-\rho)\sigma_{i,t}^2 + \rho(\ell(t) - \mu_{i,t})^T (\ell(t) - \mu_{i,t}) \tag{35}$$

$$\rho = \alpha \eta(\ell(t), \mu_{i,t}, \Sigma_{i,t}) \tag{36}$$

3.2. Modified GMM background subtraction

The above method has drawbacks under a variety of special conditions [32]. First, if the initial pixel value belongs to the foreground, there would be only one distribution with the weights equal unity. If next pixels belong to the background with the same color it takes $\log_{(1-\alpha)}(T_{thr})$ frames until adding this pixel to the background. For example if at least for 60 percent of the time, the pixel is considered as belonging to the background (with learning rate of 0.002), 255 frames are required. Second problem occurs when the likelihood factor ρ has a small value. This causes slow parameter adjustment; leading to low precision at primary frames. The third problem is that this method does not discriminate between background elements and their shadows.

To overcome the difficulties, a modification scheme is suggested in [32]. In this method estimating of the GMM is done by expected sufficient statistics update. The update equations provide an estimate at the beginning; before all T recent samples can be collected. To this end a priority of T recent frames of each Gaussian distribution is maintained to define the frequency of adopting desired distribution as an adapted distribution. It is clear that more presence in T last frames causes more chance in choosing that distribution as the matched distribution. Considering theses points, updating equations of the distribution parameters change as follows [32]:

$$\hat{w}_{i,t+1} = \hat{w}_{i,t} + \frac{1}{T}(M_{i,t} - \hat{w}_{i,t}) \tag{37}$$

$$\hat{\mu}_{i,t+1} = \hat{\mu}_{i,t} + \frac{M_{i,t}}{\sum_{\tau=t-T}^{t} M_{i,\tau}} (\ell(t) - \hat{\mu}_{i,t}) \tag{38}$$

$$\hat{\Sigma}_{i,t+1} = \hat{\Sigma}_{i,t} + \frac{M_{i,t}}{\sum_{\tau=t-T}^{t} M_{i,\tau}} ((\ell(t) - \hat{\mu}_{i,t})^T (\ell(t) - \hat{\mu}_{i,t}) - \hat{\Sigma}_{i,t}) \tag{39}$$

Where $\hat{w}_{i,t}, \hat{\mu}_{i,t}, \hat{\Sigma}_{i,t}$ are the new weights, mean and covariance at time t, respectively.

If the ith Gaussian is an adapted one in frame τ, $M_{i,\tau}=1$ otherwise it is zero. In this notation $\sum_{\tau=t-T}^{t} M_{i,\tau}$ represents the history of ith Gaussian. In the modified equations, the coefficient ρ, is replaced by $1 \Big/ \sum_{\tau=t-T}^{t} M_{i,\tau}$ to increase the Gaussian parameters adjustment rate. It should be pointed out that after updating, weights will be renormalized to maintain their sum equal to one.

4. THE PROPOSED ALGORITHM

To achieve more robust tracking, we propose employing the modified GMM background subtraction in the mean-shift framework. This will preserve and intensify the target spatial motion information that may be lost under critical conditions otherwise. Implementation of the background subtraction yields a binary motion information image (BMI) which discriminates non-stationary pixels. To reach better results some post-processing operations are proposed to be applied on the BMI.

4.1. Foreground Refining

The post-processing is done in 3 steps as follows.

a) Noise Elimination. Three different filters are used to eliminate noise: median filter to remove salt and pepper noise, morphological closing filter for filling holes and opening operation to remove small regions [33].

b) Shade Reduction. Shadows change the color value of pixels. Since GMM clustering is based on the difference of current pixel color value and the mean Gaussian distribution, it is probable that some shade pixels mistakenly be considered as background pixels. Therefore it is necessary to examine background pixels once again to prevent wrong labeled pixels.

Since a shadow does not usually affect the color component and generally makes changes to illumination intensity, most shade reduction algorithms use HSI color space [34]. However, as RGB to HSI color space transform is often computationally complex, here we use the method proposed in [35] to reduce time consuming. For the whole pixels marked as background, the distance of the pixel color value with the Gaussians mean is calculated as:

$$d_i = | h_i(t)\,\hat{\mu}_i - \ell(t) |^2 \tag{40}$$

Where:

$$h_i(t) = \frac{\ell(t)\,\hat{\mu}_i^{\,T}}{|\hat{\mu}_i|^2} \tag{41}$$

Each pixel is considered as a shadow pixel if:

$$d_i < T_{sh,i} \tag{42}$$

Where $T_{sh,i}$ is a threshold related to the Gaussian component variance. This is noticeable that these computations are performed only for background pixels; hence the computational complexity of the method remains acceptable.

c) Connected Component Analysis. Completing the above steps, the detected moving foreground objects are not necessarily connected. For a distinct object to be distinguished, a connected component labeling is done to assign each region a unique label. Based on the label, each region is processed to extract a number of features [36]. Classifying will be done regarding these features (for example, area, centre of gravity, bounding box, etc). Pixels with wrong label will be removed and the object will be detached as a uniform silhouette.

Fig. 2 shows how the post-processing operations eliminate unwanted regions.

(a) (b) (c)

Fig. 2. a) Original pets2001 image, b) BMI before the post-processing, c) BMI after the post-processing

4.2. Tracking using binary motion information

In mean-shift object tracking, the surrounding ellipse moves toward the desired target location. According to

(27), $\vec{\theta}$ moves toward pixels with large r_i . This led us to introduce a new object tracking algorithm. Here the goal is to propel $\vec{\theta}$ towards maximum non-stationary pixels. Considering the refined BMI pixels as $\hat{b}_i$, the following rule conveys $\vec{\theta}^{(k+1)}$ toward non-stationary pixels.

$$\vec{\theta}^{(k+1)} = \frac{\sum_{i=1}^{N_v} \hat{b}_i \vec{x}_i}{\sum_{i=1}^{N_v} \hat{b}_i} \tag{43}$$

Based on (43) when the object does not move, the target location is remained unchanged: $\vec{\theta}^{k+1} = \sum_{i=1}^{N_v} x_i \Big/ N_v = \vec{\theta}^{k}$. Otherwise, it departs towards the moving pixels.

For using motion and color information simultaneously (27) is modified to follows:

$$\vec{\theta}^{(k+1)} = \sum_{i=1}^{N_v} r_i \vec{x}_i + \left(\frac{\sum_{i=1}^{N_v} \hat{b}_i \vec{x}_i}{\sum_{i=1}^{N_v} \hat{b}_i} - \vec{\theta}^{(k)}\right) \tag{44}$$

The above rule can be justified based on the hybrid motion detection process where the second term in (44) is a modifying factor based on (43) to compensate the mean-shift. When the object is moved in next frame, the BMI pixel values which are relevant to the considered object are one. The first term in (44) moves $\vec{\theta}$ towards the target position based on the target color feature and the second term adds the difference of non-stationary pixels location with previous location of the target. This procedure propels $\vec{\theta}$ towards the target location. It should be pointed out that when the object has not moved, the first term of (44) does not change the ellipse and due to the equality of $\hat{b}_i$ s the second term of the relation does not change either.

A summary of the algorithm follows:

Considering the reference target model $\vec{q}$ and its location $\theta^{(0)}$ in the previous frame:

1- Put the target location in previous frame as an initial value for $\vec{\theta}$ in the current frame. This is shown by $\vec{\theta}^{(0)}$ (k=0 for the first step).
2- Compute $\vec{p}(\vec{\theta}^{(k)})$ for ellipse search window (normalized color histogram for inner pixels of the ellipse).
3- Calculate ω_i applying (21).
4- Calculate r_i by means of (25).
5- Implement GMM based background subtraction and get the BMI Image (b_i) using (31).
6- Refine the BMI applying the post-processing operations to get $\hat{b}_i$.
7- Obtain $\vec{\theta}^{(k+1)}$ using (44).
8- If the algorithm converges or reaches the end criterion, stop. Otherwise put $k \leftarrow k+1$ and return to the step 2.

5. EXPERIMENTAL RESULTS

Two representative video sequences are used to evaluate the proposed method in comparison with the conventional mean-shift based tracking algorithms**.** To examine the robustness of the method, it is applied to a number of sequences containing critical scenes. In all the experiments the RGB feature space is used and to decrease the computational burden it is quantized to $8 \times 8 \times 8$ bins, which means the color histogram can be displayed by a vector of length 512. Simulation results are generated using Matlab R2009b.

The first experiment uses the pets2001 sequence. A cyclist moves toward a tree where its color becomes similar to the background. As it is illustrated in Fig. 4 the mean-shift algorithm cannot find accurate location of the cyclist in the next frame. Whereas implementation of the suggested algorithm tracks the target properly (Fig. 5).

A soccer sequence is used for the second experiment, wherein the target player exhibits obvious illumination changes. The experimental results show the precise performance of the proposed method in comparison to the traditional mean-shift (Fig. 6 and 7).

(a) (b)

Fig.4. Target tracking in pets2001 sequence using conventional mean-shift algorithm

(a) (b)

Fig. 5. Target tracking in pets2001 sequence using the proposed algorithm

(a) (b) (c) (d)

Fig.6. Tracking results of the soccer sequence using conventional mean-shift

(a) (b) (c) (d)

Fig. 7. Tracking results of the soccer sequence using the proposed method

We also validate the performance of the proposed approach to track targets in normal scenes. Tracking results for a vehicle and a walking man using the proposed method are shown in Figures 8 and 9, respectively. In both cases the proposed algorithm tracks the target successfully.

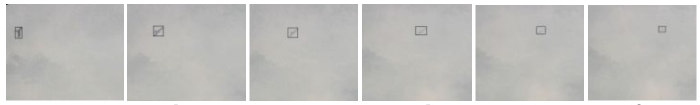
(a) (b) (c) (d)

Fig. 8. Vehicle tracking of pets2001 sequence based on the proposed algorithm

(a) (b) (c) (d)

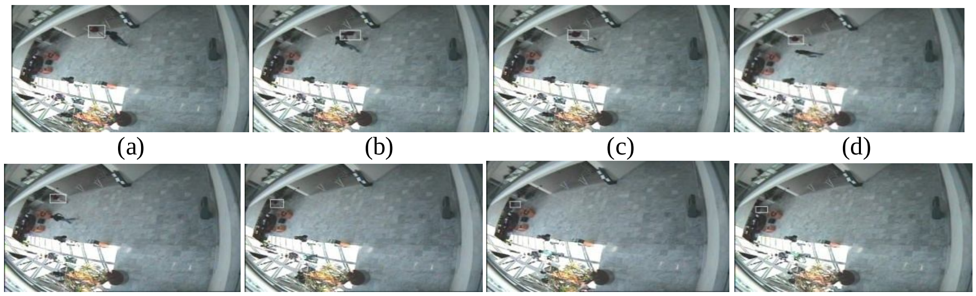
Fig. 9. Tracking a walking man in pets2001 sequence based on the proposed algorithm

The tracking system is also tested under more critical conditions. In Fig. 10 a flying bird movement is recorded while in each subsequent frame the bird appearance fading is increased. Fig. 11 shows tracking results on a CAVIAR dataset in which a person was tracked correctly even though color information is lost frame by frame. We also tested the proposed algorithm at night with low background brightness. The proposed algorithm is successful again, as shown in Fig. 12.



(a) (b) (c) (d) (e) (f)

Fig. 10. Tracking results of a flying bird using the proposed algorithm



(a) (b) (c) (d)

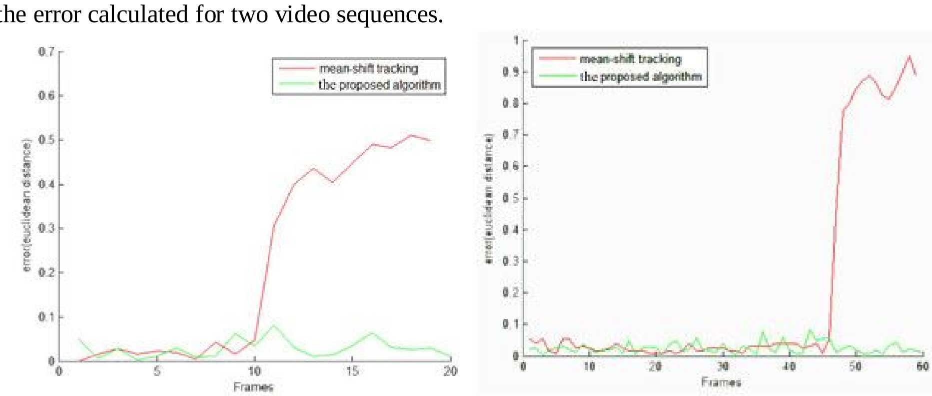
(e) (f) (g) (h)

Fig. 11. The proposed method tracking in CAVIAR sequence

(a) (b) (c) (d) (e)

Fig. 12. Tracking results in"AVSS_PV_NIGHT" video using the proposed approach

To evaluate the performance of the proposed method the Euclidean distance (error)[1] of the estimated and real target locations are calculated for both the conventional mean-shift and the proposed algorithm. Fig. 13 depicts the error calculated for two video sequences.



(a) (b)

Fig. 13. Comparison of the proposed and traditional mean-shift tracking on basis of the distance error, a) soccer sequence, b) CAVIAR sequence

Considering Fig. 13 in the traditional mean-shift algorithm when a target color tends to its background a suddenly mutation occurs in the observed error which leads to missing the target. This error is negligible for the proposed approach. This demonstrates the effectiveness of the approach for tackling common challenges such as target and background color similarity. Overall this hybrid method overcomes main challenges of popular mean-shift algorithm and can be used as a robust tracker in dynamic scenes.

6. CONCLUSION

During recent years mean-shift algorithm using color information has been one of the most commonly used approaches for object tracking. Though this algorithm contains benefits, accurate tracking of objects in critical cases of background color resemblance, illumination changes and low contrast often fails. In this paper, a GMM background subtraction based extended mean-shift algorithm was introduced to tackle these problems. The proposed approach prepares spatial information of the moving object which turns the traditional mean-shift to a robust tracking algorithm. Based on this approach, the mutation of tracking error (TE) in critical condidition occurances has a significant downturn. As a visual prespective, various experimental results show that the proposed method accurately localize the target in the event of target and background color correlation, and it is notably robust and capable of tracking objects efficiently under severe conditions.

---

[1] The performance evaluation of the proposed approach is on basis of Euclidean distance between the center of target window with its real location which detects resulted sharp changes in loss of tracking..

REFERENCES

1. Comaniciu, D., V. Rmesh, and P. Meer, 2003. Kernel-Based Object Tracking: IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 5, pp. 564–577.
2. Magee, D., 2004. Tracking Multiple Vehicles Using Foreground Background and Motion Models: Image and Vision Computing, Vol. 22, pp. 143-155.
3. Fukunaga, K. and L. D. Hostetler, 1975. The Estimation of The Gradient of a Density Function, With Applications in Pattern Recognition, IEEE Trans. Inform. Th. 21(1) 32–40.
4. Cheng, Y.,1995. Mean Shift, Mode Seeking, and Clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 8, pp. 790-799, doi:10.1109/34.40056.
5. Comaniciu, D. and P. Meer, 2002. Mean Shift: A Robust Approach Toward Feature Space Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5), 603-619,.
6. Comaniciu, D. V.Ramesh, and P.Meer, 2000. Real-time tracking of non-rigid Objects Using Mean-shift", Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2:142.149.
7. Bradski, G.R, 1998. Computer Vision Face Tracking as a Component of a Perceptual User Interface, Proc. IEEE Workshop on Applications of Computer Vision, pages 214.219.
8. Collins, R., 2003. Mean-shift Blob Tracking Through Scale Space, Proc. IEEE Conf. on Computer Vision and Pattern Recognition.
9. Zivkovic, Z. and B. Krose, 2004. An EM-like Algorithm for Color-histogram-based Object Tracking, Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04) 1063-6919/04 $20.00.
10. Zhang, L., D. Zhang, C. Wu, 2009. Robust Object Tracking Using Joint Color-texture Histogram, International Journal of Pattern Recognition And Artificial Intelligence, Vol. 23, No. 7, 1245-1263.
11. Chen, Q., Q.S. Sun, P.A. Heng, D.S. Xia, 2003. Two-stage Object Tracking Method Based on Kernel and Active Contour", IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 564-575.
12. Nummiaro, K., E. Koller-Meier, L. Van Gool, 2011. An Adaptive Color-based Particle Filter, Image and Vision Computing, vol. 21, pp. 99-110.
13. Chen, L., W. Li, W. Yin, 2011. Joint Feature Points Correspondences and Color Similarity for Robust Object Tracking, International Conference on Multimedia Technology (ICMT), 978-1-61284-771-9.
14. Ju, M., C. Ouyang, H.S. Chang, 2010. Mean-shift Tracking Using Fuzzy Color Histogram, International Conference On Machine Learning And Cybernetics.
15. Han, R., Z. Jing, and G. Xiao. Probabilistic Motion Switch Tracking Method Based on Mean shift and Double Model Filters. In Proc. International Symposium on Neural Networks, volume 2, pages 705–714, Nanjing, China, June 2007.
16. Wang, Y., J. Zhang, L. Wu and Z. Zhou, 2010. Mean Shift Tracking Algorithm Based on Multi-Feature Space and Grey Model, Journal of Computational Information Systems 6:11 (2010), pp: 3731-3739.
17. Dash, P. P., D. Patra, S. K. Mishra, J. Sethi, 2012. Kernel Based Object Tracking Using Color Histogram Technique, International Journal of Electronics and Electrical Engineering, Volume 2.
18. Hu, W., T. Tan, L. Wang, S. Maybank, 2004. A Survey on Visual Surveillance of Object Motion and Behaviors, IEEE Transactions on Systems, Man, and Cybernetics, Part c: Applications and Reviews, Vol. 34, No. 3.
19. Swain, M. and D. Ballard. , 1991. Color Indexing, Intl. J. of Computer Vision, 7(1):11.32.
20. Bascle, B. and R. Deriche, 1995. Region Tracking Through Image Sequences, in Proc. 5th Intl. Conf. on Computer Vision, Cambridge, MA, pp. 302–307.
21. Scott D.W., 1992. Multivariate density estimation. Wiley.
22. Cover, T. and J. Thomas, 1991. Elements of information theory, John Wiley & Sons, New York.
23. Wren, C.R., A. Azarbayejani, T. Darrell, A.P. Pentland, 1997. Real-time Tracking of the Human Body, IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 19, No. 7.
24. Stauffer, C., W.L. Grimson, 2000. Learning Patterns of Activity Using Real-time Tracking, IEEE

Transactions On Pattern Analysis And Machine Intelligence, Volume 22.
25. Suliman, C., C. Cruseru, F. Moldoveanu, 2010. Kalman Filter Based Tracking in an Video Surveillance System, International Conference on Development and Application System, Suceava, Romania, Volume 10.
26. Friedman, N., S. Russell, , 1997. Image Segmentation in Video Sequences: A Probabilistic Approach, 13th Conf. on Uncertainty in AI.
27. Koller, D. et al., 1994. Towards Robust Automatic Traffic Scene Analysis in Real-time, 33rd IEEE Conf. on Decision and Control, Part vol.4.
28. Grimson, W., C. Stauffer, R. Romano, L. Lee, 1998. Using Adaptive Tracking to Classify and Monitor Activities in a Site, IEEE CVPR.
29. Stauffer, C., W. Grimson, 1999. Adaptive Background Mixture Models for Real-time Tracking, IEEE Conf. on CVPR.
30. Stauffer, C. W. Grimson, 2000. Learning Patterns of Activity Using Real-time Tracking, IEEE Transactions on PAMI, 22(8): p. 747-757.
31. Lee, D. S., 2005. Effective Gaussian Mixture Learning for Video Background Subtraction, IEEE Transactions on PAMI, 27 (5), 827–832.
32. KaewTraKulPong, P. and R. Bowden, 2001. An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection, In Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems.
33. Gonzalez and Woods, 2002. Digital image processing, Second Edition, Prentice-Hall.
34. Birchfield, S., 1998. Elliptical Head Tracking Using Intensity Gradients and Color Histograms, IEEE CVPR.
35. Horprasert, T., D. Harwood, L.S. Davis, 1999. A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection. in IEEE ICCV' 99 Frame-Rate Workshop.
36. Bailey, D.G., C.T. Johnston, 2007. Single Pass Connected Components Analysis, Proceedings of Image and Vision Computing, pp. 282–287, Hamilton, New Zealand.