Incorporating Gaussian Mixture Models into Mean Shift Algorithm for Non-Rigid Object Tracking

Guo Jiayan, David Leong, Jonathan Siang, Vikram Bahl
School of Infocomm, Republic Polytechnic, 9 Woodlands Avenue 9, Singapore 738964
{guo_jiayan, david_leong, jonathan_siang, vikram_bahl}@rp.edu.sg

Abstract
In object tracking, the shape of the target object is often represented by some primitive geometric shapes such as a rectangle or ellipse. Such shapes may roughly represent simple rigid objects. When applied on non-rigid objects, some irrelevant background information may be included into the target model that could lead to tracking failure. Mean shift tracking algorithm with a technique called background-weighted histogram (BWH) is often used to decrease the background information. Unfortunately, this technique may wrongly reduce the weights on the target object features. This would result in the inability to distinguish the target object from the background. The BWH technique is also incapable of suitably estimating changes to the scale and orientation of the target object. In order to address such problems, we propose an approach to enhance the mean shift tracking algorithm by employing Gaussian Mixture Models (GMMs). Given a region of interest selected by the users, our approach first employs a level set segmentation method to separate the target object from its surrounding background. Based on the fundamental object and background, the GMMs are learned by utilizing the color histograms as features. In order to reduce the background information of subsequence frames, the GMMs are used to adjust the weight of the object features before they are incorporated into the mean shift algorithm. Our approach would also exploit the moment of the weights to determine the scale and orientation changes to the target object. We have demonstrated the effectiveness of our approach on both synthetic and challenging real-world video sequences.

Keywords: Non-Rigid Object Tracking, Mean Shift, Gaussian Mixture Models, Image Segmentation, Level Set Evolution

1. Introduction
Object tracking is the task of estimating the trajectory of a moving object throughout the video. Additionally, it could also estimate the object-centric information, for example, orientation, area and the shape of the object [1]. Object tracking can be challenging due to the following difficulties, such as object appearance change, non-rigid object deformation, illumination change, viewpoint variation, target object partially or fully occluded by other objects, background clutter, etc. A number of algorithms have been proposed to overcome these difficulties. Among various object tracking algorithms, the mean shift algorithm is one of the most well-known tracking techniques. This is because of its simplicity and efficacy.

Mean shift is a non-parametric statistical technique used to find local maxima in probability distributions. It was initially developed by Fukunaga and Hostetler [2] for cluster analysis, and was later introduced into the field of image processing by Cheng [3]. Subsequently, the mean shift algorithm was applied successfully to object tracking by Comaniciu et al. [4]. In object tracking, the mean shift algorithm maximizes the appearance similarity iteratively by comparing the histograms of the target object with target candidate - the window around the hypothesized object located in the next frame. In spite of its advantages, the traditional mean shift tracking algorithm has two main drawbacks. The first problem is the use of radial symmetric kernel to represent the target object. Since most objects are of anisotropic shapes, the symmetric kernel with isotropic shape is insufficient to represent the object shape. For example, symmetric kernel may roughly represent simple objects, e.g. balls. While it is applied on the non-rigid object in complex shape, e.g. human body, animals, etc., some irrelevant background information may be included into the target model, which could cause tracking failure. The

* This work was supported by grant MOE2011-TIF-G-019 from the Ministry of Education, Singapore.
The second drawback is the fixed scale of the kernel or the constant kernel bandwidth. The kernel with fixed scale is unable to adapt to the scale and orientation changes of the object. That is, it is unable to estimate the scale and orientation changes of the target object. In order to achieve a reliable tracking result of such objects, the scale and orientation adaptive kernel is desirable.

To address the first problem, the background-weighted histogram (BWH) [5] was proposed to reduce the background influence in target representation and localization. Specifically, it attempts to decrease the probability of the prominent background features in the target and candidate models. Since this idea is reasonable and intuitive, some other works [6, 7, 8] have employed a similar idea to their mean shift tracking. However, recent research by Ning et al. [9] proved that the mean shift tracking algorithm with BWH in [5] failed to decrease the weights of prominent background features. The mean shift tracking with BHW is identical to the mean shift tracking with usual target representation. They proposed a corrected background-weighted histogram (CBWH) to reduce the interference of salient background features. CBWH will transform only the target model, but not the candidate model. It also derives a new formula for computing the pixel weights in the candidate model. However, CBWH does not update the background histogram and it also overlooks the motion information. A more sophisticated method of using asymmetric kernel to approximately represent the shape of the target model has been proposed in [10]. This method has only been tested on tracking cars. More experiment should be carried out to verify its effectiveness, especially for those objects with more complex shape.

To attempt the second problem, some mean shift based methods were presented in [11-15]. They are not only adapting the scale, but also the orientation of the kernel. Bradski [11] further modified the mean shift tracking algorithm and developed the Continuously Adaptive Mean Shift (CAMSHIFT) algorithm. The CAMSHIFT algorithm utilizes the first and second order moments of the weight image determined by target model to estimate the object scale and orientation. Although it is not robust, it could handle various types of object movements in real time. By exploring the relativity of the weight image and the Bhattacharyya coefficient between the target model and candidate model, Ning et al. [12] proposed an alternative method to determine the scale and orientation of the target object. Zivkovic and Krose [13] employed the EM algorithm to estimate the position and the covariance matrix that describes the shape. Collins [14] adopted scale space theory [15] to estimate the scale of the target. Unfortunately, it cannot handle the rotation changes of the target, and the computational cost is also very expensive.

In this paper we propose an approach to enhance the mean shift tracking algorithm by using Gaussian Mixture Models (GMMs). Given a region of interest selected by the users, we improve the target object representation by employing a level set segmentation method to separate the target object from the surrounding background. Based on the segmented object and background, we use the GMMs to model the color histograms of the target object and the background. We then incorporate the GMMs into the mean shift algorithm to tune the weights on the object features so as to reduce the background information on the subsequent frames. On the other hand, we exploit moments of the weights to calculate the scale and orientation changes of the target object. A reliable tracking result can be achieved by adapting to the object with changing scale and orientation.

Our approach outperforms other existing methods in three ways. Firstly, unlike common tracking methods that adopt a simple shape for the target object, we achieve a more detailed representation of the shape by utilizing a level sets segmentation method to separate the target object from the surrounding background. Secondly, the mean shift algorithm is enhanced by the Gaussian Mixture Models (GMMs). Based on the segmented target object and background, we model the target object and background in the first frame using GMMs. We consider this mixture as a weighting function for the calculation of the color histogram in the next frames. Thus, the incorporation of the GMMs into the mean shift algorithm can tune the weights on the object features. Ultimately, the background information on the subsequent frames can be effectively reduced by enhancing the weight of the target object and decreasing the probability of the background features. Thirdly, our approach is also capable of adapting to the scale and orientation change of the object, and thus achieving better tracking performance.

The rest of the paper is organized as follows: Section 2 provides a brief overview of the classical mean shift tracking algorithm with background-weighted histogram. The details of our approach are elaborated in Section 3. Section 4 reveals the experimental results. In this section, we compared our
approach against three well-known tracking algorithms. We also evaluate the results we have obtained from the comparisons. We conclude this paper in Section 5.

2. Mean shift tracking algorithm and background-weighted histogram

The mean shift tracking algorithm is a target representation and localization algorithm aiming to localize the target object throughout the video frames. The algorithm represents the reference target model and the target candidate in the next frame by their feature histograms, i.e., the color histograms. It subsequently regularizes the target representations by spatially masking with an isotropic kernel, to smoothen the similarity functions. Since the spatially-smooth similarity functions are suitable for gradient-based optimization, the target localization problem is then formulated as a search in the basin of attraction for the local maxima [5]. With the similarity measure being expressed by a metric based on the Bhattacharyya coefficient, the mean shift procedure is performed to find the target candidate that is most similar to the given target model [4]. The basic mean shift tracking algorithm can be further extended with the background-weighted histogram by integrating the background information. In the next subsections, we illustrate three components of the mean shift tracking algorithm, namely the target representation, target localization, and its extension to background-weighted histogram.

2.1. Target representation

In a tracking scenario, a target object is usually defined as an ellipsoidal region or a rectangle surrounding a region of interest in the image. In order to characterize the target object, the reference target model is chosen to be the color density distribution of the target object. The target model can be approximated by a discrete color histogram of $m$ bins, $\hat{q} = \{\hat{q}_u\}_{u=1}^m \sum_{u=1}^m \hat{q}_u = 1$, with $\hat{q}_u$ being the $u$-th bin. Such discrete color histogram not only satisfies the low computational cost imposed by real-time processing, but also being robust to partial occlusions and invariant to scaling and rotation. At this point, we assume the target object is represented by an ellipsoidal region in the image. In order to form the histogram, all the pixels inside the ellipse are to be taken into account. Due to the fact that the ellipse may contain both target object pixels and background pixels, a convex and monotonically decreasing isotropic kernel with profile $k(x)$, $k : [0, \infty) \rightarrow \mathbb{R}$ is applied to every pixel so as to assign larger weights to pixels near the center of the ellipse. As peripheral pixels are often affected by the occlusions or interference from the background, they are less reliable than those pixels nearer the center. In reducing the influence of different target dimensions, all targets have to be normalized to a unit circle. This can be achieved by dividing the pixel’s coordinates with the ellipse’s semi-axes lengths $h_x$ and $h_y$. The center of the ellipse is assumed to be at the origin of the axes. Let $\{x_i\}_{i=1}^n$ be the normalized pixel location of the target model. The probability of the feature $u (u=1,2,\ldots,m)$ in the target model is computed as

$$\hat{q}_u = C \sum_{i=1}^n k(||x_i^*||^2) \delta[b(x_i^*) - u],$$

(1)

where $\delta$ is the Kronecker delta function, and $b(x_i^*)$ is the histogram bin associated with the pixel location $x_i^*$. The normalization constant $C$ is represented by $C = 1/\sum_{i=1}^n k(||x_i^*||^2)$.

In the subsequent frame, a target candidate is centered at the normalized spatial location $y$. Let $\{y_i\}_{i=1}^n$ be the normalized pixel locations in the target candidate region. Similarly, the probability density function of the target candidate is also approximated by an $m$-bin discrete color histogram, $\hat{p}(y) = \{\hat{p}_u(y)\}_{u=1}^m \sum_{u=1}^m \hat{p}_u = 1$. Using the same kernel profile $k(x)$, but with bandwidth $h$, the probability of the feature $u (u=1,2,\ldots,m)$ in the target candidate model is given by

$$\hat{p}_u(y) = C u \sum_{i=1}^n k \left( \frac{y - x_i}{h} \right)^2 \delta[b(x_i) - u],$$

(2)
where \( C_h = \frac{1}{\sum_{i=1}^{n_h} k \left( \frac{\|y - x_i\|}{h} \right) ^2 } \) is the normalization constant.

### 2.2. Target localization

The distance between target model \( \hat{q} \) and target candidate model \( \hat{p}(y) \) is defined as

\[
d(y) = \sqrt{1 - \rho(\hat{p}(y), \hat{q})},
\]

where

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

is the similarity function of a metric based on the Bhattacharyya coefficient.

To locate the target in the current frame, the distance in (3) should be minimized, which is equivalent to maximizing the Bhattacharyya coefficient \( \rho(y) \) in (4). The search for the new target location \( \hat{y}_i \) in the current frame starts at the location \( \hat{y}_o \), which is the target location in the previous frame. The probabilities \( \{\hat{p}_u(\hat{y}_o)\}_{u=1 \ldots m} \) of the target candidate at location \( \hat{y}_o \) in the current frame are computed and using linear Taylor approximation of (4), we obtain

\[
\rho(\hat{p}(y), \hat{q}) \approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{\hat{p}_u(\hat{y}_o) \hat{q}_u} + \frac{C_h}{2} \sum_{i=1}^{n_h} w_i k \left( \frac{\|y - x_i\|^2}{h} \right)
\]

where

\[
w_i = \sum_{u=1}^{m} \delta [b(x_i) - u] \sqrt{\hat{p}_u(\hat{y}_o) \hat{q}_u}.
\]

Since the first term of (5) is independent of \( y \), the second term of (5) has to be maximized. The mean shift algorithm is employed to find the local maximum by iteratively moving the current location \( \hat{y}_o \) to the new location \( \hat{y}_i \). The new target position \( \hat{y}_i \) is calculated to be a weighted sum of pixels contributing to the model.

\[
\hat{y}_i = \sum_{i=1}^{n_h} x_i w_i g \left( \frac{\|\hat{y}_o - x_i\|}{h} \right) / \sum_{i=1}^{n_h} w_i g \left( \frac{\|\hat{y}_o - x_i\|}{h} \right),
\]

where \( g(x) = -k'(x) \) is the negative derivative of the kernel profile.

### 2.3. Background-weighted histogram

In [5], the background information is integrated into the mean shift tracking algorithm for selecting the salient parts from the representations of the target model and target candidates. Let \( \{\hat{\sigma}_u\}_{u=1 \ldots m} \) (with \( \sum_{u=1}^{m} \hat{\sigma}_u = 1 \)) represent the discrete color histogram of the background. The minimal non-zero value is denoted by \( \hat{\sigma}^* \) in \( \{\hat{\sigma}_u\}_{u=1 \ldots m} \). The transformation between the target model and target candidate model is defined as coefficients \( \{v_u = \min(\hat{\sigma}^*/\hat{\sigma}_u, 1)\}_{u=1 \ldots m} \), which reduces the weights of those features in the background with low \( v_u \). The new target model and the target candidate model representations are expressed as

\[
\hat{q}_u = C_v \sum_{i=1}^{n} k (\|x_i^*\|^2) \delta [b(x_i^*) - u],
\]

\[
\hat{p}_u(y) = C_h v_u \sum_{i=1}^{n} k \left( \frac{\|y - x_i\|}{h} \right) \delta [b(x_i) - u],
\]
where $C$ and $C_h$ are the normalization constants that satisfy $\sum_{u=1}^{m} \hat{q}_u = 1$ and $\sum_{u=1}^{m} \hat{p}_u = 1$.

3. Our tracking algorithm

In many applications, it is difficult to exactly delineate the shape of the target. For example, the target object may be affected by the occlusions of other objects, or interfered by the complex background clutter. This would result in an inclusion of the background features in the target model, or some of the target features are presented unenviably in the background. The purpose of BWH [5] is to reduce the background influence in target representation and localization by decreasing the probability of the prominent background features in the target model and candidate model. However, this method failed to achieve its goal as the result is proven to be exactly the same as the mean shift tracking with usual target representation [9].

In this paper we propose an approach to enhance the mean shift tracking algorithm by the Gaussian Mixture Models (GMMs). Our tracking algorithm consists of four fundamental steps as follows:

1. The target object is initialized as an ellipsoidal region by the user in the first frame of the video sequence. A level set based active contour image segmentation method is then performed to separate the target object and the surrounding background. The purpose is to detect the contour of the target object.
2. Based on the segmented object and background, the GMMs are used to model the color histograms of the target object and the background.
3. GMMs are incorporated into the mean shift algorithm to tune the weights on the object features so as to reduce the background information on the subsequent frames.
4. Moments of the weights are exploited to estimate the scale and orientation changes of the target object.

We shall elaborate each step systematically in the following subsections.

3.1. Target initialization

In the first frame of the video, user defines an ellipsoidal region of interest as the target object. Such region usually contains both the target object pixels and the background pixels. To reduce the interference of the background, we utilize the Distance Regularized Level Set Evolution (DRLSE) image segmentation method in [16] to separate the target object from the background. DRLSE is an active contour level set method that segments image using dynamic curves. Its central idea is to use a signed distance function to represent the evolving contour, with the zero level corresponding to the actual contour. Based on the contour’s motion equation, a gradient flow for the implicit surface can be easily derived by minimizing an energy functional with a distance regularization term. When an external energy is applied to the zero level set, it drives the propagation of the contour to desired locations. The DRLSE method has the capability of intrinsically maintaining the regularity of the level set function, thus providing a direct way to estimate the geometric properties of the evolving structure. The method also ensures accurate computation and stable level set evolution. It is an expedient framework to address numerous applications in computer vision and medical image analysis. In our approach, we have applied DRLSE for object/background segmentation. Figure 1 shows the segmentation of an object and its background by DRLSE.
3.2. Object/Background modeling using Gaussian Mixture Models (GMMs)

In this paper, we aim to improve the kernel-based mean shift algorithm [5] by leveraging the GMMs to model the target object and its background. The objectives are to decrease the weights of prominent background features while at the same time, increase the weights of the salient features for the target model and candidates.

The GMMs are learnt at the first frame of the video sequence based on the segmentation results acquired in the previous subsection. After we segregate the object and its background by the level set based image segmentation method, we learn two Gaussian mixture models, one modeling the color histogram of the object and one the color histogram of the background. Let \( c = (c_r, c_g, c_b)^T \) represent the pixel’s color value. The probability of a pixel belonging to the object or background is given by

\[
p(c) = \sum_{k=1}^{K} w_k g(c | \mu_k, \Sigma_k),
\]

where \( w_k \) is the weight for respective Gaussian distribution, which can also be interpreted as the a priori probability of the \( k \)th Gaussian component in the mixture model that satisfies the constrain of \( \sum_{k=1}^{K} w_k = 1 \), and \( g(c | \mu_k, \Sigma_k), k = 1, \ldots, K \), are the component Gaussian densities. Each component density is a 3D variate Gaussian function of the form

\[
g(c | \mu_k, \Sigma_k) = \frac{1}{(2\pi)^{D/2} |\Sigma_k|^{1/2}} \exp \left\{ -\frac{1}{2} (c - \mu_k)^T \Sigma_k^{-1} (c - \mu_k) \right\},
\]

with a 3x1 mean vector \( \mu_k \) and a 3x3 covariance matrix \( \Sigma_k \), \( |\Sigma_k| \) is the determinant of \( \Sigma_k \), and \( \Sigma_k^{-1} \) is its inverse.

In order to fit the Gaussian mixture model to the corresponding color histogram, the parameters \( \lambda = \{ w_k, \mu_k, \Sigma_k \} \) have to be estimated. There are several techniques for estimating the parameters of a GMM. By far the most popular and well-established method is the expectation maximization (EM) algorithm [17]. Let \( t \) be the iteration step, and \( N \) be the size of data sample \( c_n \), the a posteriori probability for the \( k \)th Gaussian distribution is given by

E-step:
In each EM iteration, the following parameters are re-estimated to maximize the likelihood of the data.

**M-step:**

\[
Pr(k \mid c_n, \lambda) = \frac{w_{k,t} \cdot g(c_n \mid \mu_{k,t}, \Sigma_{k,t})}{\sum_{k=1}^{K} w_{k,t} \cdot g(c_n \mid \mu_{k,t}, \Sigma_{k,t})}. 
\]  

The EM algorithm iterates between steps \( t \) and \( t+1 \) to converge to a local maximum of the likelihood. In contemplation of deciding whether the pixel belongs to the GMM of the object \( p_{obj}(\epsilon) = p(\epsilon \mid \alpha = 1) \) or the background \( p_{bg}(\epsilon) = p(\epsilon \mid \alpha = 0) \), we use the maximum a posteriori (MAP) estimation. Using log-likelihoods, the typical form of the MAP estimation is given by

\[
\hat{\alpha} = \arg \max_{\alpha} \ln p(\alpha) + \ln p(\epsilon \mid \alpha), 
\]

where \( \hat{\alpha} \in [0, 1] \) indicates that a pixel’s color value \( \epsilon \) belongs to the object \( (\hat{\alpha} = 1) \) or the background \( (\hat{\alpha} = 0) \), and \( p(\alpha) \) is the corresponding a priori probability. The initialization of \( p(\alpha) \) can be set by using the segmentation results discussed in subsection 3.1.

### 3.3. Mean Shift with GMMs

In this subsection, we discuss on how to integrate GMMs into the mean shift algorithm. In our method, the weight is defined as the probability that the pixel belongs to an object or background \( p(\epsilon) \).

For each pixel \( x_i \), its weight is defined as \( p(x_i) \). Thus the new target model can be written as

\[
\hat{q}_i' = C' \sum_{k=1}^{K} p(x_i) k(\|x_i\|^2) \delta[b(x_i) - u], 
\]

with the new normalization constant \( C' \) expressed as

\[
C' = \frac{1}{\sum_{k=1}^{K} p(x_i) k(\|x_i\|^2) \delta[b(x_i) - u]}. 
\]

Similarly, the new target candidate representation is

\[
\hat{p}_i'(y) = C_h' \sum_{y=1}^{n} P(x_i) k \left( \| \frac{y - x_i}{h} \|^2 \right) \delta[b(x_i) - u], 
\]

where now \( C_h' \) is given by

\[
C_h' = \frac{1}{\sum_{y=1}^{n} P(x_i) k \left( \| \frac{y - x_i}{h} \|^2 \right) \delta[b(x_i) - u]}. 
\]
The weight of the pixel $x_i$ is computed as
\[
w'_i = p(x_i) \sum_{u=1}^{m} \delta[b(x_i) - u] \sqrt{\frac{q_u'}{\hat{p}_u'(\hat{y}_0)}}.\]  

The new estimation of the new target position $\hat{y}'_1$ is given by
\[
\hat{y}'_1 = \frac{\sum_{j=1}^{n_0} x_j w'_j g\left(\frac{\|\hat{y}_0 - x_j\|}{h}\right)}{\sum_{j=1}^{n_0} w'_j g\left(\|\hat{y}_0 - x_j\|/h\right)}.
\]

When we choose the kernel $k(x)$ with the Epanechnikov profile, which is $g(x) = -k'(x) = 1$, the above Eq.(22) can be reduced to
\[
\hat{y}'_1 = \frac{\sum_{j=1}^{n_0} x_j w'_j}{\sum_{j=1}^{n_0} w'_j}.
\]

### 3.4. Scale and orientation estimation

For estimating the target scale and orientation, the moments of the weight image are used. The weight image is corresponding to $w'_i$. The mean location, scale and orientation can be calculated by first finding the zeroth moment and the second order moments using
\[
M_{00} = \sum_{i=1}^{n_0} w'_i,
\]
\[
M_{20} = \sum_{i=1}^{n_0} w'_i x_{i,1}^2,\quad M_{02} = \sum_{i=1}^{n_0} w'_i x_{i,2}^2,\quad M_{11} = \sum_{i=1}^{n_0} w'_i x_{i,1} x_{i,2}.
\]
The mean location of the target candidate region is computed as
\[
(\bar{x}, \bar{y}) = (M_{10}/M_{00}, M_{01}/M_{00}).
\]
The second order central moments are determined by
\[
\mu_{20} = M_{20}/M_{00} - \bar{x}^2,\quad \mu_{11} = M_{11}/M_{00} - \bar{x} \bar{y} - \bar{y}^2,\quad \mu_{02} = M_{02}/M_{00} - \bar{x}^2.
\]
Then the target scale and orientation can be obtained by decomposing the covariance matrix as follows
\[
\begin{bmatrix}
\mu_{20} & \mu_{11} \\
\mu_{11} & \mu_{02}
\end{bmatrix} = U \times S \times U^T,
\]
where $U = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}$ and $S = \begin{bmatrix} \hat{\lambda}_1^2 & 0 \\ 0 & \hat{\lambda}_2^2 \end{bmatrix}$. The eigenvectors $(u_{11}, u_{12})^T$ and $(u_{21}, u_{22})^T$ represent the orientation of the two major axes of the ellipse target. The values $\hat{\lambda}_1$ and $\hat{\lambda}_2$ denote the approximated length and width of the ellipse target. The actual length $l$ and width $w$ can be defined as $l = k\hat{\lambda}_1$ and $w = k\hat{\lambda}_2$, where $k$ is a scale factor. Denoting the target area as $A_0$, we have $\pi lw = \pi (k\hat{\lambda}_1)(k\hat{\lambda}_2) = A_0$. We can then derive
The zeroth moment can be regarded as the real area of the target that is \( A_0 = M_{00} \). Therefore, the length \( l \) and width \( w \) can be computed as:

\[
\begin{align*}
l &= \sqrt{\lambda_1 M_{00} / (\pi\lambda_2)} , \\
w &= \sqrt{\lambda_2 M_{00} / (\pi\lambda_1)} .
\end{align*}
\]

### 4. Experimental results and discussions

We evaluate the proposed non-rigid object tracking algorithm with several video sequences in this section. These sequences consisted of one synthetic [12] and two real-world video sequences. In addition, we compared our proposed algorithm with the classical mean shift tracking algorithm with adaptive scale [5], the EM-shift algorithm [13] and the SOAMST algorithm [12]. Our proposed algorithm was implemented under the programming environment of MATLAB R2012a.

The first experiment was on a synthetic ellipse sequence (see Figure 2). The sequence was used in [12] to evaluate the SOAMST algorithm. The grey color ellipse was chosen as the target object. When we initialized the target region, we deliberately included some white background into the target region on the upper right and bottom left sides. Our motive was to make the initialized target region slightly deviated from the real target object region. This is usually the case for manual initialization, where the selected target may be poorly initialized. We applied different tracking algorithms on the same initialized target region, which is represented by a red ellipse. We can observe in Frame 1 (first column) that the estimated target regions by the adaptive scale mean shift and EM-shift are similar to the initialized target region, whereas SOAMST has introduced more background into the estimated target region. In contrast, our proposed algorithm was able to distinguish the object from the background better. The result of the estimated target region for our method was very close to the real target region. For subsequent frames, the adaptive scale mean shift and EM-shift were unable to localize the target position, as well as estimate the scale and orientation of the target accurately. The SOAMST could locate the target location well, but failed to estimate the scale of the synthetic grey ellipse. The experimental results show that our proposed algorithm provides better tracking results, as the tracked contour is very close to the real target boundary.

The second experiment was on a penguin video sequence, which was partially taken from the DreamWorks Animation’s movie called “Madagascar”. For this sequence, the penguin at the center of the screen was selected as the target object. Note that the target object is partially occluded by the fluttering feathers and the pillow below. Since the color of the feathers and pillow are similar to the penguin body, it increases the difficulty for accurate target object localization as well as contour detection. The tracking results in Figure 3 show that the object tracking using adaptive scale mean shift, EM-shift and SOAMST have included some feathers and part of the pillow in their estimated target region. The adaptive scale mean shift wrongly estimated the scale of the target, and was also unable to estimate the orientation of the target. The EM-shift and SOAMST, on the other hand, have inaccurately estimated the target orientation. The results show that our proposed algorithm can track the target object robustly when it undergoes occlusion.

The third experiment was on a video sequence of a rolling apple. For this experiment, we choose the apple as the target object. As shown in Figure 4, the scale of the apple is changing constantly while it's rolling, the light projected on the apple changes too. The sequence is incredibly challenging because the apple's appearance and color are not consistent due to its movement and lighting. Moreover, the shadow of the apple is also interfering with the object itself. The results show that the adaptive scale mean shift and EM-shift cannot localize the target correctly. Both methods also failed to estimate the constant changes to the scale and orientation of the target. Likewise, the performance of SOAMST is affected by the illumination variations. The algorithm could only track part of the apple whose color is identical to the initialized target. Our proposed method is not only able to achieve the goal of correctly estimating the scale and orientation of the object, but also has the capability to provide accurate localization.
Figure 2. Comparisons of object tracking on a synthetic ellipse sequence. From left to right, frames 1, 20, 40, 72 are shown.

Figure 3. Comparisons of object tracking on a penguin sequence. From left to right, frames 1, 31, 60, 124 are shown.
We compared the performance of our proposed algorithm with the adaptive scale Mean Shift tracking algorithm in [5], the EM-shift algorithm in [13] and SOAMST algorithm in [12]. Table 1 lists the mean localization errors (MLE) and the true area ratios (TAR) of the four object tracking methods exerted on the three video sequences. The TAR is defined as the ratio of the overlapped area between the tracking result and ground truth to the area of ground truth. The MLE and TAR are closely related to the scale and orientation estimation of the target being tracked. The comparison results in Table 1 show that our proposed method outperformed other tracking methods. We can see the adaptive Mean Shift object tracking algorithm and EM-shift algorithm failed to localize the object center accurately, and the scale and orientation of the target are also estimated erroneously. SOAMST performed better than the other two methods. However, it only worked best on some video sequences, and not too well on others. As such, the user may need to manually input unique parameters for different video sequences. If the parameters are not set correctly, the tracking area will have the tendency to continue shrinking or enlarging. The experimental results show that our proposed algorithm has good accuracy in tracking the target object throughout the video sequences. It has proven to be robust to occlusions, object scale variations, rotation in depth, and illumination variations.

Table 1. The MLE and TAR values of different object tracking methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Scale Adaptive MS</th>
<th>EM-Shift</th>
<th>SOAMST</th>
<th>Our Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MLE</td>
<td>TAR</td>
<td>MLE</td>
<td>TAR</td>
</tr>
<tr>
<td>Ellipse</td>
<td>21.24</td>
<td>50.08%</td>
<td>16.43</td>
<td>53.92%</td>
</tr>
<tr>
<td>Penguin</td>
<td>10.12</td>
<td>83.69%</td>
<td>7.77</td>
<td>87.51%</td>
</tr>
<tr>
<td>Apple-rolling</td>
<td>26.18</td>
<td>45.49%</td>
<td>50.31</td>
<td>14.62%</td>
</tr>
</tbody>
</table>

5. Conclusions

In this paper, we have proposed an approach to enhance the mean shift tracking algorithm by the Gaussian Mixture Models (GMMs). We improved the target object representation by employing a level set segmentation method to separate the target object from the surrounding background before using the GMMs to model the color histograms of the target object and the background respectively. We have also incorporated the GMMs into the mean shift algorithm to tune the weights on the object features. By decreasing the probability of the background features and increasing the weight of the target object, the background information on the subsequent frames are effectively reduced. We have...
also utilized the moments of the weights to calculate the scale and orientation changes to the target object. The experimental results have demonstrated the effectiveness of the proposed algorithm on tracking objects and estimating the scale and orientation changes of them. In addition, the tracking algorithm performs robustly in situations of partial occlusions, scale variations, rotation in depth, as well as illumination variations. Finally, our algorithm has shown superior tracking performance when compared to the adaptive scale Mean Shift tracking algorithm, the well-known EM-shift algorithm and the SOAMST algorithm.

6. References