International Journal of Computer Applications (0975 – 8887)
*Volume *– No.*, ___________ 2013*

Review of Mean Shift Algorithm and its Improvements

Thomas Chris
Student, D.J Sanghvi COE
D.J Sanghvi College of Engineering, Mumbai-56

S.H Karamchandani
Asst.Prof, D.J Sanghvi COE
D.J Sanghvi College of Engineering, Mumbai-56

T.D Biradar
Asst. Prof, D.J Sanghvi COE
D.J Sanghvi College of Engineering, Mumbai-56

ABSTRACT
This paper review's the origins of basic mean shift algorithm, as being a procedure which iteratively moves data points to the average of data points and its extension to the field of object tracking. Tracking of any given object forms integral part in surveillance, control and analysis applications. The video tracker presented here works on the principle of mean shift algorithm. However tracker is challenged when there tends to be low illumination, scaling, occlusions and multiple tracking. To tackle these problems, improvements are made in existing mean shift tracking algorithm of which a few are reviewed and studied.

Keywords
mean shift algorithm, tracking, iterative, improvements

1. INTRODUCTION
Given a set of sample data points whose probability density function (pdf) is unknown, density estimation is used to determine the unknown pdf by observing the data samples. B.W Silverman in [1] describes methods to estimate the density from the given data. The two approaches to estimate the density are parametric and non-parametric methods. Non-parametric methods of density estimation have fewer assumptions. In most cases non-parametric methods are used to determine the density of a given sample region. K. Fukunaga and Hostetler in [2] used sample observation and estimated the gradient of density function. From the expression obtained it is clearly seen that for a given sample region, simple mean shift about the point is obtained. Thus a key idea of mean shift was introduced by estimating the gradient of density function. Cheng [3] generalized the mean shift algorithm and proved that mean shift is gradient ascent. He introduced the concept of 'shadow' of kernel and that mean shift performed on any kernel is gradient ascent on density estimated with its shadow. D.Comaniciu in [4] extended the concept of mean shift to object tracking .Object tracking starts with obtaining the initial frame also called the reference frame and selecting the object of interest. The target is tracked in next sequence of subsequent frames to repeat until end of frames. [5] The aim of an object tracker is to generate the trajectory of an object over time by locating its position in every frame of the video. The use of object tracking has many vision applications such as video surveillance, traffic monitoring, robot vision, animation and defense areas. [5][6][7] Tracking faces challenges with low-illumination, occlusions, scale changes and multi-tracking. [8] Enormous improvements have been made in the mean shift tracking algorithm. Our paper reviews the origins of basic mean shift algorithm, extensions to object tracking and improvements like effective target representation, illumination invariant mean shift tracking, and making the tracker scale adaptive. This paper is organized as follows. Section 2 discusses basic mean shift algorithm Section 3 shows mean shift extended to tracking Section 4 Improvements in the existing tracking using mean shift algorithms and followed by conclusion in Section 5.

2. Basic Mean Shift
Some of the non-parametric methods used to determine the density estimate are histogram, naïve estimator, kernel estimator [1]. Kernel estimator is used mostly due to its advantages over the other two. Kernel estimator is given by the formula [8]:

$$\hat{f}(x) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right) \tag{1}$$

Where a given set $\{x_i\}_{i=1..n}$ of n points in the d-dimensional space R^d,$K(x)$ is the kernel with window radius (also known as bandwidth) h computed for point x.

The radially symmetric kernel is defined as,

$$K(x) = c_k k(\|x\|^2) \tag{2}$$

Where $k(x)$ is referred to as profile of the kernel K, the commonly used kernels are epanechnikov and Gaussian kernels and c_k is the normalization constant. Some of the other kernels are Triangular and Uniform.

Consider Epanechnikov kernel is used then,

$$K_E(x)=\begin{cases} \frac{1}{2}{C_d}^{-1}(d+2)(1-\|x\|^2) & \text{if } \|x\|<1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

in which C_d is the volume of the unit d-dimensional sphere and its profile is given by:

$$k_E(x) = \begin{cases} 1-x & 0 \le x \le 1 \\ 0 & x > 1 \end{cases} \tag{4}$$

Taking gradient of the density estimator, the following expression is obtained.

$$\nabla\hat{f}(x)=\frac{2c_{k,d}}{nh^{d+2}}\left[\sum_{i=1}^{n} g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)\right]\left[\frac{\sum_{i=1}^{n} x_i g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x\right] \tag{5}$$

The first term is proportional to density estimate at x computed with kernel G and the second term is mean shift obtained with kernel K. When the gradient is set to zero, it is found that at any mode, the following condition holds.

$$x = \frac{\sum_{i=1}^{n} x_i g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} \tag{6}$$

$$x_{ms} = \left[\frac{\sum_{i=1}^{n} x_i g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x\right] \tag{7}$$

Hence we iteratively update the point x. The resultant iterative procedure is called the mean shift algorithm. The mean shift procedure for point x_i is given as follows. [10]

- Mean shift vector is computed (Eq. 5)
- Translate density estimation window ${x_i}^{t+1} = {x_i}^{t} + x_{ms}$
- Iterate above two steps until convergence i.e. $\nabla\hat{f}(x_i) = 0$.

3. MEAN SHIFT EXTENDED TO TRACKING

Visible feature's is used to characterize the object of interest, in this case color is used as a feature in order to track the object using mean shift algorithm. [11] A variety of similarity measures between the target model (its color distribution) and the target candidates such as Kullback–Leibler divergence (KL Divergence), Mean squared error(MSE), Structural similarity(SSIM), Euclidean Distance and Bhattacharya coefficient can be used of which Bhattacharya coefficient is most commonly used**.** Mean shift algorithm tracks by maximizing the Bhattacharya co-efficient hence it also being equivalent to minimizing the distance between the two probability density functions. [12]

3.1Color Representation

In the reference frame the target can be defined by a square, rectangle, or a circle and color histogram is used to characterize the target object. The selected object is then modeled as an histogram having m-bins located at the origin as [11][12]

Target model: $\hat{q} = \{\hat{q}_u\}_{u=1\ldots m}$ $\sum_{u=1}^{m} \hat{q}_u = 1$

In the next frame, the targeted object shifts to new location y and its histogram is given by

Target Candidate: $\hat{p}(y) = \{\hat{p}_u(y)\}_{u=1\ldots m}$ $\sum_{u=1}^{m} \hat{p}_u = 1$

Given the reference image, probability density of the target model is given by:

$$\hat{q}_u = C \sum_{b(x_i)=u} k(||x_i^*||^2) \quad (8)$$

In the above expression, 'C' is normalization factor, $\{x_i^*\}_{i=1..m}$ is the current pixel location, k(x) is the kernel profile and $b(x_i)$ is color bins (from 1 to m).

Given the next image, probability density of the target candidate is given by:

$$\hat{p}_u(y) = C_h \sum_{b(x_i)=u} k\left(\left\|\frac{y-x_i^*}{h}\right\|^2\right) \quad (9)$$

The target now shifts to a new location y, where C_h is the normalization factor, $\{x_i^*\}_{i=1..m}$ is the current location, k(x) is the same kernel profile and $b(x_i)$ is color bins(1…m)

Bhattacharya co-efficient is used as similarity measure between two probability function $\hat{q}$ and $\hat{p}$ is given by:

Similarity function: $\hat{\rho}(y) \equiv \rho[\hat{p}(y), \hat{q}]$ (10)

3.1.1 Metric based on Bhattacharya co-efficient

Bhattacharyya co-efficient is determined in the next frame between target model and target candidate which defines a distance $d(y)$ between two distributions (target and candidate model) given by the following relation [6] [7]

$$d(y) = \sqrt{1 - \rho[\hat{p}(y)\hat{q}]} \quad (11)$$

3.1.2 Target Localization

Bhattacharya co-efficient $\rho[\hat{p}(y), \hat{q}]$ is estimated using taylor expansion. In the current frame the location of the target can be found by minimizing the distance as function of y. Minimizing the distance is equivalent to maximizing the Bhattacharya coefficient [4][6][7][8]

$$\rho[\hat{p}(y), \hat{q}] \approx \frac{1}{2}\sum_{u=1}^{m} \sqrt{\hat{p}_u(y_0)\hat{q}_u} + \frac{1}{2}\sum_{u=1}^{m} \hat{p}_u(y)\sqrt{\frac{\hat{q}_u}{\hat{p}_u(y_0)}}$$

$$\rho[\hat{p}(y), \hat{q}] \approx \frac{1}{2}\sum_{u=1}^{m} \sqrt{\hat{p}_u(y_0)\hat{q}_u} + \frac{C_h}{2}\sum_{b(x_i)=u} w_i k\left(\left\|\frac{y-x_i}{h}\right\|^2\right) \quad (12)$$

Where $w_i = \sqrt{\frac{\hat{q}_u}{\hat{p}_u(y_0)}}$ (13)

By maximizing the second term of expansion similarity can be maximized, the first term being independent of y. [7]
The mode of equation (8) sought the maxima and is given by

$$\hat{y}_1 = \frac{\sum_{i=1}^{n_h} x_i w_i g\left(\left\|\frac{\hat{y}_0 - \hat{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{\hat{y}_0 - \hat{x}_i}{h}\right\|^2\right)} \quad (14)$$

3.2 TRACKING PROCEDURE

- A pre-recorded video is taken as input and parameters related to video (height, width and length) are obtained.
- Pixel values of each frame are read into a structure.
- Select the object to be tracked i.e target(user input) and obtain parameters related to video (centre, height and width)
- Obtain the Kernel profile and gradient parameter.
- Find pdf of object to be tracked in reference frame using (6)
- Initialize maximum number of iterations.
- Initialize location of target as y_o and find $\hat{\rho}(y_o)$ using (8).
- Derive the weights using (11).
- Find the next location of target candidate using (12).
- Compute $\hat{\rho}(y_1)$ at the new location using (8)
- If $\hat{\rho}(y_1) < \hat{\rho}(y_0)$

  Then $\hat{y}_1 = \frac{1}{2}(\hat{y}_0 + \hat{y}_1)$ (15)
- If maximum number of iterations is complete then draw window around target else repeat from deriving weights.
- Stop if end of video.

4. AREA OF IMPROVEMENTS IN MEAN SHIFT TRACKING ALGORITHM

4.1 Improvements for effective Target Representation

In mean shift tracking algorithm, the target object is represented by a circle, rectangle or ellipse in a frame due to which there are always background regions surrounding the object. The presence of background region hinders the target localization accuracy when interrelationship between target and background is high. In order to improve the tracking performance in the mean shift algorithm, background weighted histogram was proposed by D.Comaniciu in [11]. The principle behind Background Weighted Histogram (BWH) is to use scale transformation to provide lower weights to the background with distinguished features. The background is represented by $\{\hat{\boldsymbol{o}}_u\}_{u=1..m}$ where $\hat{\boldsymbol{o}}^*$ is its minimal value. The scale transformation co-efficient is given by:

$$\{\mathrm{v_u} = \min(\hat{o}^*/\hat{o}_u), 1\} \tag{16}$$

Thus in background weighted histogram, the co-efficient (equation 14) is multiplied with target and candidate model density function (i.e. equation 6 and 7), to reduce the probability of distinguished background features of the target and of the candidate model.

J.Ning in [12] proved that although BWH scale transforms the target and candidate model it yields the same expression as the mean shift iteration formula and also the weight computed by BWH turns out to be directly proportional to the weight of mean shift algorithm as in equation 11. Thus, Corrected Background Weighted Histogram (CBWH) was proposed having the same concept of BWH to provide lower weights to the background .Here the only difference is that the target model is transformed and not the candidate model. The CBWH method thus attains the objective to reduce background hindrances and provide accurate target localization unlike the BWH method. The weight computed by CBWH methods results as follow:

$$w_i{}'' = \sqrt{v_{u'}}\, w_i \tag{17}$$

where w_i is weight calculated using the general target representation, $v_{u'}$ is the transformation co-efficient which provides lower weights to distinguished background features. In addition to this, the background of the target model is updated, since the background changes during the course of tracking. Thus by reducing the background features, target localization is improved which quickens up the convergence process. Experimental results have shown that CBWH require lesser number of iterations for target localization compared to BWH.Now instead of representing the target object by using standard shapes of circle, rectangle or ellipse, in which the target object tends to have background regions X.Chen in [13] proposed improved mean shift algorithm by completely eliminating the background and representing the target object by its true shape. Here only the target object is selected with no background regions. Since background affects the tracking algorithm when background changes remarkably or when it's convoluted. This algorithm can be basically divided into three steps. Firstly the graph cut image segmentation algorithm is used in order to select the object shape and to separate the object from the background. Once background is separated from the target object, next is to determine the level set kernel and obtain kernel gradient estimates.The level set kernel is given by:

$$k(x) = C\theta(x)I(x) \tag{18}$$

where 'C' is the normalization constant, $\theta(x)$is implied level set function, having a convex and monotonically decreasing profile. It stands for the signed distances of point *x* from the object edge. $I(x)$ is the indicator function which is set zero when point *x* is outside the object edge. Finally, apply mean shift tracking algorithm using the above kernel. The algorithm proposed by X.Chen [12] as compared to the other two mentioned methods is however computationally complex.

4.2 Scale Adaption

During the course of tracking the scale of the object being tracked changes, as it tends to enlarge or diminish depending upon the object's movement towards or away from the camera. Traditional mean shift algorithm is not adaptive to scale changes. Hence, L.Liu in [14] proposed adaptive kernel bandwidth, wherein the kernel bandwidth expands or shrinks by 10%. To determine whether the target is enlarging or diminishing, first the Bhattacharya co-efficient between target and its background is computed which is denoted as ρ_{tb}, then the Bhattacharya co-efficient between candidate and target background is computed and it is denoted as $\rho_{tb}{}'$.Now the author compares, if $\rho_{tb}{}' < \rho_{tb}$,then the target is expanding hence let h=(1+10%)*h until, $\rho_{tb}{}' \geq \rho_{tb}$, and make h the current kernel bandwidth. Likewise if $\rho_{tb}{}' > \rho_{tb}$ the target is shrinking hence let h = (1-10%)*h until, $\rho_{tb}{}' \leq \rho_{tb}$ and make h the current kernel bandwidth.C. Wang in [15] suggested another technique to make the kernel bandwidth adaptive. Here, the moving target object is detected using interframe differencing or background subtraction. The contour tracking algorithm is combined with mean shift to obtain the contour information which updates the kernel bandwidth adaptively. C. Wang [15] has made scale of the moving object adaptive, unlike [14] where the scale changes by fixed amount of 10%.

L.Zhang in [16] in order to make the kernel bandwidth adaptive, has proposed the use of canny operator along with mean shift algorithm. The canny edge detection algorithm is used to determine whether the object scale changes or not, then accordingly bandwidth is made adaptive. In the canny edge detection algorithm, first a gaussian filter to eliminate the noise, the image intensity gradients are determined and non-maximum suppression and double threshold are applied to determine the potential edges. Once the object edges are determined, with the extreme corner points $(x_0 y_0)$ and $(x_1 y_1)$ the diagonal distance is calculated which is denoted by β.

$$\beta = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \tag{19}$$

Now β of the previous frame is denoted as β_0 and that of the current frame as β_1.

If $\beta_1 > \beta_0$ then h $= (1 + \mathrm{a})$*h
If $\beta_1 < \beta_0$ then h $= (1 - \mathrm{a})$*h

Hence bandwidth is made adaptive where 'a' is the scaling factor, usually 0.1 in this case. This algorithm is only relevant in scenarios where the target and the background are clearly differentiated and when background interference is the least.During the course of tracking, the scale and orientation of the object changes which is not adaptively estimated in the basic mean shift tracking algorithm. Hence, J.Ning in [17] proposed the scale and orientation adaptive mean shift algorithm (also known as SOAMST) towards the betterment of the original mean shift algorithm. The procedure is as follows, the target area is estimated with proposed formula using zeroth-order moments of the weight image and a monotonically increasing function which depends on the Bhattacharyya co-efficient. Using the zeroth and first order moments of weight image the new location of target are found. The second order moments written in a covariance matrix is used to estimate the orientation, width and height of the target. The covariance matrix when decomposed consists of the estimated target area too. In the current frame the location, scale and orientation of the target are known. The covariance matrix is used to determine the size of target candidate region in the next frame and its position too.

4.3 Illumination Invariance

So far color was used as feature space to describe the object to be tracked. A.Babaeia in [18] has used multiple features like color, edge and texture to describe the object to be tracked. Each feature is extracted from the first frame and the mean shift tracking algorithm is applied on each feature independently to track the objects trajectory. With color as feature the weighted color histogram is computed for target and the candidate model. For edge as feature the gradient of intensity and edge direction are found and the edge weighted histogram for target and candidate model are computed. Likewise features for texture are extracted using Discrete Wavelet Transform (DWT).For each of the RGB window, texture is represented by its feature vector. Thus, histogram of the texture for target model and candidate is found. The mean shift algorithm is applied independently for each feature .The bhattacharyya co-efficient, weights and target new location for each feature is calculated. The weight of each feature *f* is calculated by minimum of bhattacharyya distance denoted as ε_f and in the last step new location of the target denoted as $\hat{y}_{1multiple}$ is integration of the multiple features.

$$\hat{y}_{1multiple} = \varepsilon_{color}\, \hat{y}_{1color} + \varepsilon_{edge}\, \hat{y}_{1edge} + \varepsilon_{texture}\, \hat{y}_{1texture} \quad (20)$$

Experimentally it has been shown that combing multiple features provides better results in the object being tracked, because when one feature fails to track the object the other features still tends to track.One of the challenges in using mean shift tracking algorithm is that sudden variations in the illumination can cause the tracker to fail.G.Phadke in [19] using mean shift as the tracking algorithm, proposed a method which is illumination invariant. A frame having low illumination leads low DC co-efficients and vice versa. Hence to achieve illumination invariance the DC co-efficient is modified by taking a number of frames together and providing uniform illumination. Hence by taking logarithm of the image the luminance and reflectance is separated. DCT-II is performed on the logarithmic image to transform it to the frequency domain. DC coefficients can now be modified adaptively using neighboring frames and also the proposed modified DC formula. Now consider the case when frames in a given window overall have low illumination, then a further correction of large neighborhood is taken into consideration. Eigen value of covariance of a matrix is taken, the maximum value is taken as reference, for comparison with respect to the other frames.

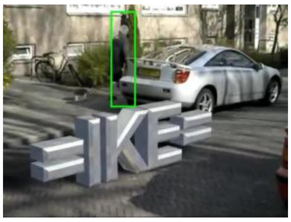
5. SIMULATION OUTPUT

Basic mean shift tracking algorithm has been applied to video sequence 1 and 2.

a.) Video Sequence-1

This video contains only one moving object. The moving speed is quite slow. The frame rate is 25 frames/sec with size 288*384 pixels.

(a) (b)



(c)

Figure 1: Video sequence-1 using Epanechnikov kernel for frames: (a) 100th frame; (b) 160th frame; (c) 352nd frame

In Video Sequence-1 tracking is performed for the person moving from 100th frame to the end of frames, the speed of moving object is slow. Mean shift tracking algorithm successfully tracks the person's movements using Gaussian, Epanechnikov and Triangular kernels except for Uniform kernel where it loses the track of selected object. Tracking result using Epanechnikov kernel is shown in Figure 1.

b.) Video Sequence-2

This video contains many moving objects. The moving speed is quite fast. The frame rate is 25 frames/sec with size 288*384 pixels.

(a) (b)

(c)

Figure 2: Video sequence-2 using Epanechnikov kernel for frames: (a) 100th frame; (b) 157th frame; (c) 226th frame

In Video-sequence-2, the car is the object to be tracked. The car is moving at a moderate speed. Mean shift tracking algorithm successfully tracks the moving car using Gaussian, Epanechnikov and Triangular kernels except for Uniform kernel where it loses the track of selected object. Tracking result using Epanechnikov kernel is shown in Figure 2.

6. RESULT

Evaluation for Video Sequence-1 and Video sequence-2 is done by plotting the graph of Bhattacharyya co-efficient versus frame index for Epanechnikov, Gaussian, Triangular and Uniform kernels using mean shift algorithm. The graph is shown below.

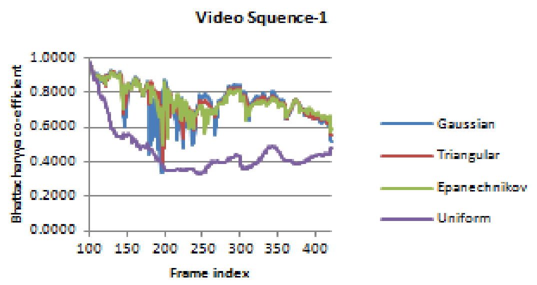


Figure 3: Comparison of kernels for Video Sequence-1

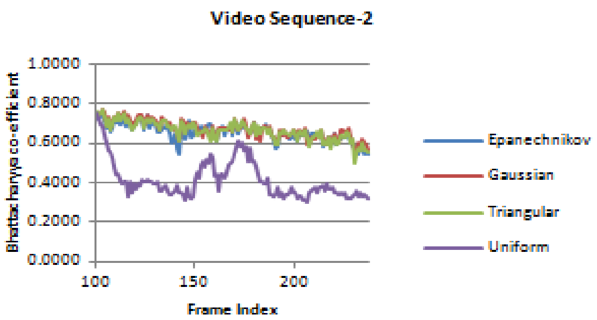


Figure 4: Comparison of kernels for Video Sequence-2

Comparison for Video Sequence 2 based on the statistical performance measure of Precision (P), Recall(R) and Negative Predictive Value (NPV) is tabulated for mean shift algorithm using different kernels.

Table 1. Statistical performance measure for different kernels for Video Sequence-2 is tabulated.

Kernels	Meanshift using Bhattacharya Co-efficient		
	P	R	NPV
Epanechnikov	69.69	100	100
Gaussian	69.69	100	100
Triangular	69.69	100	100
Uniform	100	15.15	70.05

7. CONCLUSION

Mean shift as an independent identity is a simple and computationally efficient iterative algorithm. The mean shift tracking algorithm locates the position of target object across successive video frames. A number of improvements have been made to the conventional mean shift algorithm owing to the challenges and significant improvements have been made. Meanshift tracking algorithm using Bhattacharyya distance is performed for all the two video data sets using Epanechnikov, Gaussian, Triangular and Uniform kernel. The tracking obtained by Uniform kernel is poor since the object looses track. On checking the graph of Bhattacharyya distance versus frame index, Uniform kernel provides least performance too. Kernels like Epanechnikov, Gaussian and Triangular provide good tracking results and also its Bhattacharya distance is comparatively higher than uniform kernel. The recall rate for Uniform kernel is less compared to the other kernels.The review elaborates improvements made to improve mean shift tracking performance. One of the improvements in mean shift is to effectively represent the target, by providing higher weights to the target and to provide the background with lower weights. For effectively representing the target, the background can also be completely eliminated. Another significant improvement was made in making the scale adaptive when size of the object changes. The change in target object shape can be determined by determining Bhattacharyya co-efficient, or by interframe differencing or by canny edge detection algorithm. The mean shift algorithm is also made illumination invariant by modifying the D.C coefficients. Multiple features such as color, texture and edge have been used to characterize the object of interest, by doing so if one feature fails to detect the target object; the other two features continue to detect the moving object.Thus, a few improvements in the areas of improved target representation, making the target scale adaptive and illumination invariance have been reviewed and studied. It is desired that the mean shift tracking performs suitably well under sudden changes of illumination using contrast enhancement technique grey level grouping for improved results of tracking .

8. REFERENCES

[1] B. Silverman, “Density Estimation for Statistics and Data Analysis”, 1986, New York: Chapman and Hall.

[2] K. Fukunaga, L.D. Hostetler, “The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition’, 1975, IEEE Trans. Info. Theory, vol. IT-21, 32 -40

[3] Y. Cheng, “Mean Shift, Mode Seeking, and Clustering”, 1995,IEEE Trans. Pattern Anal. Machine Intell., vol.17, 790-799

[4] D. Comaniciu, V. Ramesh and P. Meer, "Real-time tracking of non-rigid objects using mean shift," Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on, vol.2, no., pp.142, 149 vol.2.

[5] A. Yilmaz, O. Javed and M. Shah, "Object Tracking: A Survey", 2006 ACM Computing Surveys,Vol. 38, No. 4

[6] T. Moeslund, A. Hilton, and V. Kruger, “A survey of advances in vision-based human motion capture and analysis”, 2006 Computer vision and image understanding 104(2-3):90–126

[7] D. Caulfield, “Mean-Shift Tracking for Surveillance: Evaluations and Enhancements”, 2011, University of Dublin

[8] K. Hati and V.Vardhanan “Review and Improvement Areas of Mean Shift Tracking Algorithm” ,2014 ,18th IEEE International Symposium Consumer Electronics (ISCE 2014) Pg. No.1-2.

[9] D. Comaniciu, V. Ramesh and P. Meer, “Mean Shift: A Robust Approach towards feature space analysis”, May 2002 IEEE Transactions on , vol.24, no.5, pp.603,609

[10] D. Comaniciu P. Meer, “Mean Shift Analysis and Applications”, 1999, IEEE Int’l Conf. Com. Vis.,Kerkyra Greece,1197-1203

[11] D. Comaniciu, V. Ramesh and P. Meer, “Kernel-based object tracking," Pattern Analysis and Machine Intelligence”, May 2003, IEEE Transactions on , vol.25, no.5, pp.564,577

[12] J. Ning, L. Zhang, D. Zhang, and C. Wu. “Robust mean-shift tracking with corrected background weighted histogram”,January 2012, Computer Vision, IET, 6(1):62–69.

[13] X.Chen ; S.Yu and Z.Ma ,"An improved mean shift algorithm for moving object tracking ", June 25 - 27, 2008, Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China

[14] L.Liu, W.Liu, "Mean-Shift Tracking Algorithm with Adaptive Kernel-Bandwidth of Target",2013 Conference Anthology, IEEE Pg No. 1-4.

[15] C.Wang, H.Zhang,"An Adaptive Kernel-Bandwidth Mean-Shift Target Tracking Algorithm", Control Engineering and Communication Technology (ICCECT) International Conference on ,2012,Pg No. 547 - 550

[16] L.Zhang, D.Zhang, Y.Su and F.Long "Adaptive Kernel-bandwidth Object Tracking Based on Mean-shift Algorithm", 2013, Fourth International Conference on Intelligent Control and Information Processing (ICICIP)

[17] J. Ning, L. Zhang, D. Zhang and C. Wu "Scale and orientation adaptive mean shift tracking", November 2010,

[18] A.Babaeia, S.Rastegar, M.Bandarabadi and M.Rezaei "Mean Shift-Based Object Tracking With Multiple Features", March 2009, 68-72,41st Southeastern Symposium on System Theory University of Tennessee Space Institute Tullahoma, TN, USA.

[19] G.Phadke, R.Velmurugan, "IIllumination invariant Mean-shift tracking", 2013, Applications of Computer Vision (WACV), IEEE Workshop Pg no.407 – 412