International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.8, No.6 (2015), pp.33-44
http://dx.doi.org/10.14257/ijsip.2015.8.6.05

Diverse Methodologies to Improve Covariance based Object Tracking

Sanchita Singha and Sujoy Datta

School of Computer Engineering
KIIT University
Bhubaneswar, Odissa, India
sanchita.singha007@gmail.com, sjdatta88@gmail.com

Abstract

Tracking is an important process of computer vision research. But still after so many researches accuracy is still become a bottleneck. Within different tracking techniques covariance based tracking is a new technique which gives more accuracy than other techniques. There are several methods and researches have been done on covariance tracking. The covariance tracking process also uses some distance measures to calculate the dissimilarity between two target regions. Here we have list down some of the most useful distance measurement techniques which provide accurate results. We have also implemented those distance measurement techniques and shown their results with accuracy comparison. Even the distance between the target and the candidate covariance matrix is itself enough track an object, but to get more accurate result some techniques are applied on covariance tracking. Here we have listed some of those techniques which happen to provide better results after applying on covariance tracking and also pointed out the advantages and drawbacks of those techniques.

Keywords: *Covariance, manifold, Euclidian space, kernel, salient feature*

1. Introduction

The process of estimating over time the location of one or more objects using a camera is referred to as video tracking. Now a day tracking an target object become the new area of interest in the video surveillance system which become a high priority as presently each organization or building is under CCTV surveillance [1-2]. The basic steps of object tracking are feature extraction, target object representation, searching the target object in the candidate region. Feature extraction is an important process upon which the whole object tracking is based. A good feature should be robust, efficient, and easily computable [5]. In the computer vision system the image pixel intensity values such as color, gradient, edge orientation, edge magnitude etc are the most popular choice as feature. But these features are not always robust for illumination change, non rigidity characteristic, object rotation etc [3]. Color cannot capture the characteristics of the target as it is not suitable for non rigid object. Edge or Shape is not reliable for non-rigid and rotating object [4] [20]. And the efficiency of the algorithm is limited by the higher dimensional feature representation.

Porikli, Tuzel and Meer proposed the covariance based object tracking concept first [6]. Covariance matrix captures both the spatial and statistical properties of an object. It is a symmetric matrix with very low dimension. It has the information contained within the histogram as well as the appearance models. It has been proven that a single covariance matrix is enough to match regions with different viewpoints and poses. The dimension of covariance matrix is very low. It is also invariant to the identical shifting of color values, which is a very useful property when the object is tracked under illumination changes.

ISSN: 2005-4254 IJSIP

This paper is divided into the following section. In the section 2 we describe the basic of Covariance matrix, its mathematical expressions, basic of covariance tracking and diverse methodologies for covariance tracking in details. Then in section 3 we will describe some techniques which can be applied on covariance tracking. In section 4 we illustrate the comparison of the experimental results. Lastly in section 5 we outline the prime point of the discussed covariance based tracking methodologies and their future extension.

2. Introduction to Covariance Tracking

In statistics, the term covariance means that how much two random variables changes together. Covariance matrix is symmetric in nature. The diagonal elements of a covariance matrix are actually the variance of each feature and the non diagonal elements are their correlation [23]. Covariance matrix can be advantageous to use as a region representation. An object's statistical and spatial properties are contained by covariance matrix.

2.1. Feature Matrix Design

Feature extraction is the first step in the tracking procedure and allows us to highlight information of interest from the images to represent a target. A good feature should be robust, efficient, easily computed [5]. In the computer vision system the image pixel intensity values such as color, gradient, edge orientation, edge magnitude etc are the most popular choice as feature. But these features are not always robust for illumination change, non rigid object, object rotation etc. And the efficiency of the algorithm is limited by the higher dimensional feature representation.

For tracking the moving object, the target object and the candidate object both are represented in a form of rectangular window denoted by *I*, which is a colour image of three dimension [6, 8]. *I* contains *x* number of rows and *y* number of columns of intensity values [29]. Assume that *F* is the *W* X *H* X *d* dimensional feature image which has been extracted from *I*.

$$F(x,y) = \Phi(I,x,y) \tag{1}$$

The function Φ can be considered as any mapping such as colour, image gradient, edge magnitude intensity etc. For a particular rectangular shaped window $R \subset F$ let $\{f_k\}_{k=1.....n}$ is the feature vector of *d* dimension within *R*. The feature vector f_k is chosen using both; spatial attribute and appearance model [26]. Spatial attributes are obtained from the pixel coordinate values, and appearance attributes are colour, gradient etc. The approach given by Faith Porikli, Peter Meer [6] the feature vector is represented as following,

$$f_k = [\, x \quad y \quad I(x,y) \quad I_x(x,y) \quad I_y(x,y) \,.......\,] \tag{2}$$

This feature vector can also be extended by adding attributes as $R(x,y)$, $G(x,y)$, $B(x,y)$ where *x* and *y* are the location coordinate values, $I(x,y)$ is the intensity, $I_x(x,y)$ and $I_y(x,y)$ are the *x, y* image derivatives and *R(x,y), G(x,y), B(x,y)* are the red, green and blue colour values.

2.2. Covariance Matrix Embodiment

The use of covariance matrix in object tracking was first introduces by Fatih Porikli in [6]. In their proposed model the target region R with dimension *M X N* was represented by a *d X d* covariance matrix C_R as,

$$C_R = \frac{1}{MN}\sum_{k=1}^{MN}(f_k - \mu_R)(f_k - \mu_R)^T \tag{3}$$

In many papers a new term 'manifold' is used. The dictionary meaning of manifold is "having several copies". A manifold is a topological space that resembles Euclidean space near each point. More precisely, each point of an *n*-dimensional manifold has a neighborhood that is homeomorphic to the Euclidean space of dimension *n* [25].

2.3. Techniques for Dissimilarity Measurements

To find the most similar region between the target object and the candidate regions the distance between the covariance matrixes of the target object to the candidate regions is computed. The properties of covariance matrix do not apply on the Euclidian space [5]. As an example, we can say that the Euclidian space is not closed under multiplication by non real number. So an arithmetic subtraction between two matrices would not provide the distance between covariance regions.

The dissimilarity between covariance matrices can be computed by using the formula used in [6] and proposed by Förstner in [9]. The dissimilarity between two regions covariance matrix can be considered as the distance between two points on the manifold. The distance on a manifold is the minimum distance curve between the points. The curve is named as the geodesic and the length of the curve is the intrinsic distance. As we said that the arithmetic subtraction will not measure the distance between covariance matrixes so there are several methods to find out the distance.

2.3.1. Bhattacharyya Distance: The Bhattacharyya distance is useful to measure the separability between two distributions [30]. If we have two class of problems w_1 and w_2 with mean m_i and sample covariance matrix C_i then the Bhattacharya distance between w_1 and w_2 can be measured by,

$$Bhatt(w_1, w_2) = \frac{1}{8}(m_2 - m_1)^T \left(\frac{C_1 + C_2}{2}\right)^{-1} (m_2 - m_1) + \frac{1}{2}\log\left(\frac{\det\left(\frac{C_1 + C_2}{2}\right)}{\sqrt{\det(C_1).\det(C_2)}}\right) \tag{4}$$

But Bhattacharyya distance is not fast for illumination changes and for extremely fast moving object.

2.3.2. Log Euclidean Distance: Covariance matrices are symmetric and positive-definite (SPD) in nature. In the Log-Euclidean framework, the SPD matrices gets the properties of a Lie group which has the Euclidean space structure, which help us to do most common Euclidean space operations in the logarithm domain form [19]. In Log Euclidean distance between two matrices can be computed by,

$$d_L(S_1, S_2) = \| \log(S_1) - \log(S_2) \| \tag{5}$$

This method works properly for rotation and illumination variation [13]. This technique does not required complicated and computationally expensive operations like the geodesic and intrinsic mean computation of Riemannian geometry. So, we can easily work in the Euclidean space [28].

2.3.3. Cholesky Distance: The Cholesky decomposition is a concept where the covariance matrix is transformed into another parameter as S_i where, $S_i = L_i L_i^T$ and $L_i = chol(S_i)$ is the lower triangular matrix with positive diagonal entries [13]. The Choleskey distance is given by,

$$d_C(S_1, S_2) = \| chol(S_1) - chol(S_2) \| \tag{6}$$

Cholesky distance is a process of reparameterization of the covariance matrix [18]. It is easy to implement and it only takes the lower triangular matrix so the lower triangular part represent respective correlation and the diagonal entries represents the variance. As covariance matrix is a symmetric matrix so when it is represented in its lower triangular form, all the information remains intact.

2.3.4. Root-Euclidean Distance: The Root-Euclidean distance matrix is a simple distance estimator which is,

$$d_H(S_1, S_2) = \| S_1^{1/2} - S_2^{1/2} \| \quad (7)$$

Root-Euclidean distance is not very popular although it provides good results compared to Log-Euclidean and Cholesky distance estimator [13].

2.3.5. Jensen Bregmann Logdet Divergence: Cherian et al proposed another theoretically more correct and fast metric to compute the distance between two covariance matrices [14-15]. It gives a dissimilarity measure between two positive semi-definite matrices. It is defined as,

$$J(C1, C2) = \log \left| \frac{C1 + C2}{2} \right| - \frac{1}{2} \log | C1.C2 | \quad (8)$$

Some of the researchers find that JBLD provides better run time than Log-Euclidean and Förstner distance. But this method does not provide satisfactory result for same color background.

Table 1. Different Distance Measures and their Advantages and Disadvantages

	Advantage	Disadvantage
Bhattacharyya distance	Easy to implement. Our experiment proved that it provides better result than other distance measures.	It is not fast for illumination change and extremely fast moving object.
Log-Euclidean	It provides robust result to scale and rotation changes, illumination variation. Easy to implement. Provide better result for partial occlusion.	Provides less accuracy than Root-Euclidean distance. It is not robust for fast moving object.
Cholesky Decomposition	Easy implementation, fast computation.	Sometime not very robust.
Root-Euclidean	Easy implementation, fast computation.	Sometime not very robust.
Jensen Bregmann Logdet Divergence	Sometimes provide better runtime than Log-Euclidean and Förstner distance.	Not very popular. It does not provide satisfactory result for illumination change and same color background situations.

An object with motion and non-rigid structure experiences appearance, size and shape change with time. So paper [6] formulate and update the initial target representation or the model of initial frame [27]. Covariance tracker itself is enough to track a specific region but for the robustness and accuracy of the result this model update approach can be adaptable. In this approach a set of previously computed covariance matrices are used and a mean covariance matrix is calculated which combines the properties of all the previous covariance matrices.

3. Diverse Techniques of Covariance Tracking

After constructing the covariance matrix different researcher approaches diverse methodologies for the tracking the target object. Some of the most relevant techniques are discussed in detail.

3.1. Kernel Based Tracking

Region Covariance matrix used for multiple object tracking was previously done on a Particle Filter framework which has been extended to Kernel Particle Filter [10]. For a particle set at time *t*: $S_t = (S_n^{(n)})_{n=1}^N$ and associated weights $(W_n^{(n)})_{n=1}^N$ Let K is the kernel then the kernel density estimation with posterior is written as,

$$\hat{p}(x_t \mid Y_t) = \frac{1}{Nh^d} \sum_{n=1}^{N} K\left(\frac{x_t - s_t^{(n)}}{n}\right) w_t^{(n)} \tag{9}$$

Where the kernel width is denoted by h, for t time, x_t is the target set, and Y_t is the history of observation. For a given posterior estimation, the gradient can be estimated, and the mean-shift method is used so that the particle can be moved onwards the gradient direction, towards the modes of the posterior. When each particle is transferred to its sample mean it is determined by,

$$m(s_t^{(n)}) = \frac{\sum_{l=1}^{N} G(s_t^{(n)} - s_t^{(l)}) w_t^{(l)} s_t^{(l)}}{\sum_{l=1}^{N} G(s_t^{(n)} - s_t^{(l)}) w_t^{(l)}} \tag{10}$$

Here G is an arbitrary kernel. Mostly a Gaussian kernel is used in this method.

3.2. Better Occlusion Handling Technique

Some algorithm handles occlusion by keeping each of the temporary loss into a counter (TTL) which computes that for how many frames the target object is missing. If the counter reaches the user defined value then assumed that the object is lost and discarded [7]. After the object is discarded if it appeared again then to decide whether it is the current position of the target object, it is checked that whether the object region is overlapped with the tracked object's search area which is by default the last position of the bounding box of the target object. Sometimes the temporary loss occurs if the object moves away from its previous position. At that time the search area is expanded proportional to the number of frames where the target object was lost.

3.3. Otha Color Method

Otha, Kanade and Sakai [11] has proposed a simple transformation from RGB to $I_1 I_2 I_3$ model conversion by,

$$I_1 = \frac{1}{3}(R + G + B) \tag{11}$$

$$I_2 = \frac{1}{2}(R - B) \tag{12}$$

And

$$I_3 = \frac{1}{4}(2G - R - B) \tag{13}$$

This model is used so that the statistical relation between the RGB intensity changes can be realized in a different way than directly using the RGB intensity in the feature extraction process [12]. Now the otha color components can be used in the feature matrix construction. Ex-

$$f_k = [\, x \quad y \quad I(x,y) \quad I_x(x,y) \quad I_y(x,y) \ \ldots\ldots\ldots \ I_1 \quad I_2 \quad I_3 \,] \tag{14}$$

3.4. Salient Feature Matching

For the time of tracking of a human, all the parts are not always taken under consideration. Some parts which are different from the other or the salient parts are enough to track the target object. So some salient parts are selected as a region descriptor represented by the covariance matrices [16]. These descriptors are robust to appearance changes. A salient region can be defined by a small portion of image which has the most ability to distinguish within its local neighborhood.

The distance between two covariance matrix or $\rho(C_i, C_j)$ is calculated by the Log Euclidean distance function as it is easy to compute and less time consuming. For a particular pixel the position set is denoted by, $Y = \{y, y \in N(x)\}$ where *N(x)* is a rectangular window around *x* but without *x*. So the salient feature of *C(x)* is defined by,

$$d(x) = \min_{y \in Y} \rho(C(x), C(y)) \tag{15}$$

If *d(x)* is high that means the region *x* is very much different from its neighborhood regions and has more discriminating power than its neighborhood. Still this approach suffers from rotation problem. But the use of gradient decent method for tracking provides robust tracking of the object throughout the video [24].

3.5. Particle Filtering Approach:

It is a technique which approximate the posterior of unknown motion state x_i from the set of noisy samples $y_{1....t}=\{y_1........y_t\}$ [22]. It calculated the posterior probability density $p(x_t \mid y_{1:t})$ of the current object state x_t for all the samples for *t* time by a weighted particle sample set $\{X_t^n, \omega_t^n\}_{n=1}^N$ [17] [21]. The steps of particle filtering approach are-

Step 1: Particle Prediction

In the particle filtering tracking the state of the particle is predicted to achieve the state change by the dynamic model of the object. The objects state includes attributes like position, shape, size, velocity etc. When the target object's velocity change is not huge then the prediction model of the object can be defined by,

$$X_t = X_{t-1} + V_t + W_t \tag{16}$$

Where, W_t denotes the process noise and V_t is the velocity of the object defined by,

$$V_t = X_{t-1} - X_{t-2} \tag{17}$$

Step 2: Particle Balance and Particle Resample

Particle balance calculates the matching estimation between the observation model and the target object represented by particles. The matching measure deputizes the particles weight. Each particle's weight value is proportional to the likelihood function. If bootstrap filter algorithm is used then,

$$\omega_t^n = \omega_{t-1}^n p(y_t \mid x_t^n) \ \text{Where} \ \omega_{t-1}^n = 1/N \tag{18}$$

Here number of particles is denoted by N. The Gaussian probability function is used to define the likelihood function,

$$p(y_t \mid x_t^n) = \frac{1}{\sqrt{2\prod}\sigma} e^{-\frac{\rho^2}{2\sigma^2}} \tag{19}$$

Here, ρ can be any distance measure (discussed previously) between the observation model and the object model. If the value of ρ is small then the value of the likelihood function will be more. The parameter σ is selected carefully for the effectiveness of the particle resampling.

4. Experiment Result

We have tested all the discussed distance formulas on 14 different set of videos containing more than 2000 number of frames including moving and both stationary camera. Some of the results are shown in Table 2 and Table 3. The described algorithm and the different distance formulas are implemented in MATLAB as it is easy for error finding and debugging. We have calculated separate detection rate for all the distance matrix described. We have considered a 50 X 50 matrix to search the target object in the candidate region of the consecutive next frames. We have seen that for extremely fast object also the next location region is also within the 50 X 50 region from previous location of the target object. As if we consider the whole image then the time to search the next region will be a lot, which could reduce the robustness of the tracker. As we have considered the 50 X 50 neighborhood region so there can be 10000 possible regions for a particular target object. So the region which has the minimum covariance distance will become the new location of the object. And within the different methods which can be applied on covariance tracking these described procedures are proved to provide the most accurate and robust result.

Table 2. Tracking Performance Measurement of Bhattacharya Distance, Log-Euclidean Distance, Cholesky Decomposition

	Bhattacharya Distance		Log-Euclidean Distance		Cholesky Decomposition	
	Miss/ Total	Detection	Miss/ Total	Detection	Miss/ Total	Detection
Movie character 1	1/56	96.42	9/56	83.92	11/56	80.35
Movie Character 2	1/73	98.63	7/73	90.41	2/73	97.26
Man's face	14/86	83.72	20/86	76.74	2/86	97.67
Shuttle	1/100	99	3/100	97	1/100	99
Train station	5/93	94.62	4/93	95.7	26/93	79.56

Table 3. Tracking Performance Measurement of Root Euclidean Distance and JDBL Distance

	Root Euclidean Distance		JDBL Distance	
	Miss/Total	Detection	Miss/Total	Detection
Movie character 1	12/56	78.57	17/56	69.64
Movie Character 2	1/73	98.63	3/73	95.89
Man's face	5/86	94.18	5/86	94.18
Shuttle	2/100	98	23/100	77
Train station	22/93	76.34	26/93	72.04

(a) (b) (c)



(d) (e) (f)

Figure 1. Tracking using Bhattayacharya Distance for (a) frame 10, (b) frame 30, (c) frame 40, (d) frame 50, (e) frame 60, (f) frame 79



(a) (b) (c)



(d) (e) (f)

Figure 2. Tracking using Cholesky Decomposition for (a) frame 10, (b) frame 30, (c) frame 40, (d) frame 50, (e) frame 60, (f) frame 79



(a) (b) (c)



(d) (e) (f)

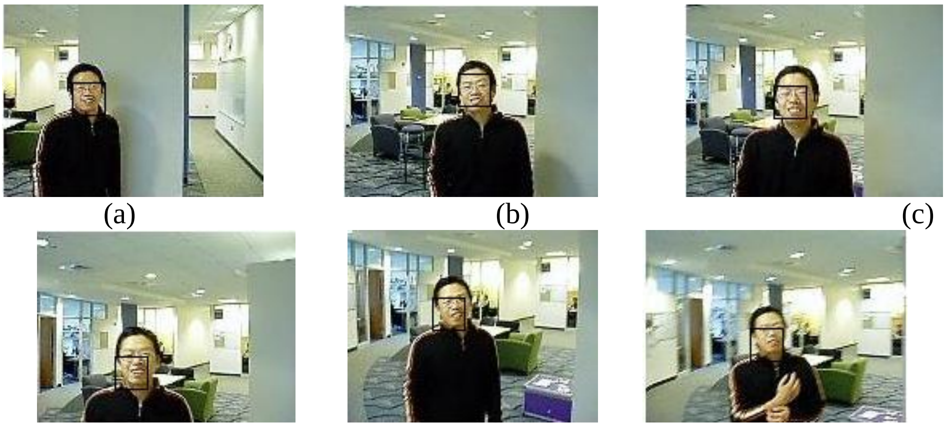
Figure 3. Tracking Using JDBL Distance for (a) Frame 10, (b) frame 30, (c) frame 40, (d) frame 50, (e) frame 60, (f) frame 79



(a) (b) (c)

(d) (e) (f)

Figure 4. Tracking using Log-Euclidean Distance for (a) frame 10, (b) frame 30, (c) frame 40, (d) frame 50, (e) frame 60, (f) frame 79



(a) (b) (c)

(d) (e) (f)

Figure 5. Tracking using Root- Euclidean Distance for (a) frame 10, (b) frame 30, (c) frame 40, (d) frame 50, (e) frame 60, (f) frame 79

5. Conclusion

Different researchers have proven that covariance tracker provides more accuracy then other trackers even if the object is moving fast. Here we have shown that covariance tracker itself is enough to track a target object but to get more accurate result different methods can be applied on covariance trackers. Apart from those methods sometimes a model update approach is also applied on covariance tracking. After applying model update approach the covariance distance will become intrinsic distance on manifold. So in future this works can be extended to a combination of model update approach with other reparameterization approaches to get more accurate and robust result. This work can be extended to multiple object tracking and better occlusion handling technique.

References

[1] S. Mahendran, D. Vaithiyanathan and R. Seshasayanan, “Object Tracking System Based on Invariant Features Proceedings of the International conference on Communication and Signal Processing”, **(2013**), pp. 1138-1142.

[2] Zhong Guo, “Object detection and Tracking in Video, Advances in Internet based system and application”, November **(2001).**

[3] T. Dharamadhat, K. Thanasoontornlerk and P. Kanongchaiyos, “Tracking object in video pictures based on background subtraction and image matching”, In Proceedings of the IEEE International Conference on Robotics and Biometrics, **(2008)**, pp. 1255-1260.

[4] D. P. Chau, F. Bremond and M. Thonnat, “Object Tracking in Videos: Approaches and Issues in The International Workshop Rencontres UNS-UD”, **(2013)** April**.**

[5] O. Tuzel, F. Porikli and P. Meer, “Region Covariance: A Fast Descriptor for Detection and Classification”, Proceedings of the European Conference on Computer Vision (ECCV), May **(2006)**.

[6] F. Porikli, O. Tuzel and P. Meer, “Covariance Tracking using Model Update Based on Lie Algebra”, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), **(2006)**.

[7] C. Spampinato, S. Palazzo, D. Giordano, I. Kavasidis, F.-P. Lin and Y.-T. Lin, "Covariance based Fish Tracking in real-life underwater environment", Proceeding of the 20th IEEE International Conference on Image Processing (ICIP), **(2013)**, pp. 1481-1485.
[8] X. Ding, C. Huang, F. Huang, L. Xu, and X.-f. Li, "Region Covariance based Object Tracking Using Monte Carlo Method", Proceedings of the 8th IEEE International Conference on Control and Automation, **(2010)**, pp. 1802-1805.
[9] W. Forstner and B. Moonen, "A metric for covariance matrices", Technical report, Dept. of Geodesy and Geoinformatics, Stuttgart University, **(1999).**
[10] H. Palaio and J. Batista, "A Kernel Particle Filter Multi-object Tracking using Gabor-Based Region Covariance Matrices", Proceedings of the 16th IEEE international conference on Image Processing (ICIP), **(2009)**, pp. 4085-4088.
[11] Y. I. Ohta, T. Kanade, and T. Sakai, "Color information for region segmentation", Computer Graphics and Image Processing, vol. 13, **(1980)**, pp. 222-241.
[12] P. P. Dash, S. Aitha and D. Patra, "Ohta Based Covariance Technique for Tracking Object in Video Scene", In Proceedings of the IEEE Students Conference on Electrical, Electronics and Computer Science, **(2012)**, pp. 1-4.
[13] I. L. Dryden, A. Kolaydenko and D. Zhou, "Non-Euclidean statistics for covariance matrices, with applications to diffusion tensor imaging", The Annals of Applied Statistics, vol. 3, no. 3, **(2009)**, pp. 1102-1123**.**
[14] A. Cherian, S. Sra, A. Banerjee, and N. Papanikolopoulos, "Efficient similarity search for covariance matrices via the jensen-bregman logdet divergence", Proceedings of the International Conference on Computer Vision (ICCV), **(2011)**, pp. 2399-2406.
[15] G. Somasundaram, V. Morellas, N. Papanikolopoulos and S. Bedros, "Object Classification in Traffic Scenes using Multiple Spatio-Temporal Features", Proceedings of the 20th Mediterranean Conference on Control & Automation (MED), **(2012)**,pp. 1536-1541.
[16] H. Hu, B. Ma, Q. Ma and W. Liang, "Covariance Based local salient Descriptors for visual tracking", In Proceedings of the IEEE International conference on Multimedia and Expo, **(2013)**, pp. 1-6.
[17] P. Wu, J. Wang and Q. Zhang, "Covariance Tracking Algorithm Based on Particle Filter and Adaptive Template Update", Proceedings of the 2nd International Conference on Education Technology and Computer (ICETC), **(2010)**, pp. 199-202.
[18] M. Donoser, S. Kluckner and H. Bischof, "Object Tracking by Structure Tensor Analysis", International Conference on Pattern Recognition, **(2010)**, pp. 2600-2603.
[19] P. C. Cargill, D. M. Quiroz and L. E. Sucar, "Object Tracking based on Covariance Descriptors and On-Line Naive Bayes Nearest Neighbor Classifier in Fourth Pacific-Rim Symposium on Image and Video Technology", **(2010)**, pp. 139-144.
[20] M. Azari, A. Seyfi, A. H. Rezaie, "Real Time Multiple Object Tracking and Occlusion Reasoning Using Adaptive Kalman Filters", Proceedings of the Machine vision and Image Processing, **(2011)**, pp. 1-6.
[21] P. Wu, J. Wang and Q. Zhang, "Covariance Tracking Algorithm Based on Particle Filter and Adaptive Template Update", Proceedings of the 2nd International Conference on Education Technology and Computer (ICETC), **(2010)**, pp. 199-202.
[22] Xinying Liu, Particle Filter Object Tracking Based on SIFT-Gabor Region Covariance Matrices, Proceedings of the IEEE International Intelligent control, Automatic Detection and High-End Equipment (ICADE), **(2012)**, pp. 201-204.
[23] A. R. M. y Ter´an, A. H. Zahraee and M. Gouiff`es, "Real-time covariance tracking algorithm for embedded systems", Proceedings of the Conference on Design and Architecture for Signal and Image Processing (DASIP), **(2013)**, pp.104-111.
[24] A. Tyagi, J. W. Davis and G. Potamianos, "Steepest Descent For Efficient Covariance Tracking in IEEE Workshop on Motion and video Computing", WMVC, **(2008)**, pp.1-6.
[25] S. Zheng, H. Qiao, B. Zhang and P. Zhang, "The Application of Intrinsic Variable Preserving Manifold Learning Method to Tracking Multiple People with Occlusion Reasoning", Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, **(2009)**, pp. 2993-2998.
[26] D. Chen, J. Han and Z. Yu, "Visual Tracking Using Region Covariance And Geometric Particle Filtering", in 3rd International Conference on Image and Signal Processing (CISP2010), **(2010)**, pp. 381-386.
[27] B. Jeon, K. Baek, C. Kim and H. Bang, "Mode Changing Tracker for Ground Target Tracking on Aerial images from Unmanned Aerial Vehicles", Proceedings of the 13th International Conference on Control, Automation and Systems, **(2013)**, pp. 1849-1853.
[28] S. Bak, D.-P. Chau, J. Badie, E. Corvee, F. Bremond and M. Thonnat, "Multi-Target Tracking by Discriminative analysis on Riemannian Manifold", Proceedings of the 19th IEEE International Conference on Image Processing (ICIP), **(2012)**, pp. 1605- 1608.
[29] P. C. Cargill, C. U. Rius, D. M. Quiroz and A. Soto, "Performance evaluation of the covariance descriptor for target detection", Proceedings of the International Conference of the Chilean Computer Science Society, **(2009)**, pp. 133-141.

[30] S. Youn and C. Lee, "Edge Detection for Hyperspectral Images Using the Bhattacharyya Distance", Proceedings of the IEEE International Conference on Parallel and Distributed System, **(2013)**, pp. 716 - 719.

Authors

Sanchita Singha, She is presently pursuing her M.Tech degree in Computer Science and Engineering Dept from School of Computer Engineering, KIIT University, India.

Sujoy Datta, He has completed, his M.Tech in Computer Science and Data Processing from IIT Kharagpur, India.