# An Intelligent Visual Tracking System Based on Adaptive Mean Shift Tracking Method

P. Bhuvaneswari
PG Scholar of Electronics and Communication Engineering,
Anna University Regional Office,
Madurai 7, Tamil Nadu, India.

S.Veluchamy
Faculty of Electronics and Communication Engineering,
Anna University Regional Office,
Madurai 7, Tamil Nadu, India.

***Abstract*— Generally all sparsity based models yields tracking performance in impressive manner, however computational complexity and low estimation of scale and orientation changes of target is aggravated in motion blur and background clutter challenges. In this paper, proposed method is based on a framework which projects templates matrix in candidates space. By selecting and weighting sparse coefficients, DSS map with pooling method leads to choose best candidate for tracking and by scale and orientation adaptive mean shift tracking method we can estimate scale, orientation of target adaptively. The result shows that better accuracy in tracking and robust to above challenges.**

***Keywords*— Customized APG, DSS map and Adaptive Mean Shift Tracking.**

## I. INTRODUCTION

An object in image processing is an identifiable portion of an image that can be interpreted as a single unit. Object representation is the shape and appearance representation. Object recognition is the task of finding a given object in an image or video sequence. Visual Tracking is the process of locating and determining the dynamic configuration of one or many moving objects in each frame of one or several cameras. It is to associate targets in consecutive video frames. It can be especially difficult when the objects are moving fast relative to rate of frame.

For these situations video tracking systems usually employ a model which describes how the image of the target might change for different possible motions of the object. To perform video tracking, an algorithm must analyze consecutive video frames and outputs of the movement of targets between frames. There are so many algorithms, each having strengths and weaknesses. Two major components of a visual tracking system: target representation and localization, and also filtering and data association. Applications are Motion-based recognition: human identification, automatic object detection, etc. Although we have variety of sparsity methods for tracking, we have a situation that increases the complexity of problem when the tracked object changes orientation over time.

### *A. Related Research Work*

Using patch-based appearance models [2], we can divide bounding box of a target into multiple patches and then selecting pertinent patches to construct appearance model. Due to this, we can obtain accurate tracking but it has more computational complexity than [4], in which online tracking involved and [7], in which 2D principal component analysis involves and in matter of solving whole object occlusion problem, this is not very efficient. This computational complexity decreases in[10], but here it is difficult to yield a sequence of closed form of updates. In [9], Block Orthogonal Matching Pursuit (BOMP) algorithm, involves to facilitates tracking performance compared with prototype structured sparse representation model, but there is difficult to track fully occluded objects. By using Monte Carlo Tracking Technique [8], we can handle complete occlusion, drawback here is tough to locate position of target when large pose variations and drastic illumination changes occurs. In paper [11], Collaborative model (SDC and SGM) used here to handle occlusion, it alleviate drift problem also and in paper [12], sparse approximation problem in particle filter framework is used to get excellent performance with previously proposed trackers. The same particle filter framework with reverse sparse representation used in paper [1], results tracking object more superior in terms of cost-performance ratio. but here scale and orientation changes cannot be adaptively estimated. In paper [3], for non rigid objects, visual tracking is obtained by spatial masking with an isotropic kernel. But disadvantage is tracking specific task alone and in paper [6], Mean Shift Method used to get object shape well tracked and not completely solved for complex sequences. Hence we can go for SOAMST (Scale and Orientation Adaptive Mean Shift Tracking) algorithm in paper [5], we can obtain scale and orientation changes of target but low performance in tracking when motion blur and heavy occlusion occurs, hence through research, we can use this concept of AMST algorithm in analysis of DSS map visual tracking, we can get accurate tracking results even in motion blur and background clutter.

This paper is organized as follows. The proposed system includes formation of DSS (Discriminative Sparse Similarity) map and establishing AMST algorithm in section II. Section III gives description of datasets used to test proposed method and performance analysis described in section IV. Section V illustrates experimental results and discussions. Finally, conclusion given is section VI.

## II. PROPOSED SYSTEM

Through the above analysis, tracking of object will be complex when noise in images, rapid object motion, scene illumination changes occurs. Hence proposed method is used to handle motion blur and background clutter robustly in an object tracking with scale and orientation changes. The object is tracked based on reversed multitask sparse tracking framework which gives matrix templates into candidate space. By selecting and weighting sparse coefficients DSS map, pooling method projects best candidate and by SOAMST algorithm, we can adaptively estimate scale and orientation of target leads to achieve accuracy in tracking an object. The block diagram of the proposed system is shown below. In first stage, input video is selected for tracking and recognition process and then frame conversion is done.

After that formulate multitask reverse sparse representation and then include laplacian term to keep coefficients similarity level in accordance with candidate similarity. From this we can get laplacian multitask reverse sparse representation and DSS map is constructed based on that similarity relationship that contains large template set combined by multiple positive and hundreds of negative templates. Then additive pooling approach is done. Finally, Weight image obtained from target model and target candidate model to estimate target scale and orientation. Using this we can compute moment features and then estimate width, height, orientation of object based on zeroth order moment, second order center moment and Bhattacharyya coefficient between target and target candidate model. By establishing this AMST algorithm in DSS map method, it provides good estimation accuracy of location of target with scale and orientation changes as output.

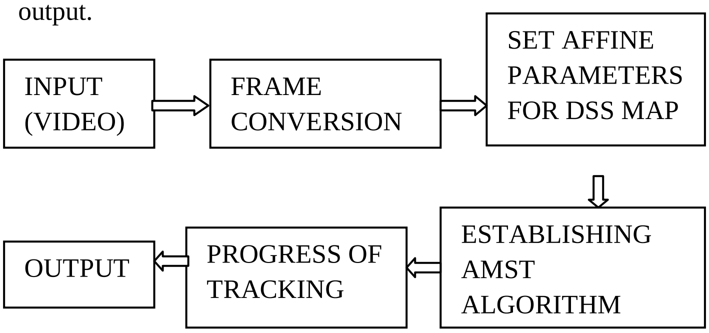

INPUT (VIDEO)
FRAME CONVERSION
SET AFFINE PARAMETERS FOR DSS MAP
ESTABLISHING AMST ALGORITHM
PROGRESS OF TRACKING
OUTPUT

Figure1. Block diagram for proposed method

### *A. Bayesian Interference Framework*

It is a framework for estimating posterior distribution of the state variables that characterize dynamic system, to provide robust tracking algorithm. Let observation set of target $\mathbf{A}t$ = [$\mathbf{A}_1$, $\mathbf{A}_2$,…, $\mathbf{A}_t$], and let $\mathbf{Z}_\mathbf{t}$ be the state variable of an object at time *t*. Then we use affine transformation to motion of object between two consecutive frames. Then optimal state $\mathbf{Z}_\mathbf{t}$ can be computed by maximum posterior estimation as,

$$\hat{\mathbf{Z}}_t = \underset{\mathbf{Z}^i_t}{\arg\max}\, p\,(\mathbf{Z}^i_t\,|\mathbf{A}t\,) \qquad (1)$$

where $\mathbf{Z}^i_t$ indicates the state of the *i -th* sample. The posterior probability can be obtained from the Bayesian framework recursively as,

$$p(\mathbf{Z}_t|\mathbf{A}_t) \propto p(\mathbf{A}_t\ |\mathbf{Z}_t\ )p(\mathbf{Z}_t\ |\mathbf{Z}_t{-}1)p(\mathbf{Z}_\mathbf{t}{-}1|\mathbf{A}_t{-}1)b\mathbf{x}_t{-}1 \qquad (2)$$

where $p(\mathbf{Z}_\mathbf{t}\ |\mathbf{Z}_\mathbf{t}{-}1)$ is the dynamic model and $p(\mathbf{A}_t\ |\mathbf{Z}_\mathbf{t}\ )$ denotes the observation likelihood. The state variable $\mathbf{Z}_\mathbf{t}$ is composed of six independent parameters {$\Phi_1$, $\Phi_2$, $\Phi_3$, $\Phi_4$, $\ulcorner_1$, $\ulcorner_2$**},** in which {$\Phi_1$, $\Phi_2$, $\Phi_3$, $\Phi_4$} are deformation parameters , **{**$\ulcorner_1$, $\ulcorner_2$**}** contain 2D transformation information. Finally, dynamic model can be modeled by the Gaussian distribution,
as,

$$p(\mathbf{Z}_t\ |\mathbf{Z}_{t\text{-}1}) = N(\mathbf{Z}_t\ ;\ \mathbf{Z}_{t\text{-}1,}\ \sigma) \qquad (3)$$

where $\sigma$ represents diagonal covariance matrix whose elements are variances of parameters. By this method, we get the candidates set $\mathbf{X}$ = [$\mathbf{X}_1$, $\mathbf{X}_2$, ..., $\mathbf{X}_m$] ∈ $R^{bXc}$, in which *b* is the feature dimension and *c* is the number of candidates. The observation model $p(\mathbf{A}_t\ |\mathbf{Z}_t\ )$ essentially reflects the likelihood of observing $\mathbf{A}_t$ at state $\mathbf{Z}_t$ . In this paper, $p(\mathbf{A}_t\ |\mathbf{Z}_t\ )$ is proportional to discriminative score get by exploiting the additive pooling scheme on the DSS map.

### *B. Discriminative Reverse Sparse Representation*

In conventional sparse representation, an observed image patch associated with state is reconstructed by numerous target templates, we can construct candidate set $\mathbf{X}$ to represent each target template as in equation (4) below.

$$\underset{m}{\arg\min}\ \|\mathbf{t} - \mathbf{Xc}\|_2^2 + \lambda\|\mathbf{m}\|_1,$$
$$\text{s.t.}\mathbf{m}\geq 0 \qquad (4)$$

where **t** denotes representative template and $\lambda$ is parameter to adjust sparsity penalty and m represents coefficient vector. Thus, we add constraint entry, m ≥ 0, which means all elements of **m** are nonnegative for the reason that each element represents the similarity between template and candidate, and negative elements are neglected.

Using the L1 minimization, tracker be efficient but lack of negative templates makes discriminative power poor for ignoring the background information around target, which may results the tracker drift away from target.

Hence multiple positive target templates are exploited to make the tracker more responsive. So, we use numerous negative templates, which are capable of sketch out the periphery of target area. The positive and negative template sets are respectively defined as $\mathbf{L}_{pos}$ = [$\mathbf{l}_1$, $\mathbf{l}_2$, ..., $\mathbf{l}_p$] and $\mathbf{L}_{neg}$ = [$\mathbf{l}_{\mathbf{p+1}}$, $\mathbf{l}_{p+2}$, ..., $\mathbf{l}_{p+n}$], where *p* and *n* denote the number of positive and negative templates respectively.

With the assumptions above, our problem formulation is expressed by finding the combination of particles and corresponding coefficients as following:

$$\begin{cases} \arg\min_{\mathbf{m}_1} \|\mathbf{t}_1 - \mathbf{X}\mathbf{m}_1\|_2^2 + \lambda\|\mathbf{m}_1|_1|\ldots\ldots \\ \arg\min_{\mathbf{m}_p} \|\mathbf{t}_p - \mathbf{X}\mathbf{m}_\mathbf{p}\|_2^2 + \lambda\|\mathbf{m}_\mathbf{p}\|_1 \\ \arg\min_{\mathbf{m}_{p+1}} \|\mathbf{t}_{p+1} - \mathbf{X}\mathbf{m}_{\mathbf{p+1}}\|_2^2 + \lambda\|\mathbf{m}_{p+1}\|_1\ldots\ldots \\ \arg\min_{\mathbf{m}_{p+n}} \|\mathbf{t}_{p+n} - \mathbf{X}\mathbf{m}_{\mathbf{p+n}}\|_2^2 + \lambda\|\mathbf{m}_{\mathbf{p+n}}\|_1 \end{cases} \quad (5)$$

where $\mathrm{m}_\mathbf{i} = [\mathbf{m}_\mathbf{i}^\mathbf{1}, \mathbf{m}_\mathbf{i}^\mathbf{2}, \ldots, \mathbf{m}_\mathbf{i}^\mathbf{h}]^T$ expresses the sparse coefficients of the i -th template and $\mathbf{m}_i \geq 0$, i = 1, 2, . . . ,(p + n*)* means all the elements in ***mi*** are nonnegative. An illustration of the basic idea of this formulation shown below.

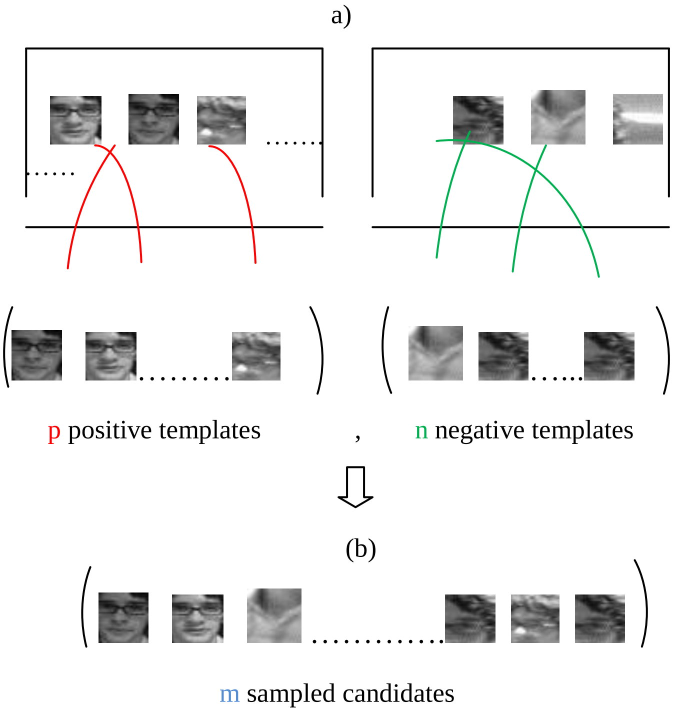


Figure2. The basic idea of the multi-task reverses sparse representation scheme illustration. (a) The positive and the negative template sets, (b) The sampled candidates.

*C. Laplacian Multi-task Reverse Sparse Representation*

The multi-task reverse sparse representation problem start from the equation as,

$$\arg\min \|\mathbf{L} - \mathbf{XC}\|_2^2 + \lambda\sum_i \|\mathbf{c}_i\|_1$$

$$\text{s.t. } \mathbf{c}i \geq 0,\ i = 1, 2, \ldots, (p+n). \quad (6)$$

To preserve the similarity of codes for candidate features, we introduce a Laplacian regularization term. It is begin with formula as,

$$\arg\min_{c} \|\mathbf{L} - \mathbf{XC}\|_2^2 + \lambda\sum_{I} \|\mathbf{c}_i\|_1 + \delta/2\sum_{ij} \|\mathbf{c}_i - \mathbf{c}_j\|^2 B_{ij}$$

$$\text{s.t. } \mathbf{c}i \geq 0,\ i = 1, 2, \ldots, (p+n), \quad (7)$$

where $\delta$ is parameter to adjust new regularization term and **B** is a binary matrix. So, the Laplacian multi-task reverse optimization problem is reformulated as:

$$\arg\min_{c} \|\mathbf{L} - \mathbf{XC}\|_2^2 + \lambda \sum_{i} \|\mathbf{c}_i\|_1 + \delta \mathrm{tr}(\mathbf{CZC}^T)$$

$$\text{s.t. } \mathbf{c}i \geq 0,\ i = 1, 2, \ldots, (p+n). \quad (8)$$

where **Z** = **D**−**B** is the Laplacian matrix , and the degree of $\mathbf{c}_\mathbf{i}$ is defined as,

$$D_i = \sum_{j=1}^{p+n} Bi\,j \text{ and}$$

$$\mathbf{D} = diag\,(D_1, D_2, \ldots, D_{p+n}) \quad (9)$$

At last apply the accelerated proximal gradient (APG) approach to solve minimization problem with,

$$F(\mathbf{C}) = \arg\min_{c} \|\mathbf{L} - \mathbf{XC}\|_2^2 + \lambda\mathbf{1}^T\mathbf{C1} + \delta \mathrm{tr}\,(\mathbf{CZC}^T)$$

$$G(\mathbf{C}) = \psi(\mathbf{C}) \quad (10)$$

where *F*(**C**) is differentiable convex function and *G*(**C**) is non-smooth convex function. Following the APG method, we need to solve an optimization problem as:

$$\mu_{k+1} = \arg\min_{c} \xi/2\|\mathbf{C} - \varepsilon_{k+1} + \nabla F(\varepsilon_{k+1})/\xi\|_2^2 + G(\mathbf{C}) \quad (11)$$

where $\xi$ is the Lipschitz constant and finally, equation (11) is equivalent to,

$$\mu_{k+1} = \max(\mathbf{0}, \mathbf{g}_{k+1}) \quad (12)$$

*D. Refined DSS map*

To avoid potential instability and achieve better robustness, we refine the DSS map with adaptive weights. The weight $B_{ij}$ for an element $X_{ij}$ in the similarity map is constructed based on the difference between the *j* -th candidate $\mathbf{y}_j$ and the *i* –th template $\mathbf{t}_i$ :

$$B_{ij} \propto \exp(-\|\mathbf{t}_i - \mathbf{y}_j\|_2^2). \quad (13)$$

A candidate with small difference from foreground template indicates they share higher similarity with each other. For this, we separate the weight map into two submaps:

$$\mathbf{B}_{pos} = [\mathbf{b_1}^\mathbf{T}, \ldots, \mathbf{bp}^\mathbf{T}]^\mathbf{T},$$

$$\mathbf{B}_{neg} = [\mathrm{b}_{\mathbf{p+1}}{}^\mathbf{T}, \ldots, \mathbf{b}_{\mathbf{p+n}}]^\mathbf{T}, \quad (14)$$

where $\mathbf{b}_1 = [Wi1, \ldots Wim]$ for $i = 1, 2, \ldots, (p+n)$

Finally, get two weighted DSS maps through:

$$\tilde{\mathbf{X}}_{pos} = \mathbf{B}_{pos} \odot \mathbf{X}_{pos},$$

$$\tilde{\mathbf{X}}_{neg} = \mathbf{B}_{neg} \odot \mathbf{X}_{neg}, \quad (15)$$

where ʘ is the Hadamard product.

*E. Additive Pooling*

It contains 2 steps follows:

**Step 1:** positive and negative templates (Equation (1)and(2) respectively) brought from similarity map ˜X (Eqn(3)) as,

$$xi{-}pos = [\,\tilde{}X1i, \ldots, \tilde{}Xpi] \quad (16)$$

$$xi{-}neg = [\,\tilde{}X(p+1)i, \ldots, \tilde{}X(n+p)i\,] \quad (17)$$

$$xi = [\,\tilde{}X1i, \ldots, \tilde{}Xpi, \tilde{}X(p+1)i, \ldots, \tilde{}X(n+p)i] \quad (18)$$

then add largest *l* coefficients in *˜xi−pos* and *˜xi−neg* to get,

yi−pos and yi−neg as,

$$yi{-}pos = L(\tilde{}xi{-}pos, 1)+, \cdots + L(\tilde{}xi{-}pos, l) \quad (19)$$

$$yi{-}neg = L(\tilde{}xi{-}neg, 1)+, \cdots + L(\tilde{}xi{-}neg, l) \quad (20)$$

**Step 2:** discriminative score and score set for all is given by,

$$yi = yi{-}pos - yi{-}neg$$
$$Y = \{yi\},\ \ i=1,...,m \qquad (21)$$

*F. Adaptive Mean Shift Concept*

In the mean shift iteration, the estimated target moves from y to new position y1, which is given as,

$$y1 = \frac{\sum_{i=1}^{n_h} x_i w_i g(\|y\text{-}xi/h\|2)}{\sum_{i=1}^{n_h} x_i w_i g(\|y\text{-}xi/h\|2)} \qquad (22)$$

By using Equation(22), the mean shift tracking algorithm finds object in the new frame that most similar to the object in reference frame. Note that the key parameters in the mean shift tracking algorithm are the weights *iw*. Estimate the target area by using equation as,

$$M_{00} = \sum_{i=1}^{n} w(xi) \qquad (23)$$

The Bhattacharyya coefficient is used to adjust $M_{00}$ in estimating target area, denoted by *T*. We propose the following equation to estimate it:

$$T= b(\rho)M_{00} \qquad (24)$$

where $b(\rho)$ is monotonically increasing function with respect to the Bhattacharyya coefficient $\rho(\ 0 \le \rho \le 1)$. From the moment features, the Covariance matrix in equation can be written as,

$$\text{Cov} = \begin{bmatrix} \mu_{20}\,\mu_{11} \\ \mu_1\mu_{02} \end{bmatrix} \qquad (25)$$

Suppose that the target is represented by an ellipse, for which the lengths of the semi-major axis and semi-minor axis are denoted by *a* and *b*, respectively.

Instead of using λ1 and λ2 directly as the width *a* and height *b*, it has been shown that the ratio of λ1 to λ2 can well approximate the ratio of *a* to *b*, i.e., $\lambda1/\ \lambda2 \approx a\ b$ . Thus we can set $a = k\lambda1$ and $b = k\lambda2$ , where *k* is a scale factor. Since we have estimated the target area *A*, there is $\pi ab = \pi\ (k\lambda1)(\ k\lambda2) = A$ . Then it can be easily derived that,

$$K=\sqrt{}A/(\ \pi\ \lambda1\ \lambda2) \qquad (26)$$
$$a=\sqrt{}\lambda1A/(\ \pi\ \lambda2)\ \ ,\ \ b=\sqrt{}\ \lambda2A/(\ \pi\ \lambda1) \qquad (27)$$

Now the covariance matrix becomes,

$$\text{Cov} = \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix} \text{x} \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix} \text{x} \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix}^T \qquad (28)$$

From this we can estimate width, height and orientation of target.

## III. DESCRIPTION OF DATASETS

The algorithm is tested using three following datasets.

*Dataset 1:* In palm sequence video, tracking is done clearly with scale and orientation changes. As target undergoes abrupt motion, it is difficult to locate its position. However when palm is moving rapidly in frames 25 and 92, the estimated target scale and orientation are accurate and robust to motion blur challenge.

*Dataset 2:* In car sequence video, there is a complex background in frames 38, 50. By introducing both template sets to model foreground and background information, we get enough discriminative information and store them in map. And by AMST algorithm we obtained accurate tracking results with scale and orientation changes and robust to this background clutter.

*Dataset 3:* In walking man sequence, both in-plane and out-of-plane rotations occurs. However this tracker clearly tracks and gives scale and orientation changes successfully.

## IV. PERFORMANCE ANALYSIS

The table shown below lists average number of iterations by three different datasets on three video sequences.

Table I. Performance on average number of iterations.

| Datasets | Adaptive scale | EM-Shift | Our method |
|---|---|---|---|
| Palm sequence | 14.62 | 6.52 | 3.30 |
| Car sequence | 11.25 | 6.27 | 2.34 |
| Walking man sequence | 13.43 | 6.35 | 2.59 |

The adaptive scale is the highest because it runs algorithm three times and our method estimates only one time for each frame. Hence it is faster than others.

Another table below gives TAR (True Area Ratios) values by three trackers on three datasets. TAR value is given by the ratio of overlapped area between tracking result and ground truth to area of ground truth.

Table II. TAR values by competing tracking methods.

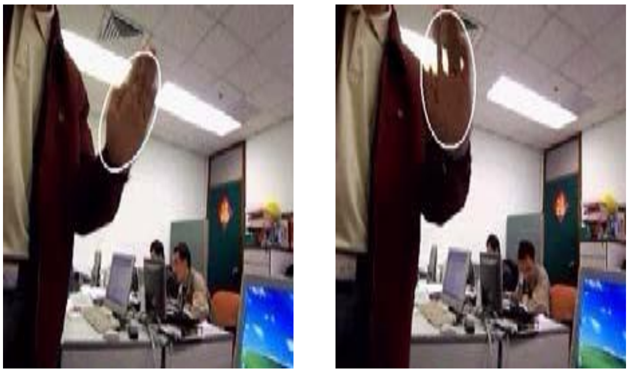
| Datasets | OWN | EM-SHIFT | OUR METHOD |
|---|---|---|---|
| Palm sequence | 60% | 85.82% | 96.51% |
| Car sequence | 85.63% | 90.30% | 91.84% |
| Walking man sequence | 72.5% | 64.23% | 89.53% |

Our proposed method achieves better results than others in above table II comparison.

## V. RESULTS AND DISCUSSIONS

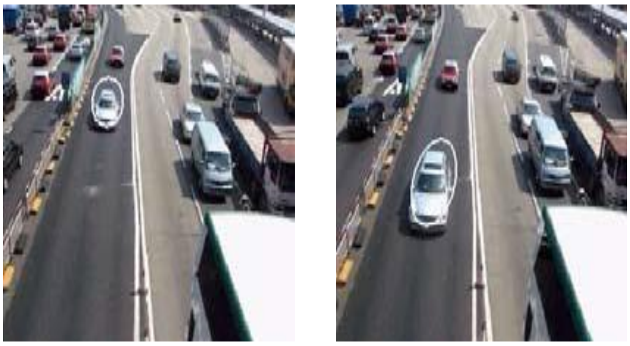
The proposed algorithm is implemented by high level language MATLAB and the hardware requirements such as, Pentium IV 2.4 GHZ processor, 512 MB RAM, 2.0 GHZ processor speed and windows XP software.

Tracking results for three datasets are provided below.



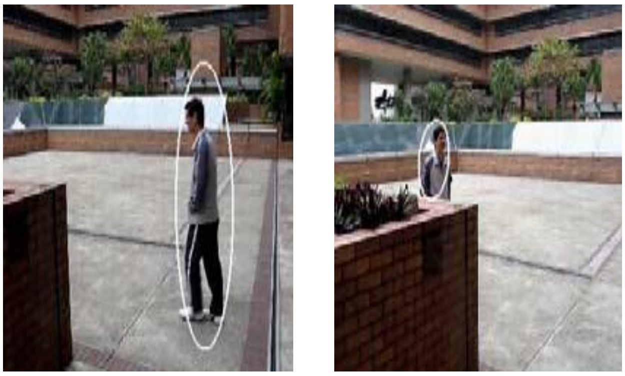
Frame number 25 Frame number 92

Figure 1. Tracking results for palm sequence.



Frame number 38 Frame number 50

Figure 2. Tracking results for car sequence.



Frame number 58 Frame number 100

Figure 3. Tracking results for walking man sequence.

In figure 1, palm sequence getting tracked and here motions are rapid. Even though in motion blur we are getting better and clear tracking with scale and orientation changes.

In figure 2, car sequence has been tracked effectively even when background clutter occurs.

In figure 3, walking man sequence tracked effectively even in occlusion.

## VI. CONCLUSION

In this paper, an intelligent visual tracking based on adaptive mean shift algorithm proposed. The DSS map represented as matrix templates to get tracking and adaptive mean shift concept used here to estimate scale and orientation changes of target. This proposed method achieves better accuracy in tracking with an exact scale and orientation changes even in challenges like motion blur, background clutter, etc.,

## REFERENCES

[1] Bohan Zhuang, Huchuan Lu, Ziyang Xiao, and Dong Wang, “Visual Tracking via Discriminative Sparse Similarity Map”, IEEE Transactions on Image Processing, Vol.23, no.4, 1872-1881, 2014.

[2] Dae-Youn Lee, Jae-Young Sim, Chang Su Kim, ”Visual Tracking Using Pertinent Patch Selection and Masking”, Computer Vision Foundation, 2014.

[3] Dorin Comaniciu, Visvanathan Ramesh, Peter Meer, “Kernal-Based Object Tracking”, TEEE Transactions, Pattern Analysis and Machine Intelligence, Vol.25, no.5, 564-577, 2003.

[4] Feng Chen, Qing Wang, Song Wang, Weidong Zhang, Wenli Xu, “Object Tracking Via Apperance modeling and Sparse Representation”, Image and Vision Computing, 29, 787-796, 2011.

[5] Jifeng Ning, lei Zhang, David Zhang and Chengke Wu, “Scale and Orientation Adaptive Mean Shift Tracking “, National Sceince Foundation Council of China under Grants and the Chinese University Sceintific Fund under Grant No.QN2009091, 1-23, 2009.

[6] Katharina Quast and Andre Kaup, “Scale and Shape Adaptive mean Shift Object Tracking”, Paper on 17th European Signal Processing Conference, 1513-1517, 2009.

[7] Ming Li, Fang Lan Ma, Fuzhong Nian, “Robust Visual Tracking Via Apperance Modeling and Sparse Representation”, Journal of Computers, Vol.9, No.7, 2014.

[8] P.Perez, C.Hue, J.Vermaak, M.Gangnet, ”Color-based probabilistic tracking”, European Conference paper on Computer Vision (ECCV), 661-675, 2002.

[9] Tianxiang Bai, Youfu Li, ”Robust Visual Tracking Using Flexible Structured Sparse Representation”, IEEE Transactions on Industrial Informatics, Vol.10, No.1, 2014.

[10] Tianzhu Zhang, Bernard Ghanem, Si Liu, Narendra Ahuja, ”Robust Visual Tracking via Structured Multi-Task Sparse Learning,” Published on Springer Science and Business Media New York, Computer Vision,101:367–383, 2013.

[11] Wei Zhong, Huchuan Lu, Ming-Hsuan Yang, ” Robust Object Tracking via Sparsity-based Collaborative Model,” IEEE Transactions, 1838-1845, 2012.

[12] Xue Mei, Haibin Ling, ”Robust Visual Tracking using L1 Minimization”, 12th International Conference on Computer Vision (ICCV), 1436-1443, 2009.

[13] M. Yang, Y. Wu, and G. Hua, “Context-aware visual tracking,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 7, pp. 1195–1209, Jul. 2009.

[14] G. Hager, M. Dewan, and C. Stewart, “Multiple kernel tracking withSSD,” in Proc. Conf. Comput. Vis. Pattern Recognit., pp. 790–797, 2004.

[15] B. Babenko, M.-H. Yang, and S. Belongie, “Visual tracking with online multiple instance learning,” in Proc. Conf. Comput. Vis. Pattern Recognit., pp. 983–990, 2009.