Defence Science Journal, Vol. 69, No. 5, September 2019, pp. 495-502, DOI : 10.14429/dsj.69.13042

Accurate Tracking of Manoeuvring Target using Scale Estimation and Detection

Zahir Ahmed Ansari[#,*], M.J. Nigam[@], and Avnish Kumar[#]

[#]*DRDO-Instruments Research & Development Establishment, Dehradun - 248 008, India*
[@]*Indian Institute of Technology, Roorkee - 247 667, India*
[*]*E-mail: acadzahir@gmail.com*

ABSTRACT

Camera zoom operation and fast approaching/receding target causes scaling of acquired target in video frames. Fast moving target manifests in large inter-frame motion. In general, non-uniform background degrades performance of tracking algorithms. Fast Fourier transform (FFT)-based Correlation algorithms improve tracking in this scenario, but their applications is limited to small inter-frame motion. Increasing search region has implication on execution speed of the algorithms. Rapid target scaling, non-uniform background and large inter-frame motion of target hinder accurate and long term visual tracking. These challenges have been addressed for extended target tracking by augmenting fast discriminative scale space tracking (fDSST) algorithm with probable target location prediction and target detection. Localisation of fast motion has been achieved by applying fused outputs of Kalman filter and quadratic regression based prediction before applying fDSST. It has helped in accurate localisation of fast motion without increasing search region. In each frame, target location and size have been estimated using fDSST and further refined by target detection near this location. Smoothing and limiting of trajectory and size of detected target has enhanced tracking performance. Experimental results show considerable improvement of precision, success rate and centre location error tracking performance against state-of-the-art trackers in stringent conditions.

Keywords: Visual tracking; Target detection; Prediction; Correlation filter; Quadratic regression; extended Kalman filter; Target scaling

1. INTRODUCTION

Tracking is the process of finding the dynamics and following a moving target over time[1]. In generic visual tracking, only the initial location and size of a target is known and the trajectory of a target in an image sequence acquired from camera is estimated. Tracking approaches can be classified as discriminative[1,3] or generative[4-6] methods. In generative approach, target appearance is described using statistical models or templates and it is matched in new frame to locate the target. But in discriminative approaches machine learning techniques are applied to differentiate between the target appearance and the surrounding background. A search region is defined to locate the target based on expected motion profile. Faster targets cause larger inter-frame motion, so they need larger search region. Search region affects execution speed of tracking algorithms.

Since camera has limited field-of-view (FOV), target will move out of frame if camera is not moved to follow the target. Camera is mounted on the set of gimbals and close loop tracking is used to follow target over wide field-of-regard (FOR). Automatic video tracker (AVT) calculates tracking error with respect to the aim point. This tracking error is suitably modified with track loop controller and given as steering command to stabilisation loop to move camera such that target is always around aiming point. One of the advantages of close loop tracking is that inter-frame motion reduces due to following of target by camera steering. So, for the fast moving target also a smaller search region will be required.

2. RELATED WORK

Visual tracking is an extensively researched topic. It involves[1] feature extraction, target representation, target localisation and track maintenance. Most common tracking algorithms[2,3] are: Normalised cross-correlation, centroid trackers, KLT tracker, mean shift tracker, L1 tracker, multiple instance learning[7], incremental visual tracking (IVT) and discriminative correlation filters (DCF) based trackers.

Due to fast execution, FFT-based discriminative correlation filters (DCF) are successfully applied to visual tracking[4-6,8]. They have demonstrated the capability of accurate target localisation in challenging tracking scenarios. But initially they were very slow in execution. First quantum improvement in execution rate was proposed by Bolme[5], *et al.*, who trained the correlation filter by minimising the total squared error between the actual and the sharp desired correlation output. Circular correlation was computed efficiently using only FFTs and point wise operations. Henriques[9], *et al.* have used least squares regressor (ridge regression) and achieved fast kernelised correlation filters. To improve the tracking performance, multidimensional features[10] are being used where multi-channel filters are trained. Impediment of tracking

Received : 03 May 2018, Revised : 01 June 2019
Accepted : 27 June 2019, Online published : 17 September 2019

execution rate due to increase in feature dimension has been addressed by approximate formulations for learning multi-channel filters[4,6].

Most DCF based methods mainly focus on the problem of translation motion estimation. They assume very slow change in scale of the target. A multi-resolution extension of a kernelised correlation translation filter for translation and scale estimation was proposed by Li and Zhu[11]. Generally, scale estimation is combined in DCF based tracking[12] either by joint scale space filter or multi-resolution translation filter. Both schemes have high computational cost.

Danelljan[8], *et al.* have proposed Discriminative Scale Space Tracking (DSST) where a novel scale adaptive tracking approach by learning separate discriminative correlation filters for translation and scale estimation have been used. Sub-grid Interpolation of Correlation Scores in translational motion estimation and dimensionality reduction in scale estimation has helped them further reduce the computational cost of DSST approach without sacrificing its robustness and accuracy. The resulting fast DSST tracker (fDSST)[12] capable of translational motion and scale estimation has been deliberated in 3.1.

This paper proposes a tracking algorithm of manoeuvring target by applying fused outputs of Kalman filter and quadratic regression based prediction before applying fDSST. It also incorporates target detection for accurate target localisation and scale estimation.

3. PROBLEM STATEMENT AND PROPOSED SCHEME

FFT-based correlation filters have promising performance and high execution rate. Fast moving targets like fighter planes have large inter-frame motion and change shape sharply. This requires fast learning of correlation filter without being affected by model drift. Accurate estimation of scale variations is a challenging problem. For estimating scale along with translational motion, performing exhaustive search at multiple resolutions of template is computationally complex. In real-time applications, computational efficiency is also a crucial factor.

In this paper, a tracking frame work to achieve accurate tracking of fast and manoeuvring target in real time has been proposed as shown in Fig. 1. Proposed scheme has four basic units: Tracking, prediction, detection and smoothing. Proposed scheme has been implemented and tested experimentally. Target location (θ_t) and scale (S_t) estimation has been explained in subsequent subsections.

3.1 Tracking Algorithm

Tracking is a procedure for localisation of the target. fDSST has performed very well[12] and has high execution rate for this purpose. Therefore, it has been used as basic tracking algorithm in the proposed framework. fDSST is a multi-feature based DCF tracking algorithm capable of translational motion and scale estimation. Simplified block representation of fDSST has been given in Fig. 1.

3.2 Trajectory Prediction

In fDSST, the search sample z_t in current frame t is extracted at estimated target location in the previous frame. This scheme will give good result for only slow moving target. Localisation of fast motion has been addressed by predicting target's probable location in the current frame using prediction. Fused outputs of Extended Kalman Filter (EKF) and quadratic regression based prediction has been used to predict target's probable location in the current frame before applying fDSST for target localisation. It has helped in accurate localisation of fast motion without increasing search region. Thus, it reduces execution time also.

3.2.1 Extended Kalman Filter

Extended Kalman Filter (EKF) is the popular choice of prediction for manoeuvring target. This algorithm has small computational requirement and also is recursive so it can be used for real time processes. The Kalman filter algorithm involves two steps: Prediction and correction.

Tracking model includes target's dynamics. Typically objects do not move with constant velocity. A model of a target tracking problem where the target is constrained to move with a constant acceleration has been used. Let x and y represent the target tracking error (degree) in azimuth and elevation direction respectively. The system state can be described by $x_k = [x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}]^T$. It is assumed that sampled observations are acquired at discrete time interval Δt.

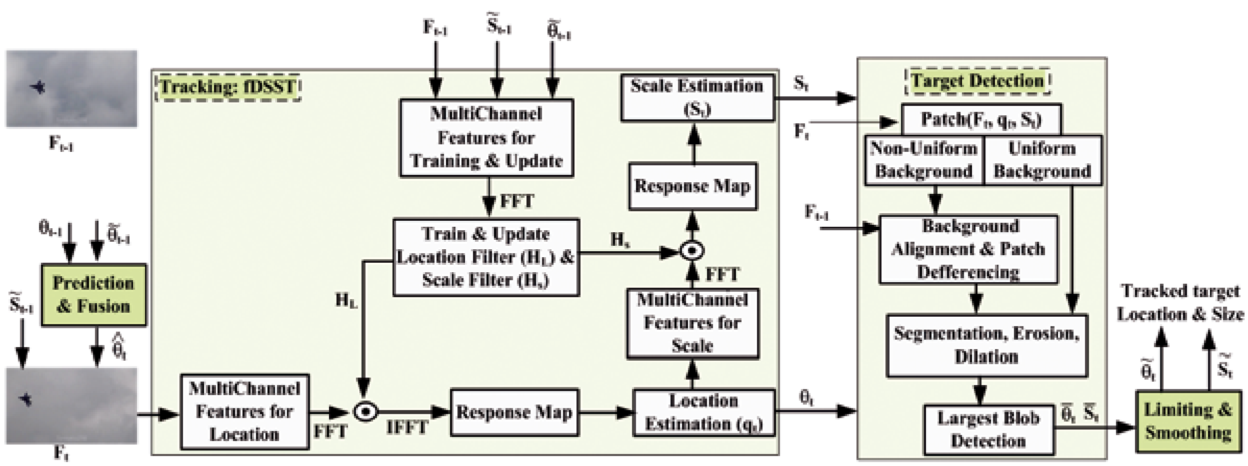

Fig. 1. Proposed tracking scheme.

The Kalman filter algorithm implements a discrete time, linear State-Space System described as Eqns (1) and (2). Equation (1) is state equation and Eqn (2) is measurement equation.

$$x_k = Ax_{k-1} + w_{k-1} \quad (1)$$

$$z_k = Hx_k + v_k \quad (2)$$

A is the state transition model and it describes state transition between time steps. Similarly H is the measurement model and it describes state to measurement transformation. Q is the process noise covariance and R is the measurement noise covariance. For constant acceleration tracking system A and H have been defined as follows.

$$A = \begin{bmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ 0 & 1 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (4)$$

The w_{k-1} and v_k are Gaussian white noise, whose variances are defined by

$$w_k \sim N(0,Q), v_k \sim N(0,R) \quad (5)$$

In the first frame only prediction has been applied. In subsequent frames, correction has been followed by prediction. So with current measurement, states have been corrected and future state has been predicted.

Kalman filter and its tuning has been discussed by Yaakov[13], *et al.* Q has been selected using Eqn (6).

$$Q = \begin{bmatrix} \frac{1}{4}T^4 & \frac{1}{2}T^3 & \frac{1}{2}T^2 \\ \frac{1}{2}T^3 & \frac{1}{2}T^2 & T \\ \frac{1}{2}T^2 & T & 1 \end{bmatrix} \sigma_v^2 \quad (6)$$

T is the sampling period and is of the order of the magnitude of maximum acceleration incremental over a sampling period (T), Δa_m. A practical range is $0.5\Delta a_m \le \sigma_v \le \Delta a_m$. Considering measurement error, step size of single pixel and its distribution as Gaussian, R has been taken as 1/12.

3.2.2 Quadratic Regression

Quadratic curve fitting has been also used to predict motion of target in each frame. Least squares approach for fitting a second order polynomial has been used. Last seven smoothen locations $(\tilde{\theta}_t)$ have been used in quadratic curve fitting and prediction.

Predicted location $(\hat{\theta}_t)$ to extract search patch z_t for target location is the combination of target location in the previous frame with fDSST algorithm (θ_{t-1}) and final location after smoothing $(\tilde{\theta}_{t-1})$, EKF predicted location (θ_{kp}) and quadratic regression based predicted location (θ_{qrp}) as given in Eqn (7):

$$\hat{\theta}_t = \frac{(\theta_{t-1} + \tilde{\theta}_{t-1} + 2.0\theta_{kp} + 1.5\theta_{qrp})}{5.5} \quad (7)$$

More weightage has been given to the EKF predicted location (θ_{kp}) and quadratic regression based predicted location (θ_{qrp}).

In general term, θ is related with angular position, and S is related to the scale of target. (θ, S) along with their subscript and superscript have been listed as follows

(θ_t):	Target location, estimated by fDSST
$(\bar{\theta}_t)$:	Target location, estimated after blob detection
$(\tilde{\theta}_t)$:	Target location, smoothened by limiting and smoothing
(θ_{kp}):	Target location, predicted by EKF
(θ_{qrp}):	Target location, predicted by quadratic regression
$(\hat{\theta}_t)$:	Predicted location using fused information from fDSST, limiting and smoothing, EKF and quadratic curve fitting
(S_t):	Scale of target, estimated by fDSST
$(\tilde{S}_t)$:	Scale of target, smoothened by limiting and smoothing
$(\bar{S}_t)$:	Scale of target, estimated after blob detection

3.3 Target Detection

Unlike other tracking schemes[14], where tracking and detection are parallel processes, here, accurate target detection has been done around fDSST located target. Different target detection schemes have been applied depending on the nature of background.

A permissible detection region factor (β) with respect to fDSST estimated target has been defined. The detection region size has been taken as Eqn (8):

$$Sd_t = \frac{(S_t + S_{t-1})}{2}(1+\beta) \quad (8)$$

S_t is the target size estimated using fDSST. Target region average intensity $(TgtAvgInt)$ has been calculated by cropping around centre of fDSST estimated target as per dimension given by Eqn (9):

$$Sdt_t = S_t\sqrt{\frac{1}{3}} \quad (9)$$

For background nature estimation, four side strips have been cropped from image based on Eqn (8) as per dimension given by Eqn (10). Only permissible boundary strips size of 1/3rd of pixels has been taken.

$$Sdb_t = \frac{(Sd_t - S_t)}{3} \quad (10)$$

Mean and standard deviation of all four strips has been calculated. Strip with highest standard deviation in intensity has been discarded. Absolute differences of remaining three strips means have been calculated and tabulated in a 3x3 matrix. Further elements of each row have been added. Strip causing highest sum has been eliminated. Average of mean of remaining two strips has been taken as average background intensity $(BckgndAvgInt)$. Similarly, average of standard deviation of background $(BckgndStdInt)$ has been calculated from these two strips. Threshold for segmentation has been calculated as per Eqn (11).

$$ThresholdSeg = \frac{(BckgndAvgInt + TgtAvgInt)}{2} \qquad (11)$$

If $(ThresholdSeg - BckgndAvgInt) \geq 2BckgndStdInt$ background is declared as uniform and vice versa.

3.3.1 Uniform Background

If background is almost uniform, current frame has been segmented with adaptive threshold level $(ThresholdSeg)$. Polarity of difference between threshold $(ThresholdSeg)$ and target average intensity $(TgtAvgInt)$ has been used adaptively to change label of segmented image. Erosion with disk structuring element with radius 2 pixels has been performed on segmented image to remove spurious targets. Again dilation with disk structuring element with radius 7 pixels has been performed to connect segmented target parts.

3.3.2 Non-Uniform Background

If background is not uniform, previous frame has been aligned with respect to the current frame using background motion estimation. For background motion estimation a patch of 3 times of size cropped image (Sd_t) has been taken from current and previous frame. Motion between background of frames has been estimated using FFT-based correlation. Previous frame has been aligned with current frame using estimated motion. Difference image of aligned frames has been normalised for intensity (0-255). Mean (Avg) and standard deviation (Std) of normalised image have been used to calculate adaptive threshold for segmentation. If intensity is greather than $(Avg + 3Std)$ or less than $(Avg - 3Std)$, it is part of the target. This segmented image has been filtered with median filter of size 5x5. Again dilation has been applied followed by erosion. In dilation structuring element is disk with radius 7 pixels and in erosion structuring element is disk with radius 6 pixels.

Largest blob has been declared as detected target. Target detection near estimated location has helped in acquiring accurate size and location.

3.4 Target Dynamics Limiting and Smoothing

Target location and size has been limited and smoothened to avoid sudden changes due to inaccuracy of target detection. Target location and size have been limited in accordance to detection area limit factor (β). Basic purpose of smoothing is to restrict rate of change of targets's location and size. Proposed scheme has been shown in Fig. 2. It is based on approach proposed in[15] which was further improved in[16].

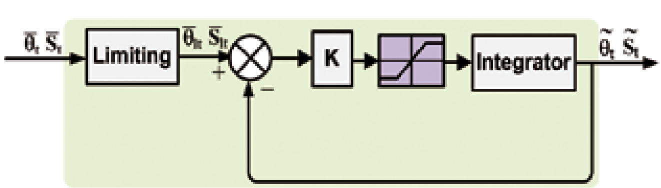

Figure 2. Limiting and smoothing scheme.

Smoothing scheme has a feedback structure. Considering required gain at low frequency, gain margin and phase margin for stability of the loop and 25 Hz frame rate, proportional controller gain (k) has been designed as 28. Generally, gain margin (GM) and phase margin (PM) are taken as greater than 6 dB and 30°, respectively. PM indicates transient behaviour. Damping ratio (ξ) and PM of second order system are approximately related as $\xi \approx \frac{PM}{100}$. ξ of the order of 0.55-0.7 gives good transient behaviour and fast response for step command. So, GM of 6dB and PM of around 60° has been target. Saturation block will limit derivative of output (i.e. rate of change of location and size). The limit of saturation block has been adaptively changed depending upon nature of input to the smoother. If inputs are increasing/decreasing, this limit has been increased/decreased. For location, rate of increment of limit is 15 and rate of decrement is 7.5. Similar for size, rate of increment of limit is 10 and rate of decrement is 3.33. Maximum and minimum limit of saturation block has been fixed. For translational motion and size, maximum limit are 150 and 100 pixels respectively. For translational motion minimum limit is 30 pixels and for size it is 10 pixels. These parameters have been tuned experimentally. This filtered parameters ($\bar{\theta}_{lt}$, $\bar{S}_{lt}$) is limited to permissible detection region (γ).

4. EXPERIMENTAL RESULTS

After development of proposed tracking scheme, its performance has been evaluated against standard datasets. Proposed scheme has been implemented in MATLAB. All experiments have been performed on an Intel(R) Xeon(R) CPU E5-2620 v3 @2.40GHz 2.40 GHz with 4 GB RAM computer. Implementation details have been presented in section 3.

For the tracking approaches presented in section 4.2, the same parameter settings are used for all experiments and videos. All algorithms have been quantitatively evaluated on the list of dataset given in section 4.1. Extensive experiments have been performed to test the proposed method and validate its robustness to the scale and orientation changes of the target.

4.1 Datasets

Large number (15) of challenging datasets having fast and manoeuvring target has been used to extensively evaluate performance of proposed framework. These types of datasets are not commonly available. So, first six datasets have been extracted from fighter planes videos available on www.youtube.com (i.e. 4 Days on the hills of Wales, Low flying in Mach Loop 480p.mp4, 2014 F-22 RAPTOR DEMO @ CALIFORNIA CAPITAL AIR SHOW 480p.mp4, Buzzed by F18's at Death Valley 480p.mp4, F-15C %20Grim Reapers%20, Low Level Mach-Loop 480p.mp4, Low-Flying Jets LFA 7 Training area Mach Loop 480p.mp4). In order to perform quantitative analysis, each video sequences has been manually annotated for the ground truth. Remaining nine datasets have been selected from[17-18]. These datasets with ground truth can be found at dropbox. Also, all datasets with tracking annotations of all considered state-of-art tracker and proposed framework can be found here (https://www.dropbox.com/sh/8m98ebmtxzjocj2/AAAJbbeZN3XQ_xzEtlvt8ABua?dl=0). Colour codes of trackers are as used in Fig. 4.

4.2 Tracking Performance

Tracking performance of proposed scheme has been evaluated and compared with other four state-of-the-art trackers. In scale adaptive with multiple features (SAMF) tracker two type of appearance features have been considered: Gray and colour HOG.

Before evaluating tracking, few suitable changes have been made to make them perform better for this type of scenario of fast movement of target. Since interframe motion is larger, padding for search region has been increased to 2.5 for all algorithms.

In DSST no other changes have been made except padding. In fDSST and proposed algorithms, along with change in padding, feature update rate (η) has been increased to 0.075 from 0.025. Also to cater for fast change in shape and size scale step has been changed to 1.075 from 1.02. In SAMF search size has been updated to [1 0.98 0.96 0.94 1.02 1.04 1.06] from [1 0.985 0.99 0.995 1.005 1.01 1.015]. Also, for better performance, In SAMF_Color HOG, interp_factor has been changed to 0.075 from 0.01, kernel. sigma to 0.2 from 0.5 to match with SAMF_gray.

For the comparative study of performance of proposed algorithms, four commonly used parameters[19] of tracking has been considered: Precision, overlap, centre location error (CLE) and speed of execution. Average distance precision, commonly known as precision is defined as the percentage of frames in a dataset where the Euclidean distance between the tracking output and ground truth centres is smaller than a threshold. Similarly average overlap is defined as the percentage of frames in a dataset where the intersection over-union overlap with the ground truth exceeds a certain threshold. This is normalised and it accounts for both position and size of the predicted and ground-truth bounding boxes simultaneously.

CLE in each frame of a dataset is defined as Euclidean distance in the centre of tracking output and ground truth. Commonly used threshold of 20 pixels for precision and 0.5 for overlap has been used in this paper. The value of average overlap for 50 per cent of overlap criteria is also called success ratio.

Average Precision and overlap of trackers against all of the 15 datasets have been tabulated in Tables 1 and 2, respectively. For precision and overlap, proposed framework has performed best (bold) in 10 and 12 datasets, respectively. Second best result has been underlined.

Average CLE has been consolidated in Table 3. Smaller the average CLE, better is tracking performance. In most cases, proposed method has performed better.

Along with precision and overlap, speed of algorithms was also compared. It may be noted that visualisation of tracked frame with annotation was enabled during execution rate estimation. From the average execution rate as tabulated in Table 4, it is clear that fDSST has performed faster in the most of the case. But in a few cases, proposed scheme has edged it.

Table 1. Tracking performance: Precision (per cent)

	Proposed	fDSST	DSST	SAMF_ Gray	SAMF_ HOG
Fighterplane_1	**94.7**	58.6	55.9	<u>60.9</u>	32.8
Fighterplane_2	**80.4**	51.7	26.9	69.2	<u>69.5</u>
Fighterplane_3	**98.4**	<u>91.3</u>	81.2	51	78.7
Fighterplane_4	98.9	**100**	**100**	<u>99.3</u>	**100**
Fighterplane_5	87	**97.5**	81.5	86	<u>89.5</u>
Fighterplane_6	<u>93.5</u>	**100**	69	86	51
Airplane_001	**98**	<u>43</u>	42.5	16.5	17
Airplane_004	**100**	21.5	21.5	21	<u>27</u>
Airplane_006	**95**	55.5	52.5	37.5	<u>75.5</u>
Airplane_007	**87**	32	22	<u>38</u>	<u>38</u>
Airplane_011	**100**	<u>77.7</u>	59.3	51.3	30
Big_2	74.5	**100**	**100**	**100**	<u>97</u>
Planestv_2	**81.5**	13.5	25.5	<u>37</u>	15
Planestv_3	**98.0**	<u>17.7</u>	9	2.67	10.7
Planestv_7	54.1	65.3	52.9	53.5	**60.6**

Table 2. Tracking performance: Success ratio (per cent)

	Proposed	fDSST	DSST	SAMF_ Gray	SAMF_ HOG
Fighterplane_1	**72.8**	<u>11.8</u>	9.47	11.5	11.2
Fighterplane_2	**45.3**	<u>13.3</u>	12.7	3.63	12.4
Fighterplane_3	**73.3**	<u>35.4</u>	16.3	3	16.1
Fighterplane_4	**92.7**	57.1	<u>60.7</u>	46.5	50.2
Fighterplane_5	<u>69</u>	**82**	31.5	6.5	11.5
Fighterplane_6	**83.5**	<u>29.5</u>	3	6.5	<u>29.5</u>
Airplane_001	**100**	16	<u>35</u>	15.5	16
Airplane_004	**95**	<u>54.5</u>	25	21.5	36
Airplane_006	**88**	<u>47.5</u>	43.5	27.5	55
Airplane_007	**69**	<u>50</u>	47.5	39.5	30.5
Airplane_011	**50**	<u>31.3</u>	29.3	11.3	30
Big_2	71	**100**	<u>96</u>	95.5	83
Planestv_2	<u>45.5</u>	**64.5**	**64.5**	21	40.5
Planestv_3	**86.3**	<u>37.7</u>	15.7	4.67	15.3
Planestv_7	**65.9**	<u>54.1</u>	48.2	22.9	23.5

In last nine datasets (1280 x 720 pixels) all algorithms have been slow due to large frame size as compared to first six datasets (640 x 360 pixels). Due to large frame format, more data were required to be processed.

Table 5 shows the average performance considering all datasets. Proposed scheme is distinctly better than rest in first three parameters (precision, success ration and CLE). In case of speed, fDSST has performed better.

Average precision and success rate of all datasets for different limits have been shown in Fig. 3. Precision curves simply show the percentage of correctly tracked frames for a range of distance thresholds. A higher precision at low thresholds means the tracker is more accurate. Similarly a higher overlap at low threshold indicates better performance.

Table 3. Tracking performance: Average CLE (pixels)

	Proposed	fDSST	DSST	SAMF_ Gray	SAMF_ HOG
Fighterplane_1	**7.48**	19.3	37.2	19	30.1
Fighterplane_2	**11.9**	23.5	27.9	18.3	15.8
Fighterplane_3	**5.04**	9.72	13.7	19.8	14.3
Fighterplane_4	**3.11**	7.43	5.69	8.45	7.59
Fighterplane_5	8.32	**5.44**	9.73	13.4	7.51
Fighterplane_6	**5.81**	8.02	14.4	13.4	25
Airplane_001	**5.15**	32.2	34.1	60.5	49.3
Airplane_004	**5.54**	28	28.2	32.4	26.2
Airplane_006	**5.75**	19.9	19.2	24.7	17.7
Airplane_007	**8.32**	51.3	22	41.1	41.3
Airplane_011	**4.43**	12.4	18.9	26.3	82.7
Big_2	11.6	4.02	**3.66**	5.23	6.05
Planestv_2	**13.8**	35.3	31	49.9	38.8
Planestv_3	**7.25**	28.7	52.4	60.9	39.8
Planestv_7	25.2	20.7	32.6	45.6	**17.1**

Table 4. Tracking performance: Execution rate (FPS)

	Proposed	fDSST	DSST	SAMF_ Gray	SAMF_ HOG
Fighterplane_1	58.1	**157**	90	17.2	15.8
Fighterplane_2	23.5	**122**	61.6	16	14.6
Fighterplane_3	32.5	**49.7**	35.4	12.4	13.9
Fighterplane_4	21.6	**42**	2.82	10.8	11
Fighterplane_5	52.1	**145**	104	18.4	16.7
Fighterplane_6	21.5	**126**	98.3	18.4	16.3
Airplane_001	**28.6**	14.4	10.1	2.92	3.37
Airplane_004	8.85	**55.6**	1.82	3.64	4.28
Airplane_006	**17.2**	10.8	2.7	6.86	6.19
Airplane_007	8.55	**10.3**	2	3.9	5.77
Airplane_011	**19**	11.2	3.31	1.51	2.14
Big_2	11.8	**25.5**	1.44	2.93	4.85
Planestv_2	**9.16**	6.42	1.15	2.77	3.6
Planestv_3	5.77	**8.48**	1.25	2.49	3.85
Planestv_7	6.17	**9.1**	2.22	3.71	5.8

Table 5. Average tracking performance

	Proposed	fDSST	DSST	SAMF_ Gray	SAMF_ HOG
Precision (%)	**91.28**	62.54	53.65	54.87	52.97
SR (%)	**74.16**	42.35	32.87	20.24	28.68
CLE (Pixel)	**8.10**	19.72	23.85	28.28	28.46
Execution (fps)	22.31	**54.49**	28.16	8.43	8.78

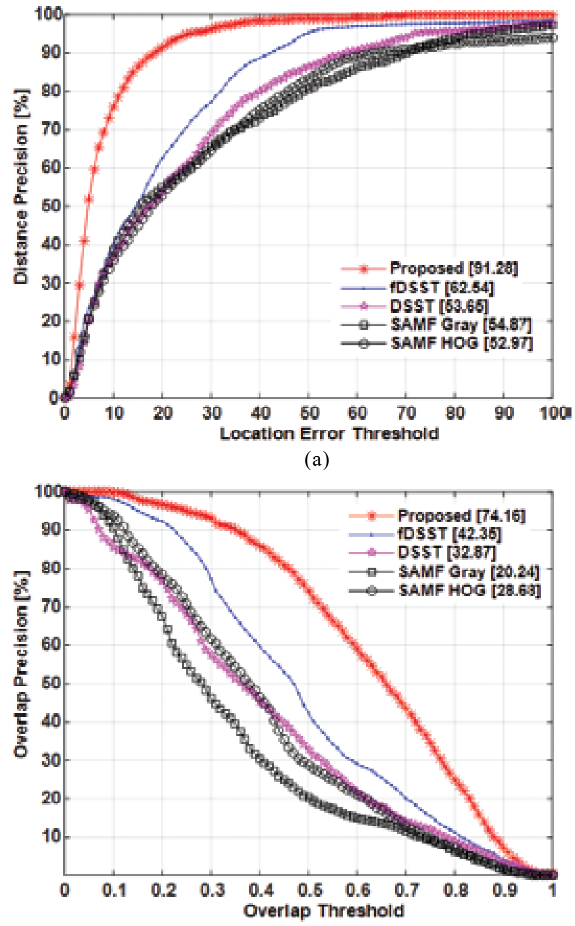

Figure 3. Comparative Performance: (a) Precision plot (b) Success plot.

Both plots indicate that the proposed method has performed better than the rest with a decent margin.

Also, all trackers were qualitatively evaluated for complete datasets. Representative performances of all five trackers against few datasets have been listed in Fig. 4. It can be observed that the proposed scheme is the most accurate and adaptive to target shape and size changes.

Proposed algorithm has performed well with uniform background (Figs. 4(a) and (d)) as well as non-uniform background (Figs. 4(b) and (c)). Its major merit is quick adaptation to the shape and size changes. Smoothing of location and size of detected target has improved tracking performance. The major contributions in this work are addition of detection with fDSST tracking algorithm. Target location prediction and smoothing of detected target dynamics have also been included.

5. CONCLUSIONS

Accurate scale estimation and target detection based scheme for fast moving target tracking has been presented. A framework has been developed, where detection has improved location and scale estimated by tracking algorithm. Target location prediction helped in a significant increase in tracking performance for fast moving target without increasing search region. Experimental results show that the proposed scheme

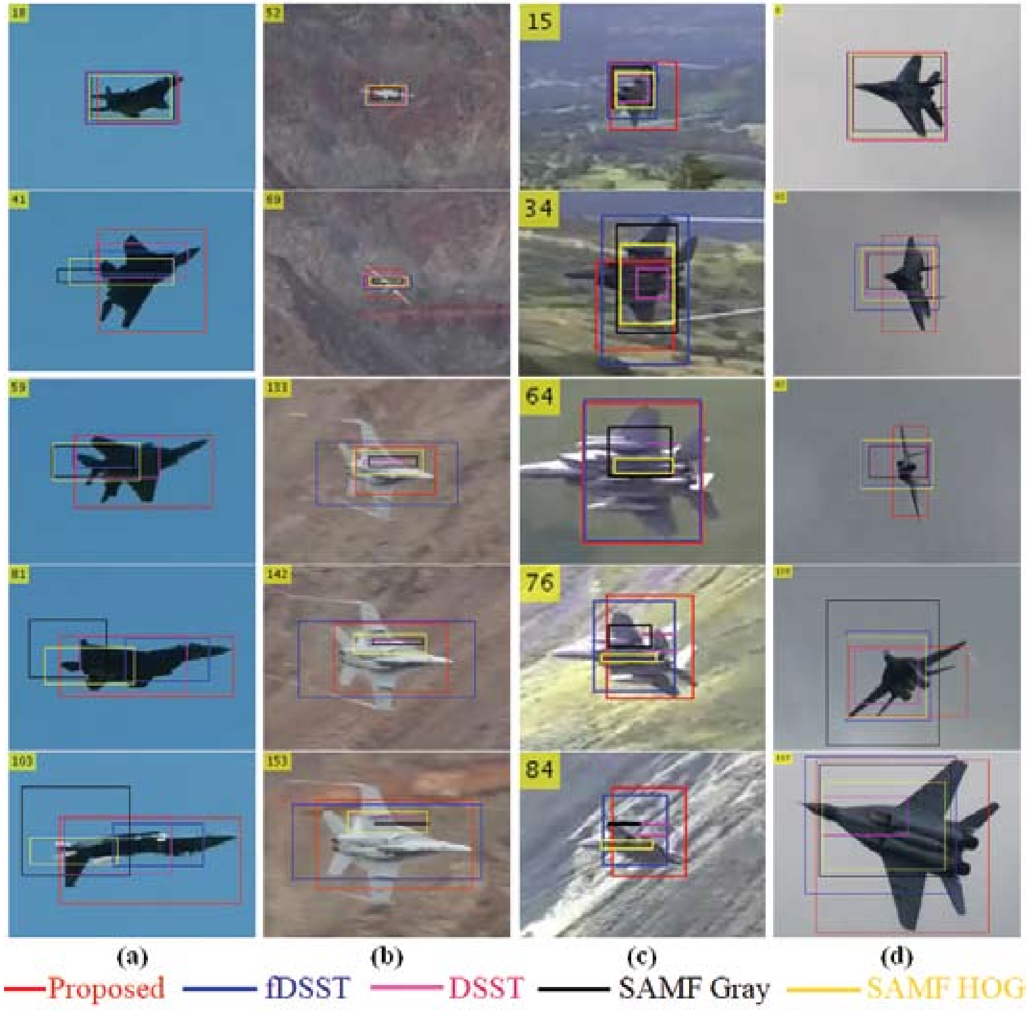

Figure 4. Tracking in representative frames (show bounding box of all 5 trackers with frame number): (a) Airplane_001 (b) Fighterplane_3 (c) Fighterplane_5 and (d) Airplane_004 (best view in Colour).

facilitates tracking of fast and manoeuvring target with high precision and success ratio. It has performed distinctly well against manoeuvring and scale change compared to standard (benchmark) tracking algorithms. Experimental results confirm the efficacy of the proposed algorithm.

Novelty of this work resides in the addressing of a practical problem of accurately tracking of fast moving and scaling target. It has been achieved by using fDSST, trajectory prediction using EKF, quadratic regression, target detection and trajectory smoothing.

REFERENCES

1. Maggio, Emilio & Cavallaro, Andrea. Video tracking: Theory and practice. 1st ed. A John Wiley and Sons, United Kingdom, 2011, pp. 1-2.
2. Arnold, W.M. Smeulders; Dung, M. Chu ; Rita, Cucchiara; Simone, Calderara ; Afshin, Dehghan & Mubarak, Shah. Visual tracking: An experimental survey. *IEEE Trans. Pattern Anal. Mach. Intell.,* 2014, **36**(7), 1442-1468. doi: 10.1109/TPAMI.2013.230.
3. Yilmaz, A.; Javed, O. & Shah, M. Object tracking: A survey. *ACM Computing Surveys* (CSUR), 2006, **38**(4), 1-45. doi: 10.1145/1177352.1177355.
4. Danelljan, M.; Khan, F. Shahbaz; Felsberg, M. & Weijer, J. van de. Adaptive color attributes for real-time visual tracking. *In* Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 2014, pp. 1090-1097. doi:10.1109/CVPR.2014.143.
5. Bolme, D. S.; Beveridge, J.R.; Draper, B.A. & Lui, Y.M. Visual object tracking using adaptive correlation filters. *In* Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 2010, pp. 2544-2550, doi: 10.1109/CVPR.2010.5539960.
6. Henriques, J.; Caseiro, R.; Martins, P. & Batista, J. High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015, **37**(3), 583-596. doi:10.1109/TPAMI.2014.2345390.
7. Babenko, B.; Yang, M.-H. & Belongie, S. Visual tracking with online multiple instance learning. *In* IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, 2009, pp. 983-990, doi: 10.1109/CVPR.2009.5206737.
8. Ager, G. H¨.; Khan, F.S. & Danelljan, M.F.M. Accurate

scale estimation for robust visual tracking. *In* European Conference on Computer Vision, 2014. pp. 1-14.
doi: 10.5244/C.28.65.

9. J. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in European Conference on Computer Vision (ECCV), Lecture Notes in Computer Science, vol 7575. Springer, Berlin, Heidelberg, 2012, pp. 702-715, **doi:** https://doi.org/10.1007/978-3-642-33765-9_50.
10. Galoogahi, H.; Sim, T. & Lucey, S. Multi-channel correlation filter. *In* IEEE International Conference on Computer Vision (ICCV), Sydney, NSW, 2013, pp. 3072-3079.
doi: 10.1109/ICCV.2013.381.
11. Li, Y. & Zhu, J. A scale adaptive kernel correlation filter trackerwith feature integration. *In* ECCV Workshop on the Visual Object Tracking Challenge, Zurich, Switzerland, 2014, pp. 254-265,
doi:10.1007/978-3-319-16181-5_18.
12. Danelljan, Martin; Häger, Khan, Gustav Fahad Shahbaz & Felsberg, Michael. Discriminative Scale Space Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.,* 2017, pp. 1561-1575,
doi: 10.1109/TPAMI.2016.2609928.
13. Bar-Shalom, Yaakov; Li, X. Rong & Kirubarajan, Thiagalingam. Estimation with applications to tracking and navigation: Theory algorithms and software.: John Wiley & Sons, Inc., 2001, ISBN: 0-471-22127-9.
14. Mikolajczyk, Krystian & Kalal, Jiri Matas Zdenek. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.,* 2012, **34**(7), 1409-1422.
doi: 10.1109/TPAMI.2011.239.
15. Ansari, Zahir Ahmed; Nigam, Madhav Ji & Kumar, Avnish. Fast steering and quick positioning of large field-of-regard, two-axis, four-gimbaled sight. *Optical Engineering,* 2017, B(7), 073101:1-12.
doi: 10.1117/1.OE.56.7.073101.
16. Ansari, Zahir Ahmed; Nigam, Madhav Ji & Kumar, Avnish. Improved scheme for quick positioning of dual gimbal sight. *In* Third International Conference on Advances in Control & Optimization of Dynamical Systems (ACODS-2018), Hyderabad, 2018,
doi: 10.1016/j.ifacol.2018.05.115.
17. Bharati, S.P.; Wu, Y.; Sui, Y.; Padgett, C. & Wang, G. Real time obstacle detection and tracking for sense-and-avoid mechanism in UAVs. *IEEE Trans. Intell. Vehicles*, 2018, **3**(2), 185 - 197
doi: 10.1109/TIV.2018.2804166.
18. Bharati, S.P.; Nandi, S.; Wu, Y.; Sui, Y. & Wang, G. Fast and robust object tracking with adaptive detection. *In* IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), San Jose, California, 2016, pp. 706-713,
doi: 10.1109/ICTAI.2016.0112.
19. Wu, Y.; Lim, J. & Yang, M.-H. Online object tracking: A benchmark. *In* IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, 2013, pp. 2411-2418.
doi: 10.1109/CVPR.2013.312.

CONTRIBUTORS

Dr Zahir Ahmed Ansari has obtained BTech from AMU Aligarh and MTech from IIT Kharagpur in visual information processing and embedded systems. His areas of interest are line-of-sight stabilisation and Visual Tracking. He has completed his PhD from IIT Roorkee. He joined IRDE, Dehradun in 2003 and involved in development of gimballed seeker, LOS stabilised sights and electro-optical tracking systems.
In the current study he has done problem formulation, literature survey, algorithm development and experimental evaluation.

Dr Madhav Ji Nigam received his BTech degree in electronics and communication engineering from Regional Engineering College, Warangal, in 1976, his ME degree in control and guidance, in 1978 and his PhD in electronics and communication engineering in 1992 from the University of Roorkee (Now IIT-Roorkee). Presently, he is working as professor in IIT, Roorkee. His area of research includes guidance, control and tracking for peace time work and industry.
In the current study he guided and supervised this work. He has major contribution in problem formulation and shaping this research paper.

Mr Avnish Kumar received the BTech (Electrical) and MTech (Electronics) from IIT Roorkee. Currently he is working in IRDE, Dehradun and is guiding a team of scientists for the design and development of line-of-sight stabilisation, electro-optical surveillance and fire control systems for various platforms.
In the current study he supervised this work. His vast experience in this field has helped in the analysis of the results.