AFAM-PEC: Adaptive Failure Avoidance Tracking Mechanism Using Prediction-Estimation Collaboration

BABER KHAN\textsuperscript{1}, AHMAD ALI\textsuperscript{2}, ABDUL JALIL\textsuperscript{1}, KHIZER MEHMOOD\textsuperscript{1}, MARIA MURAD\textsuperscript{1}, AND HAMDAN AWAN\textsuperscript{3}, (Member, IEEE)

\textsuperscript{1}Department of Electrical Engineering, International Islamic University Islamabad, Islamabad 44000, Pakistan
\textsuperscript{2}Department of Software Engineering, Bahria University, Islamabad 44000, Pakistan
\textsuperscript{3}Telecommunication Systems and Software Research Group (TSSG), Waterford Institute of Technology (WIT), Waterford, X91 WR86 Ireland

Corresponding author: Baber Khan (baber.khan@iiu.edu.pk)

\textbf{ABSTRACT} During recent years correlation tracking is considered fast and effective by the virtue of circulant structure of the sampling data for learning phase of filter and Fourier domain calculation of correlation. During the occurrence of occlusion, motion blur and out of view movement of target, most of the correlation filter based trackers start to learn using erroneous samples and tracker starts drifting. Currently, adaptive correlation filter based tracking algorithms are being combined with redetection modules. This hybridization helps in redetection of the target in long term tracking. The redetection modules are mostly classifier, which classify the true object after tracking failure occurrence. The methods perform favorable during short term occlusion or partial occlusion. To further increase the tracking efficiency specifically during long term occlusion, while maintaining real time processing speed, this study proposes tracking failure avoidance method. We first propose, a strategy to detect the occlusion using two cues from the response map i.e., peak correlation score and peak to side lobe ratio. After successful detection of tracking failure, second strategy is proposed to save the target being getting more erroneous. Kalman filter based predictor continuously predicts the location during occlusion. Kalman filter passes this result to Support Vector Machine (SVM). When the target reappears in frame, support vector machine based classifier classifies the correct object using the predicted location of Kalman filter. This decreases the chance of tracking failure as Kalman filter continuously updates itself during occlusion and predicts the next location using its own previous prediction. Once the true object is detected by classifier after the clearance of occlusion, this result is forwarded to correlation filter tracker to resume its operation of tracking and updating its parameters. Together these two proposed schemes show significant improvement in tracking efficiency. Furthermore, this collaboration in redetection phase shows significant improvement in the tracking accuracy over videos containing six challenging aspects of visual object tracking as mentioned in the literature.

\textbf{INDEX TERMS} Adaptive correlation filter, Kalman filter, occlusion, support vector machine classifier.

\textbf{I. INTRODUCTION} Visual object tracking has always been considered as an active area of interest in the research field of computer vision because of its side spread applications and challenging issues like motion blur, object deformation, noisy environment, fast motion, clutter and finally occlusion [1], [2]. Long term tracking is considered effective if an algorithm tracks an object of interest for long duration of time in all or any of the above challenging scenarios. Without considering orientation estimation of the object, tracking process can be divided into two sub parts i.e. i) translation estimation and ii) scale estimation of target in next frame [3].

For translation estimation, broadly tracking algorithms can be divided into two groups; i) generative and ii) discriminative. In the generative scheme, the information of the object is used while considering tracking as search problem. The discriminating scheme considers the tracking as
classification problem, while using the object and its background information. Discriminative tracking using correlation filter is studied by a number of researchers in the field of object tracking [4]–[11]. Exploiting circular structure and computing correlation in frequency domain which is simply multiplication extreme fast tracker is presented in [12]. Due to adaptive nature of correlation filter, online fast learning mechanism makes correlation filter suitable for fast appearance changing object tracking. Though correlation filters are very successful in visual object tracking but still two major limitations exist: First they do not have the inherent capability of tracking resumption once the object is lost or the object moves out of the camera’s field of view. Second is that the less reliable tracked frame causes the correlation filter to learn wrong target appearance and this learning error accumulates with the passage of frames. The first limitation is addressed in [9], [13] by considering the redetection module, where redetection is carried out in each frame, which in turn increases the computational cost. Another approach to reduce the computational cost is by defining a threshold to activate the redetection module is presented in [14]. The second limitation of correlation filter based tracking is solved in [3] by learning multiple correlation filters having different learning rates. To cover the fixed template size problem of kernel filters, correlation filter adaptive to scale changing is presented in [15]. Dense spatio-temporal context information is used in [16] to increase the efficiency and robustness of the correlation filters. Simple tracking approach with an appearance model based on multi-scale image feature extraction using data-independent basis is presented in [17]. Particle filters is also incorporated in kernelized correlation filters to redetect the tracker when response map becomes less reliable [18]. Fusion of multiple features in correlation filter framework is proposed in [19]. In this method adaptive weights are assigned to each feature to minimize the interference of noise. To enhance the quality of response map in correlation filter based algorithms, metric learning model strategy is given in [20]. Convolutional Neural network based tracking strategies are also proposed by numerous researcher during recent years, for latest examples, see [21], [22]. These neural network based algorithms require a lot of training data and large computational time.

In this paper we proposed a new scheme which incorporates the Kalman filter and support vector machine (SVM) into discriminative correlation tracking. Moreover, MULTI-CUE based reliability detection scheme i.e. frames which are most reliable to update the target model is presented.

The main contributions of the scheme proposed in this paper are:
1) The Kalman filter based prediction under heavy occlusion shows better tracking efficiency compared with [14], in case of linear motion (because linear Kalman filter is used in this paper).
2) It is shown that peak correlation score alone is not good enough to detect heavy occlusion, motion blur, scale variation, background clutter, out of plane rotation and deformation. We provide comparison with the pervious works [14], [16], [17]. Hence, the peak to side lobe ratio is incorporated with peak correlation score in proposed strategy.

3) The presented algorithm in this paper is able to avoid the template update under erroneous input (wrongly tracked object) and gives signiﬁcant improvement in tracking accuracy.

In the end we compare the results of the proposed algorithm with the state-of-the-art trackers on selected challenging videos having six attributes, from benchmark data sets OTB50 [23], OTB100 [24], TC128 [25], and UAV-123 [26].

The remainder of the paper is organized as follows: Section II presents the related work and background. Section III presents the proposed tracking scheme. Section IV presents implementation details. Section V presents analysis and evaluation of the proposed tracking scheme. Finally, Section VI presents conclusion.

II. RELATED WORK AND PROBLEM BACKGROUND

In recent years there has been an increase in interest of researchers in computer vision in the field of Visual object Tracking (VoT) because of the availability of high-speed computational resources along with availability of benchmark datasets and results. This section will give insight of most closely related tracking schemes to the proposed method: i) correlation filter based tracking ii) tracking learning and detection also known as tracking by detection. Further detailed discussions are available in [2], [27]–[30].

A. TRACKING LEARNING AND DETECTION (TRACKING BY DETECTION)

This method considers the object tracking as detection problem in every frame. To make the correlation filter adaptive to the appearance changes of the target of interest, recently proposed methods draw positive and negative sample around the expected target to update the classifier discussed in [3]. However, slightly erroneous labeling of samples accumulate over time and tracker start drifting. This problem is known as sampling ambiguity. To handle it many methods have been proposed such as ensemble tracking [30], randomized ensemble tracking [31], adaptive randomized ensemble tracking [32], online multiple instance learning [33], and transfer learning based tracking [34], [35]. Another problem, with the approach explained at the start of this paragraph, is tradeoff between stability and adaptivity. To keep the system stable along with reasonable model adaptivity, tracking scheme has been decomposed into three modules i.e. training learning and detection given in [36], [37]. Basic idea in the subsequent method is to update the detector with conservative rate using the extra sample obtained from the results of aggressively updated tracker. This online detector can be used in case of occurrence of tracking failure. Examples of
such tracker are given in [9], [13], [14]. Online detector for reinitialization of tracker in case of tracking failure is also proposed in [3]. The detection module is activated only if the response is lower than specified threshold. In our proposed tracking scheme, we also use, support vector machine based online trained detector module which is different from the already proposed [14], [14] techniques. We activate the support vector machine based detector module on the basis of two parameters rather than only peak correlation value. In our approach, Adaptive Failure Avoidance Tracking Mechanism using Prediction-Estimation Collaboration (AFAM-PEC) response map is utilized to calculate the peak to side lobe ratio along with peak correlation value.

**B. CORRELATION TRACKING**

Correlation filters are applied in many application areas like object detection and recognition [38]. This operator simply works as element to element multiplication in frequency domain, researchers have applied correlation filter extensively to visual object tracking in the last decade due to its less computational cost attribute. Minimum output sum of squared error (MOSSE) filter is proposed in [4] for tracking on monotonic images, where the filter is updated on every frame. This filter is computationally inexpensive having processing speed of more than hundred frames per second. Kernelized correlation filter is proposed in [12], [39], which employs the properties of circulant matrices for extreme fast learning and detection with the help of fast Fourier transform. Efforts have been made to enhance the tracking performance by using correlation filters. Examples of algorithms based on correlation filter includes multi-channel filters [39]–[41], spatio-temporal context learning [16], scale handling and estimation [42]–[44] and spatial regularization [45]–[47]. Most of these techniques are very good in adopting the fast changing appearance of the model, but due to non-availability of long term memory of target appearance, these techniques are susceptible to drift in case of occlusion and out of the view movement of target object. This problem is solved by keeping the long-term memory of the target and deploying two filters, one for short term memory and other for long term memory [3]. At the same time this increases the computational cost and more memory will be consumed. Compressive tracking algorithms are also presented in recent years which extracts the features from multi scale feature space whose basis function do not depend upon the data. One of the examples of these types of trackers is given in [17].

Unlike existing techniques that employ only correlation filter for translation estimation even during occlusion, we introduced the predictor module, to handle the drifting/tracking failure in case of occlusion, motion blur and out of the view movement of the target. In our approach (AFAM-PEC) predictor is incorporated with the short term memory correlation filter. Peak to side lobe ratio and peak correlation score from response map is calculated to predict the occlusion/motion blur/out of view movement of an object. Based on these two parameters confidence score is calculated, which will decide the reliability of the tracking result for that specific frame. Short term memory filter will stop updating its weights if tracking reliability is less then certain threshold say $t_r$. Measurement follower predictor is used to predict the next position of the object during occlusion time. Once the tracking result reliability approaches specified threshold say $t_r$, again short term memory filter is activated to estimate the next state of the object.

---

**Algorithm 1 Proposed Tracking Scheme**

Input: Box at $x_0$ Containing Target, Estimated position and scale of an object $x_t = (\hat{x}_t, \hat{y}_t, \hat{s}_t)$, Context Regression model $R_{con}$, target appearance regression model $R_{tar}$, support vector machine based classifier $D_{sym}$ and measurement follower predictor $P_{mf}$.

Repeat

1. Extract the search window in current frame $t$ at $(\hat{x}_{t-1}, \hat{y}_{t-1})$ and draw out the features;
   \text{//Translation estimation}
2. Compute the correlation response map $y_t$ using $R_{con}$ and equation no (4) and estimate the new position $\hat{y}_t$;
   \text{//Scale estimation}
3. Build the target pyramid around $(\hat{x}, \hat{y}_t)$ and compute the correlation map $y_s$ using $R_{tar}$ and (4);
4. Extract the parameters from response map $y_s$ Estimate the optimal scale $s$ using (7); $X_t = (x_t, y_t, \hat{s})$;
   \text{// Target prediction and redetection}

Do

- Use $P_{mf}$ to predict the next state of object, $x'_t = x_t(P_{mf})$ using $(\hat{x}_t, \hat{y}_t)$;
- Use detector $D_{ef}$ to find the all possible states $x_i$ in $X(D_{ef})$;
- Compute $y'_i$, for each state in $X(D_{ef})$;
- Stop updating $R_t$ and $j_t$;
- if $\max(y'_i) > T_I$ then
  \begin{align*}
  x_t &= x'_i;
  
  \end{align*}
  \text{where } i &= \argmax, X_i y'_i;
  \text{input to kalman } = x_i \text{ in (13)};
  \text{update the parameters of the detector using (9)};
  \text{else}
  \begin{align*}
  x_t &= x'_t;
  \text{input to kalman } = x_t
  \end{align*}
  \text{while }\max(y'_i) < T_r \text{ or } PSR(y'_i) < T_{psr} \text{ end}

Update $R_{con}$ using (5,6);
if $\max(y'_i) < T_o$ and $PSR(y'_i) > T_{psr}$ then
  Update $R_t$ using $j_t$ and (5,6);
end

Until end of video sequence;
TABLE 1. Notations/variable explanation.

| Denotation                        | Symbol | Note                                                               |
|----------------------------------|--------|--------------------------------------------------------------------|
| Estimated position and scale     | $x_r$, $y_r$, $s_i$ | position of the object and $s_i$ is estimated scale             |
| Correlation response map         | $y_i$  | At $t^\theta$ frame                                               |
| Regression model                 | $R_{con}$ | With respect to context                                           |
| Regression model                 | $R_{tar}$ | Model of target                                                  |
| Detector module                  | $D_{con}$ | SVM based                                                          |
| Predictor module                 | $P_{\psi}$ | Kalman based                                                      |
| Estimated new position           | $\hat{y}_i$ | At $t^\theta$ frame                                               |
| Predicted state                  | $x_r^T$  | State by $P_{\psi}$ at $t^\theta$ frame                          |
| Estimated states                 | $X(D_{con})$ | All possible states by detector                                    |
| Estimated state                  | $x_i^T$  | Estimated possible state 1 for $t^\theta$ frame                   |
| Response map value               | $y'_i$  | Response map value for estimated state $i$                        |

where $\sigma$ is mapping to kernel space and $\lambda$ denotes regularization parameter, which is always greater then or equal to zero. As labeling is not binary, hence $w$ contains the coefficients of gaussian ridge regression model [48]. By using fast Fourier transform the above objective function is minimized to the (2).

$$w = \sum_{p,q} c (p, q) \sigma (i_{p,q}),$$  \hspace{1cm} (2)

where $c$ is calculated by (3) using discrete Fourier transform as follows:

$$C = f (c) = \frac{f (y)}{f (\sigma (i) \cdot \sigma (\hat{i})) + \lambda},$$  \hspace{1cm} (3)

$f$ denotes the discrete Fourier transform (DFT) and $y = \{y(p, q) | (p, q) \in [0, 1, \ldots, P-1] \times [0, 1, \ldots, Q-1]\}$.

In the new frame, response map over image patch $u$ of size $P \times Q$ is calculated by using inverse discrete Fourier transform as per (4).

$$\hat{y} = f^{-1} (C \odot f (\sigma (u) \cdot \sigma (\hat{i}))),$$  \hspace{1cm} (4)

where $\odot$ is element-wise multiplication, $\hat{x}$ is learned target appearance model and maximum value of $\hat{y}$ is the new target location. Two correlation filters are trained using single frame, one to model the target appearance solely and other to model the surrounding along with the target. As surrounding information does not change quickly and remains temporally stable, it is very useful to differentiate the target from the background in case of occlusion [3], [14], [16]. Weighted cosine window is applied to feature channels to remove the boundary discontinuities of the response map.

Context Regression model $R_{con}$ is adaptive to cater the occlusion, abrupt motion and deformation with the learning rate as:

$$\hat{\beta}^T = (1 - \beta^T) \hat{\beta}^{T-1} + \beta^T \hat{\beta},$$  \hspace{1cm} (5)

$$\hat{A}^T = (1 - \beta^T) \hat{A}^{T-1} + \beta^T \hat{A},$$  \hspace{1cm} (6)

Target appearance regression model $R_{tar}$ is learned from the most reliable and confidently tracked frames. Reliability is determined using maximal value of $\hat{y}$ [14]. Unlike the existing techniques [3], [14] to maintain the model stability in true letter and spirit, two thresholds are defined to update the target regression model $R_{tar}$ using (4). First threshold $T_a$ is on peak correlation value. Second threshold $T_{psr}$ is on peak to side lobe ration of the response map. If both the criterion are met, then only target appearance regression model is updated using (4) i.e. $\max (\hat{y}) > T_a \& PSR(\hat{y}) > T_{psr}$. Not that only the peak correlation value is enough to ensure the model stability in case of long term occlusion, which can be seen in Fig. 11.

In case of long term occlusion, which can be seen in Fig. 11.

$$\hat{x} = \arg \max (\hat{y}_1), (\max (\hat{y}_2), \ldots, (\max (\hat{y}_N)),$$  \hspace{1cm} (7)

III. THE PROPOSED TRACKING SCHEME

Our objective is to develop robust online training based visual object tracking algorithm, to handle the long term occlusion more effectively in comparison to other already proposed long term tracking methods. Without considering the orientation of the object, tracking is simply the estimation of the translation and scale of an object [3], [14]. In our proposed framework, the translation estimation is based on the correlation of temporal context and scale estimation is based on the discriminative correlation filter.

In this section, the main components of the proposed tracking procedure are described. First of all long term correlation tracking [3], [14] is described in IIIA. Next we describe the estimator module [14] in section IIIB, we use support vector machine classifier. Section IIIC describes the Kalman filter based predictor. Finally, in section IID, predictor strategy is described, which assist the translation estimation filter during the long term occlusion. Different notations and variables used in the following sections are given in the Table 1.

A. LONG TERM CORRELATION TRACKING

Correlation filter based tracker calculate the weights $w$ by training on an image patch $i$ of $P \times Q$ pixels to model the target appearance [3], [14], [34], [35], [39], [42], [44], where all the circulant shifts $i_{p,q}$, $(p, q) \in [0, 1, \ldots, P-1] \times [0, 1, \ldots, Q-1]$, are considered as samples for training with the gaussian function label $y(p, q)$, i.e.

$$w = \arg \min_w \sum_{p,q} |\sigma (x_{p,q}) \cdot w - y(p,q)|^2 + \lambda (w)^2,$$  \hspace{1cm} (1)
where each \( s \in S \),
\[
S = \{ a_n | \left[ -\frac{N-1}{2}, -\frac{N-3}{2}, \ldots, -\frac{N-1}{2} \right] \}.
\]

Unlike [34], we make the updating of target regression model more robust and \( R_{psr} \) is updated using (4) if it satisfies the condition \( \max(y_i) > T_a \) \& \( \text{PSR}(y_i) > T_{psr} \).

### B. SUPPORT VECTOR MACHINE BASED ESTIMATOR

To increase the robustness of tracking algorithm, detection module is necessary to recover the target when tracking failure occurs due to long term occlusion and reentering into the camera view, erroneous input to model update module or out of camera view movement of an object.

Researchers proposed this model of online detector and carried out re-detection on each frame in [13], [49], [50]. To decrease the computational efficiency, certain threshold is defined to activate the detector. Detector is activated only if the maximum value of response map is less than certain predefined threshold [3]. Unlike to these two approaches, support vector based detector in collaboration with Kalman filter based predictor is implemented in our approach i.e. proposed detector is activated if either of the two conditions are true i.e. i) \( \max(y_i) < T_r \) \& \( \text{PSR}(y_i) < T_{psr} \).

SVM is trained incrementally by considering thick training samples around the estimated position. Binary labels have been assigned with respect to overlap ratio as given in [50]. We assume the training set \( \{ f_i, c_i \}_{i=1, 2, \ldots, N} \) is given having \( N \) number of samples in the frame. \( f_i \) is the feature vector of \( i^{th} \) sample and \( c_i \) is the binary class label for \( i^{th} \) sample i.e. \( c_i \in \{ +1, -1 \} \). SVM classifier is defined as follows:
\[
\min_{h} \frac{1}{2} ||h||^2 + \frac{1}{N} \sum_i l(h; (v_i, c_i)),
\]
where \( h \) is hyperplane of SVM detector, \( l(h; (v, c)) = \max \{ 0, 1, 2, \ldots, N \} \) is given having \( N \) binary class label for \( i^{th} \) sample and \( (v, c) \) is inner product between \( v \) and \( h \). Passive aggressive algorithm is applied to update the hyperplane parameters as follows:
\[
h \leftarrow h - \frac{l(h; (v, c))}{\|\nabla_h l(h; (v, c))\|^2 + \frac{1}{T}} \nabla_h l(h; (v, c)),
\]
where \( T \in (0, +\infty) \) controls the rate of updating of \( h \), \( \nabla_h l(h; (v, c)) \) is the gradient of loss function.

Unlike existing techniques, in our proposed work the parameters of the detectors are updated using (9), when the \( \max(y_i) < T_r \), where \( y_i \) is the response map value for \( i^{th} \) possible state calculated by the detector out of \( i \) number of states in \( X(D_e) \) for \( i^{th} \) frame.

### C. KALMAN FILTER BASED PREDICTOR

In literature different Kalman filter [51] based tracking algorithms have been proposed, for example [52]–[55]. To increase the efficiency of tracking algorithms, Kalman filter based algorithms have been hybridized with many other tracking algorithms, some of the examples are in [56]–[58]. Different from the existing techniques, we incorporated Kalman filter in synchronization with estimator module to avoid the tracking failure caused by long term occlusion, motion blur or clutter background. Kalman filter works in closed loop cycle with prediction and correction steps given by (10)-(14) respectively. In our proposed tracking framework, Kalman filter is activated in case of the failure of tracking caused by any of the above-mentioned issues. During occlusion Kalman filter takes the current state from the main tracking algorithm (in our case it is KCF) defined by (1)-(4) and predicts the next state by using (10) and (11). Main tracking algorithm (KCF) will stop updating its parameters and target appearance regression model. In the next frame Kalman filter corrects itself using the previous location predicted by (10) and (11) during occlusion. Formulation of Kalman filter is given by (10)-(14).

**Prediction:**
\[
x''_{t} = A x'_{t-1} + B u, \tag{10}
\]
\[
S_t = AS_{t-1}AT + Q, \tag{11}
\]
where, \( x''_t \) is predicted state at \( t^{th} \) frame, \( A \) is state transition matrix, \( S_t \) is posteriori error covariance matrix, \( Q \) is covariance matrix of dynamic noise, \( B \) is input noise and \( A \) is state transition matrix.

**Correction:**
\[
K_{t-1} = S_{t-1}HT \left( HS_{t-1}HT + R \right), \tag{12}
\]
\[
X''_{t+1} = X_t + K_{t-1} (y_t - HX''_{t}), \tag{13}
\]
\[
S_{t+1} = (I - KHT)S_t, \tag{14}
\]

where \( H \) is measurement matrix, \( y_t \) is the measurement to the Kalman from the main tracking algorithm. Depending on the condition this measurement may come from either of the three sources i.e.; i) main tracking algorithm ii) estimator module iii) Kalman filter self-prediction in the previous frame. Depending on these two conditions; i) \( \max(y_t) < T_r \) or \( \text{PSR}(y_t) < T_{psr} \) and ii) \( \max(y'_t) > T_r \). Kalman filter continues to predict the next state during occlusion and send predicted state to predictor-estimator collaboration module.

### D. PREDICTOR-ESTIMATOR COLLABORATION

Unlike existing techniques, collaboration module is proposed to handle long term occlusion, motion blur and clutter background. Most of the already existing methods model the target using its appearance. Major problem associated with these methods is their incapability of predicting the state of the object during occlusion. When the object re-enters the field of view of frame after occlusion, different tracking techniques have been proposed to recapture the object such as [3], [14]. Different from existing frameworks, our proposed scheme (AFAM-PEC) activate the predictor and estimator at the same time, when the object gets occluded i.e. \( \max(y_t) < T_r \) or \( \text{PSR}(y_t) < T_{psr} \). During the occlusion period predictor starts predicting the location of the target and SVM based classifier starts estimating the position of the object. If estimated position by SVM based classifier satisfies
FIGURE 1. Pictorial representation of proposed tracking technique. Position is estimated, on the estimated position we have estimated the scale using pyramid technique. Reliability of tracking is judged using conditional block at the right of scale estimation. Result is passed to collaborator if the tracking result is not reliable. SVM classifier estimate the new position and its reliability is checked using second conditional block shown in collaborator module, if the results are reliable enough based on the threshold then new position is SVM estimator based otherwise new position will be given by Kalman filter predictor. where $P_{kf}$ represents the predictor and $D_{rf}$ represents the detector. $y_{s}$ is estimated position at estimated scale, $R_{con}$ is context regression model, $R_{tar}$ is target appearance regression model.

FIGURE 2. Modeling of context regression model and target appearance regression model.

the condition $\max (y'_{i}) > T_{t}$, this position is considered a correct estimate and is given to Kalman filter as measurement to predict the next location. However, if estimated position does not satisfy the condition $\max (y'_{i}) > T_{t}$, Kalman filter based predicted position is given to the estimator to estimate the next location in the next frame and same is given to Kalman filter to predict the next location in the next frame. This approach shows significant improvement in results in comparison to [14], [16], [17]. Flow chart of this module is given in Fig. 3.

IV. IMPLEMENTATION DETAILS

The complete flow of proposed tracking scheme is presented in Algorithm 1. The corresponding flow chart of novel detector-estimator collaboration module is presented in Fig. 3. In this paper, we computed multilayer features at fraction of cost using the technique presented in [59]. Histogram of the oriented gradient with 31 bins in histogram along with histogram of local intensities (HoI) with $6 \times 6$ windows using 8 bins is implemented. To cater fast illumination variations, HoI is applied on brightness channel and transformed brightness channel as given in [60] is implemented. Context regression model $R_{con}$ is trained using forty seven channels feature vector. Whereas target appearance regression model $R_{tar}$ is trained using HoG features with 31 number of bins only. Constant velocity model of Kalman filter is implemented. Gaussian kernel is used in both target appearance regression model and context aware regression model. Correlation in (2) and (3) is computed in Fourier domain. Detection is done by sliding window scanning fashion similar to [61]. SVM classifier is trained considering very large number of samples around the estimated location. Samples having overlap ratio with the target model bounding box greater than 50% are given positive labels, whereas samples having overlap ratio less than 10% are assigned negative labels. Regularization parameter in (1) is assumed to be $10^{-4}$, search window in frame is 180% of the target object size, width of kernel is set 0.1, learning rate $\beta$ is considered 0.01. For scale handling, 21 number of scales are considered and $\alpha_{scale}$ factor is considered 1.08. To turn on the SVM based detector and Kalman filter based predictor, threshold $T_{r}$ is considered 0.25. Detectors results are considered reliable only if the threshold $T_{r} > 0.5$. The second threshold $T_{t}$ is considered 0.5. Threshold $T_{a}$ for updating of target regression model $R_{tar}$.
B. Khan et al.: AFAM-PEC

FIGURE 3. Flow chart of Predictor-estimator collaborator module.

TABLE 2. Challenging aspects for visual object tracking.

| Attribute name       | Abbreviation | Explanation                                                                 |
|----------------------|--------------|-----------------------------------------------------------------------------|
| Occlusion            | OCC          | Target is partially or fully hide behind another object                     |
| Scale variation      | SV           | Bounding boxes ratio of initial frame and present frame is out of range ts. ts > 1 (s=2) |
| Out-of-plane rotation| OPR          | Rotation of target object out of image plane.                               |
| Motion blur          | MB           | Blurring of target region due to motion                                     |
| Fast motion          | FM           | Ground truth motion is greater than 20 pixels.                               |
| Deformation          | DEF          | Non-rigid object deformation                                                |
| Background clutters  | BC           | Target object background having similar color or texture as that of target. |

is 0.5. Most of the parameters are based on [3], [14], with slight variation or no variation at all. The proposed tracking scheme is implemented in MATLAB (2019) on intel core i7, 7th generation, 2.80 GHz processor, RAM 16GB, machine with 64bit windows 10 operating system.

FIGURE 4. Comparison of LCT [14] and proposed algorithm (AFAM-PEC) for jogging1 video. 4(a) Qualitative analysis. 4(b) Distance precision plot; Proposed algorithm (AFAM-PEC) achieved distance precision of 100% at threshold of 20-pixels. 4(c) Distance precision plot; LCT [14] is unable to achieve 100% distance precision at threshold of 20-pixels.

V. RESULTS AND DISCUSSIONS

Our proposed tracking scheme is evaluated and compared on number of selected videos form benchmark datasets OTB50 [23], OTB100 [24], TColor-128 [25], and UAV-128 [26]. OTB50 contains 50 videos, OTB100 contains 100 videos, TColor contains 128 color sequence and UAV-128 contains 128 videos, captured using unmanned air vehicle. Each video has one or more object tracking challenges associated with it. We choose the videos having seven attributes namely; i) occlusion ii) scale variation iii) motion blur iv) fast motion v) out-of-plane rotation vi) deformation and vii) background clutter to support and evaluate our proposed tracker. Explanation of each attribute is given in TABLE 2. Qualitative evaluation is given in Fig.4a, Fig. 5a, Fig. 6a, Fig. 7, and Fig. 8. We evaluate our tracker quantitively using; i) Distance precision metric as per Fig. 4b, Fig. 4c, Fig. 5b, Fig 5c, Fig. 6b, Fig. 6c and TABLE 3, ii) Overlap success rate metric as per Fig. 9, Fig. 10 and TABLE 4. Processing time comparison is also given in TABLE 5. We compared our tracker on benchmark dataset videos with long term correlation tracker (LCT) [14], spatio-temporal context learning (STC) [16], and real time compressive tracking (CT) [17].

This paragraph explains the attributes of each video which also known as challenging aspects by visual tracking community. Six videos shown in Fig. 7 i.e. Jogging1, Jogging2, Walking2, Human3, Girl2, Skating2 are selected from OTB100 dataset. These six videos are also part of TColor-128...
TABLE 3. Quantitative analysis; distance precision at threshold of 20 pixels over 11 challenging videos selected from OTB50, OTB100, tcOLOR-128, and UAV-123.

| S. NO. | Sequence | OUR AFAM-PEC | LCT [14] | STC [16] | CT [17] |
|--------|----------|--------------|----------|----------|---------|
| 1      | Jogging1 OTB100/tColor-128 | **0.9739** | 0.967 | 0.208 | 0.2 | 21 |
| 2      | Jogging2 OTB100/tColor-128 | **0.9902** | 0.970 | 0.172 | 0.1 | 66 |
| 3      | Walking2 OTB100/tColor-128 | **0.722** | 0.406 | 0.442 | 0.3 | 82 |
| 4      | Girl2 OTB100/tColor-128 | **0.940** | 0.186 | 0.262 | 0.1 | 88 |
| 5      | Human3 OTB100/tColor-123 | **0.795** | 0.0135 | 0.088 | 0.0 | 50 |
| 6      | Skating2 OTB100/tColor-128 | | 0 | 0.0233 | **0.2** | 17 |
| 7      | Busstation cel clr TColor-128 | **0.9835** | 0.102 | 0.099 | 0.1 | 02 |
| 8      | Bike3 UAV-123 | 0.124 | 0.151 | **0.206** | 0.0 | 62 |
| 9      | Car4 UAV-123 | **0.98** | 0.98 | 0.64 | 0.2 | 85 |
| 10     | Car9 UAV-123 | **0.917** | 0.85 | 0.201 | 0.2 | 12 |
| 11     | Building3 UAV-123 | 1 | 1 | **.963** | .38 | 6 |
| Mean Success Rate | - | **0.765** | 0.511 | 0.30 | 0.2 | 06 |

dataset. Whereas, Fig. 8 shows five videos i.e. Bike3, Car4, Car9, Busstation and Building3. Out of these five videos, four videos are part of UAV-123 Dataset and single video Busstation is from TCOLOR-128 dataset. Hence total of six videos are from OTB100, seven video sequences from TCOLOR-128 and four video sequences from UAV-123 are used to evaluate our propose AFAM-PEC tracker. Jogging1 and Jogging2 sequences have Occlusion, Deformation and Out of plane rotation. Walking2 sequence has attributes of scale variation, occlusion and low resolution. Girl2 and Human3 video sequence have maximum challenges i.e. 5. Challenges associated with Girl2 video sequence are namely, scale variation, occlusion, deformation, motion blur and out of plane rotation. Whereas Human3 video contains scale variation, occlusion, deformation, out of plane rotation and background clutter. Skating2 sequence has four attributes associated with it i.e. scale variation, occlusion, fast motion and out of the plane rotation. Bike3 contains Fast Motion, Occlusion and Out of Plane rotation. Car4 and Car9 has Occlusion and Scale Variation. Bustation video sequence has Clutter Background and Occlusion. Finally, Building3 video sequence contains Out of the Plane Rotation. Therefore, a total of seven attributes are associated with these selected eleven videos. Each attribute is explained in Table 2.

Quantitative results in Table 3 and TABLE 4 show that our tracker perform well for long term occlusion challenge. For Jogging2 sequence, the proposed tracker gives distance precision of 100% at 20-pixel threshold and LCT [14] tracker gives the second highest precision of 97%. Whereas all the remaining trackers fail to track the object after occlusion. Likewise, on Jogging1 sequence, the proposed scheme again achieves 99% precision. On Walking2 sequence the proposed tracking scheme again gives 100% distance precision. Whereas all the remaining three trackers loses the target when the girl in the video sequence gets occluded with boy at frame number 202. On Girl2 sequence, the proposed tracking algorithm again achieves good performance having distance precision of 95% and all the other trackers fail to track the target object. On Human3 video sequence our tracker outperforms all the remaining three trackers by achieving distance precision of 99%. It is also worth mentioning that human3 video contains five challenging attributes out of total nine attributes given in [21]. Skating2 video sequence has extra challenging attribute of fast motion. Though the proposed tracker along with all the other trackers fail to track the target object but still our tracker achieves the second highest distance precision and tracks the target object for a greater number of frames than LCT [14]. On this sequence CT [17] tracker tracks the...
target object more than any other tracker. Bustation3 video sequence selected from TColor-128 dataset contains severe occlusion and cluttering. Hence all the tracker loses the target very early while the proposed tracking scheme AFAM-PEC achieves the distance precision of 100%. On Bike3 video sequence all the trackers fail to track the target. The proposed AFAM-PEC achieves the highest precision of 38% among all. On Car4 video sequence again our proposed tracking scheme achieves the 100% distance precision outperforming the all other trackers. Similarly, on Car9 video sequence our proposed tracker and LCT [14] both gave the distance precision of 98% but CT [17] and STC [16] losses the target and gave the precision of less than 25%. Building3 sequence is relatively simpler without having much challenging aspects. So, all the trackers successfully track the target by achieving 100% distance precision. Mean distance precision is also given in TABLE 3. Our proposed AFAM-PEC achieves the highest mean distance precision of 85%, LCT [14] achieves the second highest mean precision of 54%, STC [16] achieves the third highest mean distance precision of 38% while CT [17] with lowest mean distance precision of 26%.

We first implemented Kalman filter based tracking algorithm, as Kalman filter is a measurement follower algorithm. The output of LCT [14] algorithm is given as measurement to Kalman to see the results. Fig. 4 shows the results for LCT [14] and Kalman based tracking algorithm. Kalman filter based tracking algorithm achieves distance precision of almost 100% but the LCT [14] and most of the traditional algorithms achieves less than 100%. This is because when object gets occluded, LCT [14] stop estimating correct position of the object and when the object comes out of occlusion LCT [14] tracker re-detects the target object as shown in Fig. 4. In our implementation Kalman based tracker continuously predicts the new state of the target object even during occlusion which increases the distance precision. To further investigate this behavior, these algorithms have been applied to other videos and the results are shown in Fig. 5 along with the distance precision plot. Frame number 62, 65 and 67 observe this behavior. Fig. 6 (Walking2 video sequence) represents another interesting fact that if the measurement (by baseline tracker in our case LCT [14]) given to Kalman filter is wrong then Kalman will predict the false state in next frame as it is a measurement follower. Now if the baseline tracker continues to give the wrong measurement to Kalman filter based tracker even after the occlusion of the target is over, Kalman filter will be predicting the false states and target will be lost. This phenomenon is clearly represented in Fig. 6, where after 234th frame, baseline tracker misguides the Kalman filter and both the algorithm starts following wrong object. This behavior is corrected by proposing the algorithm which works by using the collaboration of the predictor and estimator. Table 3 shows the distance precision of 100% over this video sequence. To further strengthen our argument, Fig. 11 presents the peak to side lobe ratio and. This figure depicts that the proposed tracking algorithm achieves a higher PSR earlier than the LCT [14] algorithm. Distance precision plot of Jogging1, Jogging2 and Walking2 sequences are given in Fig. 4, Fig. 5 and Fig.6 respectively, which shows that the proposed tracking scheme outperforms the other three tracking algorithms.

Before qualitative analysis, let us analyze the overlap success rate metric. Fig. 9 shows the comparison of the proposed algorithm with each of LCT [14], STC [16], and CT [17]. Whereas, Fig. 10 gives the comparison of all four tracking algorithms on single plot. Table 4 shows the overlap success rate for various video sequences at threshold of 0.5. For Jogging1 video LCT [14] and proposed algorithm gives almost equal success rate of 97%. CT [17] and STC [16] gives success rate of 20% and 22% as they fail to track after occlusion. The algorithm proposed in this work achieves the success rate of 99% for Jogging2 video. Whereas, LCT [14] achieves 97% while other two algorithms achieve less than 20%. Similarly, on all the other remaining video sequences AFAM-PEC achieves the highest success rate, details are given in TABLE 4. Mean overlap success rate is also calculated. Proposed AFAM-PEC outperforms the other trackers by achieving mean overlap success rate of over 75%. LCT [14] achieves the mean success rate of 51%, STC [16] achieves 30%, and CT [17] achieves 20%.

TABLE 5 shows the frames processed per second (FPS) time of all the 4 trackers over eleven selected videos.
Proposed tracking scheme AFAM-PEC shows not much increase in computational cost while maintaining higher tracking efficiency in terms of mean precision and overlap success rate. For example, our proposed AFAM-PEC process 27.89 frames per second, whereas LCT [14] process 28.30 frames per second. It is difference of even less than one frame. Similar is the case on all the other videos. CT [17] and STC [16] loses the target in most of the video sequences, that is why TABLE 5 shows high FPS for CT [17] and STC [16].

For the qualitative analysis, the results of four trackers i.e. this paper (AFAM-PEC), LCT [14], STC [16] and CT [62], over eleven video sequences are presented in Fig. 7 and Fig. 8. Top to bottom rows of Fig. 7 contain Jogging1, Jogging2, Girl2, Human3, Walking2 and Skating2 video sequences. Row wise...
FIGURE 8. Qualitative results of the proposed scheme (AFAM-PEC), LCT [14], STC [16], and CT [17] on five challenging sequences selected from TColour-128 and UAV-123 databases. First row to the last row: bike3, car4, car9, busstation, and building3 video sequences are presented respectively.

analysis is given in this paragraph. In the first row all the trackers successfully track the object until frame number 71. While in frame number 79 the proposed algorithm is the only one to track the object exactly while all the others have window on pole instead of target object. At frame number 91, LCT [14] tracker successfully redetects the target. After this frame the proposed algorithm and LCT [14] successfully tracks the object till the end of video while other two algorithms fail to track after the occurrence of occlusion. Similarly, in the second row of Fig. 7, the proposed algorithm successfully tracks the object after occlusion, earlier than all the remaining three algorithms. LCT [14] shows second best behavior over this sequence by tracking the object successfully till the end, whereas remaining two algorithms fail to track the object when it reappears after the occlusion. The only issue with LCT [14] reported for Jogging1 and Jogging2 is the estimation of the position of object during occlusion. In the third row of Fig. 7, for Girl2 video sequence, all the trackers successfully track the object until occurrence of cluttering. It can be seen in frame number 98 after the cluttering that all the trackers are successful in tracking, but CT [17] fails to track. When the second challenge, associated with this video occurs i.e. full occlusion, all the trackers fail after the reappearance of the target object except the proposed tracker.

Tracker proposed in this paper successfully tracks the target object after occlusion which is visible in frame number 168. We run all the trackers over this video for 600 frames.
TABLE 4. Quantitative analysis; overlap success rate at threshold of 0.5 pixels over 11 challenging videos selected from OTB50, OTB100, tcolor-128, and uav-123.

| S. NO. | Sequence                  | OUR AFAM-PEC | LCT [14] | STC [16] | CT [17]  |
|-------|---------------------------|--------------|----------|----------|----------|
| 1     | Jogging 1 OTB100/TColor-128 | 0.997        | 0.970    | 0.228    | 0.224    |
| 2     | Jogging 2 OTB100/TColor-128 | 1            | 0.973    | 0.185    | 0.175    |
| 3     | Walking2 OTB100/TColor-128 | 1            | 0.404    | 0.794    | 0.436    |
| 4     | Girl2 OTB100/TColor-128   | 0.947        | 0.19     | 0.27     | 0.115    |
| 5     | Human3 OTB100/TColor-123  | 0.988        | 0.018    | 0.100    | 0.055    |
| 6     | Skating2 OTB100/TColor-128| 0.070        | 0.019    | 0.090    | 0.190    |
| 7     | Busstation cel clr TColor-128 | 1           | 0.113    | 0.110    | 0.108    |
| 8     | Bike3 UAV-123             | 0.379        | 0.269    | 0.275    | 0.069    |
| 9     | Car4 UAV-123              | 1            | 0.997    | 0.991    | 0.294    |
| 10    | Car9 UAV-123              | 0.985        | 0.982    | 0.216    | 0.212    |
| 11    | Building3 UAV-123         | 1            | 1        | 1        | 1        |
| Mean precision | -            | 0.85         | 0.54     | 0.387    | 0.261    |

The fourth row of Fig. 7 contains the images from Human3 sequence, again at this sequence our proposed algorithm shows better results i.e. all the trackers lose the object in the start of sequence but the proposed tracker successfully tracks the object, which is visible in frame numbers 252, 301 and 410. All the trackers are tested over 590 frames of this sequence. In Walking2 sequence, the target object gets occluded at frame number 50 and after this frame, CT [17] and LCT [14] loses the target and starts following wrong object. STC [16] is still able to track the object after occlusion but fails to handle the scale properly, whereas our proposed scheme successfully tracks the object keeping the right scale. Last sequence shown in Fig. 7 is Skating2. Although not any tracker is able to track the object over full video sequence, but our proposed algorithm tracks the object over a greater number of frames than the state of the art i.e. LCT [14]. The proposed algorithm tracks the object until 125th frame and after this it starts drifting, whereas CT [17] tracks the object for the maximum number of frames. It can be seen in frame number 484 that all the trackers lose the target. First, STC [16] loses the target after this LCT [14] then our proposed tracker and at the end CT [17] tracker loses the target. Although our proposed tracker loses the target before CT [17] in this video sequence, but it has benefit of performing better then CT [17] on all the other videos which contain six challenges as per TABLE. 2 excluding fast motion. Top to bottom rows of Fig. 8 contain Bike3, Car4, Car9, Busstation, and building3.
TABLE 5. Quantitative analysis; frames per second (fps) of 11 challenging videos selected from OTB50, OTB100, tcolor-128, and uav-123.

| S. No. | Sequence           | OUR AFAM-PEC | LCT [14] | STC [16] | CT [17] |
|--------|--------------------|--------------|----------|----------|---------|
| 1      | Jogging1 OTB100/TColor-128 | 27.89        | 28.30    | 34.37 target lost | 24.73 target lost |
| 2      | Jogging2 OTB100/TColor-128 | 21.51        | 22.12    | 38.23 target lost | 27.07 target lost |
| 3      | Walking2 OTB100/TColor-128 | 24.65        | 26.39    | 34.68 scale issue | 24.24 target lost |
| 4      | Girl2 OTB100/TColor-128  | 15.62        | 16.63    | 12.50 target lost | 21.96 target lost |
| 5      | Human3 OTB100/TColor-123 | 21.18        | 18.09 target lost | 131.02 target lost | 19.93 target lost |
| 6      | Skating2 OTB100/TColor-128 | 15.28        | 20.71 target lost | 58 target lost | 27.82 target lost |
| 7      | Busstation_cel_clr TColor-128 | 42.40        | 54.55 target lost | 20.67 target lost | 12.60 target lost |
| 8      | Bike3 UAV-123        | 33.14        | 57.47 target lost | 22.83 target lost | 13.03 target lost |
| 9      | Car4 UAV-123         | 13.01        | 16.75    | 23.16 target lost | 13.08 target lost |
| 10     | Car9 UAV-123         | 10.44        | 12.71 scale issue | 29.23 target lost | 18.18 target lost |
| 11     | Building3 UAV-123    | 14.60        | 16.55    | 22.89 target lost | 13.19     |

video sequences. These sequences are made using UAV. In Bike3 sequence target is relatively small as compared to other video sequences. In this video not any tracker is able to track the target correctly but still proposed tracker performs better and track the target correctly up to 43rd frame. In second row of Fig. 8 Car4 video sequence is show. In this video our proposed AFAM-PEC and LCT [14] both track the target successfully till the end of video but proposed AFAM-PEC achieves better overall success rate which can be seen in frame numbers 270, 274 and 600. STC [16] also able to track the target till the end of video but CT [62] fails to track the object after the occlusion, which is visible in frame number 235. In the third row of Fig. 8, all the trackers successfully track the car till the occurrence of occlusion in frame number 42. At occlusion LCT [14] struck while our AFAM-PEC successfully tracks the car even when it is occluded. After the occlusion STC [16] also fails to track correctly as per frame number 270. While proposed AFAM-PEC and LCT [14] track the object till the end of video sequence. Though visually it seems that both the trackers have similar performance but as per quantitative analysis our AFAM-PEC gives better distance precision and overlap success rate. In second last row of Fig. 8 AFAM-PEC outperforms all the other trackers by successfully tracking the target after occlusion in frame number 51. All the remaining three trackers fail to track the object after occlusion in this video sequence. Last row of Fig. 8 shows the building3 sequence. All the trackers successfully track the target because of simplicity of the video.

VI. CONCLUSION

In this study, adaptive correlation filter based tracking failure avoidance mechanism is presented. Kernelized correlation filter is used as baseline tracker. Failure avoidance mechanism is proposed and integrated with Kernelized correlation filter. In our proposed scheme we first address the occlusion detection problem by soughing two parameters form the response map i.e. i) peak to side lobe ratio ii) peak correlation value. These two parameters work together to detect the occlusion. Second, we incorporated Kalman filter based predictor and SVM based estimator to kernelized correlation filter. Third, we proposed collaboration module between predictor and estimator to avoid the tracking failure. We choose videos from the standard three datasets (OTB100, TColor-128, and UAV-123) having six challenging attributes to perform the experiments. With the help of experiments, it is
shown by us that proposed approach (AFAM-PEC) performs better against state-of-the-art tracking algorithms in terms of distance precision and overlap threshold. However, Kalman is continuously predicting the location of the object whether the object is is present in the frame or not. Efficiency of the tracker may be further enhanced by designing criterion to stop prediction of Kalman filter if the object is out of view/occluded for some specified time.

REFERENCES

[1] S. Liu, Y. Wu, E. Wei, M. Liu, and Y. Liu, “StoryFlow: Tracking the evolution of stories,” IEEE Trans. Vis. Comput. Graphics, vol. 19, no. 12, pp. 2436–2445, Dec. 2013, doi: 10.1109/TVCG.2013.196.
[2] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” ACM Comput. Surv., vol. 38, p. 45, Dec. 2006, doi: 10.1145/1177352.1177355.
[3] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, “Adaptive correlation filters with long-term and short-term memory for object tracking,” Int. J. Comput. Vis., vol. 126, no. 8, pp. 771–796, Aug. 2018, doi: 10.1007/s11263-018-1061-3.
[4] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” in Proc. IEEE Comput. Soc. Conf. Vis. Pattern Recognit., Jun. 2010, pp. 2544–2550, doi: 10.1109/CVPR.2010.5539960.
[5] A. Lukežič, T. Vojíř, L. Č. Zajec, J. Matas, and M. Kristan, “Discriminative correlation filter tracker with channel and spatial reliability,” Int. J. Comput. Vis., vol. 126, no. 7, pp. 671–688, Jul. 2018, doi: 10.1007/s11263-017-0106-1.
[6] P. M. Kaju, D. Mishra, and R. K. S. S. Gorthi, “Detection based long term tracking in correlation filter trackers,” Pattern Recognit. Lett., vol. 122, pp. 79–85, May 2019, doi: 10.1016/j.patrec.2019.02.028.
[7] L. Zhang and P. N. Suganthan, “Robust visual tracking via co-trained kernelized correlation filters,” Pattern Recognit., vol. 69, pp. 82–93, Sep. 2017, doi: 10.1016/j.patcog.2017.04.004.
[8] A. Lukežič, L. Č. Zajec, T. Vojíř, J. Matas, and M. Kristan, “FacCoLoT — A fully-correlational long-term tracker,” in Proc. Asian Conf. Comput. Vis., in Lecture Notes in Computer Science: Including subsequences Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, 2019, pp. 595–611, doi: 10.1007/978-3-030-20890-5_38.
[9] M. Zhang, Q. Wang, X. J. Xing, J. Gao, P. Peng, W. Hu, and S. Maybank, “Visual tracking via spatially aligned correlation filters network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2019, pp. 2411–2418, doi: 10.1109/CVPR.2019.3.12.
[10] D. Yuan, N. Fan, and Z. He, “Learning target-focusing convolutional regression model for visual object tracking,” Knowl.-Based Syst., vol. 194, Apr. 2020, Art. no. 105526, doi: 10.1016/j.kaosys.2020.105526.
[11] D. Yuan, X. Li, Z. He, Q. Liu, and S. Lu, “Visual object tracking with adaptive structural convolutional network,” Knowl.-Based Syst., vol. 194, Apr. 2020, Art. no. 105554, doi: 10.1016/j.kaosys.2020.105554.
[12] Y. Wu, J. Lim, and M. H. Yang, “Online object tracking: A benchmark,” in Proc. IEEE Comput. Soc. Conf. Vis. Pattern Recognit., Jun. 2013, pp. 1834–1848, doi: 10.1109/CVPR.2013.3.12.
[13] Y. Wu, J. Lim, and M.-H. Yang, “Object tracking benchmark,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1834–1848, Sep. 2015, doi: 10.1109/TPAMI.2014.2388226.
[14] P. Liang, E. Blasch, and H. Ling, “Encoding color information for visual tracking: Algorithms and benchmark,” IEEE Trans. Image Process., vol. 24, no. 12, pp. 5630–5644, Dec. 2015, doi: 10.1109/TIP.2015.2482905.
[15] M. Mueller, N. Smith, and B. Ghanem, “A benchmark and simulator for UAV tracking,” in Proc. Eur. Conf. Comput. Vis., in Lecture Notes in Computer Science: Including subsequences Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, 9905, 2016, pp. 445–461, doi: 10.1007/978-3-319-46448-0_27.
[16] V. Ramakrishna and M. G. Alex, “A survey on visual object tracking: Datasets, methods and metrics,” Int. Res. J. Eng. Technol., vol. 3, no. 11, pp. 1–6, 2016.
[17] X. Li, Y. Li, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, “A survey of appearance models in visual object tracking,” ACM Trans. Intell. Syst. Technol., vol. 4, pp. 1–48, Sep. 2013.
[18] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, “A survey of appearance models in visual object tracking,” J. Vis. Commun. Image Represent., vol. 33, no. 8, pp. 1619–1632, Aug. 2011, doi: 10.1016/j.jvcir.2011.05.005.
[19] S. Avidan, “Ensemble tracking,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 2, pp. 261–271, Feb. 2007, doi: 10.1109/TPAMI.2007.35.
[20] Q. Bai, Z. Wu, S. Sclaroff, M. Betke, and C. Monnier, “Randomized ensemble tracking,” in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2013, pp. 2040–2047, doi: 10.1109/ICCV.2013.255.
[21] W. Li and Y. Lin, “Adaptive randomized ensemble tracking using appearance variation and occlusion estimation,” Math. Problems Eng., vol. 2016, pp. 1–11, Jan. 2016, doi: 10.1155/2016/870479.
[22] B. Babenko, M.-H. Yang, and S. Belongie, “Robust object tracking with online multiple instance learning,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 8, pp. 1619–1632, Aug. 2011, doi: 10.1109/TPAMI.2010.226.
[23] J. Gao, H. Ling, W. Hu, and X. Jing, “Transfer learning based visual tracking with Gaussian processes regression,” in Proc. Eur. Conf. Comput. Vis., in Lecture Notes in Computer Science: Including subsequences Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, 8691, 2014, pp. 188–203, doi: 10.1007/978-3-319-10578-9_13.
[24] B. Huang, T. Xu, J. Li, Z. Shen, and Y. Chen, “Transfer learning-based discriminative correlation filter for visual tracking,” Pattern Recognit., vol. 100, Apr. 2020, Art. no. 107157, doi: 10.1016/j.patcog.2019.107157.
[25] Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-learning-detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 7, pp. 1409–1422, Jul. 2012, doi: 10.1109/TPAMI.2011.239.
Y. Chen, P.-H. Chen, and S.-C. Chien, “An objective tracking method based on improved TLD,” *Algorithms*, vol. 13, no. 1, p. 15, Jan. 2020, doi: 10.3390/ai3010015.

B. V. K. V. Kumar, A. Mahalanobis, and R. D. Juday, *Correlation Pattern Recognition*. Cambridge, UK.: Cambridge Univ. Press, 2005.

J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “High-speed tracking with kernelized correlation filters,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015, doi: 10.1109/TPAMI.2014.2345390.

C. Ma, X. Yang, C. Zhang, and M.-H. Yang, “Learning a temporally invariant representation for visual tracking,” in *Proc. IEEE Int. Conf. Image Process.* (ICIP), Sep. 2015, pp. 857–861, doi: 10.1109/ICIP.2015.7350921.

M. Danelljan, F. S. Khan, M. Felsberg, and J. V. D. Weijer, “Adaptive color attributes for real-time visual tracking,” in *Proc. IEEE Comput. Soc. Pattern Recog.*, Jun. 2014, pp. 1090–1097, doi: 10.1109/CVPR.2014.143.

Y. Qi, L. Qin, S. Zhang, Q. Huang, and H. Yao, “Robust visual tracking via scale-and-state-awareness,” *Neurocomputing*, vol. 329, pp. 75–85, Feb. 2019, doi: 10.1016/j.neucom.2018.10.035.

M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Accurate scale estimation for robust visual tracking,” in *Proc. Brit. Mach. Vis. Conf.* (BMVC), 2014, pp. 1–11, doi: 10.5244/c.28.65.

M. Zhang, J. Xing, J. Gao, and W. Hu, “Robust visual tracking using joint scale-spatial correlation filters,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 1468–1472, doi: 10.1109/ICIP.2015.731044.

J. Fan, H. Song, K. Zhang, Q. Liu, F. Yan, and W. Lian, “Real-time manifold regularized context-aware correlation tracking,” *Frontiers Comput. Sci.*, vol. 14, no. 2, pp. 334–348, Apr. 2020, doi: 10.1007/s11704-018-5100-y.

K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, “Visual tracking via adaptive spatially-regularized correlation filters,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.* (CVPR), Jun. 2019, pp. 4665–4674, doi: 10.1109/CVPR.2019.00480.

Y. She and Y. Yi, “Learning multi-feature based spatially regularized and scale adaptive correlation filters for visual tracking,” in *Proc. Int. Conf. Multimedia Modeling*, in Lecture Notes in Computer Science: Including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, vol. 11961, 2020, pp. 480–491, doi: 10.1007/978-3-030-37731-1_39.

K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

J. S. Supancic, III, and D. Ramanan, “Self-paced learning for long-term tracking,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2379–2386, doi: 10.1109/CVPR.2013.308.

Y. Hua, K. Alahari, and C. Schmid, “Occlusion and motion reasoning for long-term tracking,” in *Proc. Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science: Including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, vol. 801, 1994, pp. 151–158, doi: 10.1007/bfb0028345.

P. Dollár, R. Appel, S. Belongie, and P. Perona, “Fast feature pyramids for object detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014, doi: 10.1109/TPAMI.2014.2300479.

R. Zabih and J. Woodfill, “Non-parametric local transforms for computing visual correspondence,” in *Proc. Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science: Including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, vol. 801, 1994, pp. 151–158, doi: 10.1007/bfb0028345.

P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog. (CVPR)*, vol. 1, Dec. 2001, p. 1, doi: 10.1109/cvpr.2001.990517.

K. Zhang, L. Zhang, and M. Yang, “Real-time compressive tracking,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, in Lecture Notes in Computer Science, vol. 7574, 2012, pp. 864–877, doi: 10.1007/978-3-642-33712-3_62.

AHMAD ALI received the bachelor’s degree in computer science from the COMSATS Institute of Information Technology, Pakistan, in 2010, and the M.S. degree in electrical engineering from the University of Engineering and Technology at Taxila, Taxila, Pakistan, in 2013. He is currently pursuing the Ph.D. degree in electronics engineering with International Islamic University Islamabad, Islamabad, Pakistan.

He has authored one journal and four conference publications. His research interests include computer vision, pattern recognition, and image processing.

Mr. Khan’s awards and honors include the campus silver medal and the institute’s silver medal for the B.S. degree in computer engineering.

ABDUL JALIL received the bachelor’s and master’s degrees in electronics from Quaid-i-Azam University, Islamabad, Pakistan, in 1986 and 2000, respectively, and the Ph.D. degree in image and signal processing from Mohammad Ali Jinnah University, Pakistan, in 2006. Then, he remained affiliated with the University of Sussex, as a Postdoctoral Researcher for nine months, from 2008 to 2009. From 2002 to 2016, he served as an Academician with the Pakistan Institute of Engineering and Applied Sciences (PIEAS), Islamabad, Pakistan, in 2010 and 2015, respectively. He has authored several publications in image and speech processing. He is currently doing research in the fields of computer vision and artificial intelligence.

BABER KHAN received the B.S. degree in computer engineering from the COMSATS Institute of Engineering and Technology (UET) Lahore, Lahore, Pakistan, in 2003, and the M.S. degree in systems engineering and the Ph.D. degree from the Pakistan Institute of Engineering and Applied Sciences (PIEAS), Islamabad, Pakistan, in 2010 and 2015, respectively. He has authored several publications in image and speech processing. He is currently doing research in the fields of computer vision and artificial intelligence.
KHIZER MEHMOOD received the B.Sc. degree in electronics engineering from International Islamic University Islamabad (IIUI), Islamabad, Pakistan, in 2010, and the M.Sc. degree in electrical engineering from the University of Engineering and Technology at Taxila, Pakistan, in 2013. He is currently pursuing the Ph.D. degree with International Islamic University Islamabad. His areas of interests are computer vision, visual object tracking, and image processing.

MARIA MURAD received the bachelor’s degree in telecommunication engineering from the University of Engineering and Technology (UET), Peshawar, Pakistan, in 2010, and the master’s degree in electrical engineering from the UET at Taxila, Taxila, Pakistan, in 2012. She is currently pursuing the Ph.D. degree in image processing with International Islamic University Islamabad (IIUI), Islamabad, Pakistan. She has been working as a Lecturer with International Islamic University Islamabad, since April 2011. Her research interests are signal and image processing.

HAMDAN AWAN (Member, IEEE) received the Ph.D. degree in computer science and engineering from the University of New South Wales (UNSW), Sydney, Australia, in August 2017. He stayed with York University, Canada, as a Post-Doctoral Fellow, for two years, from December 2017 to December 2019, where he worked on the DARPA’S Radio Bio Project. In December 2019, he joined the Telecommunications Systems and Software Research Group, Waterford Institute of Technology, Ireland, as an H2020 Research Fellow, where he is working on the FET-Open Gladiator Project. He has so far published more than 25 research papers in highly selective IEEE TRANSACTIONS and the IEEE/ACM conferences. He has also co-authored the article that received the Best Paper Award from the IEEE/ACM NanoCom Conference, in 2018. His major research interests include molecular communications, nano-networks, information theory aspects of biological communication, and computer vision. He has also been a recipient of Post Doctoral Writing Fellowship after the Ph.D. degree at the UNSW, for three months, from September to November 2017.