Applied Soft Computing Journal 73 (2018) 1081–1094

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Scaled and oriented object tracking using ensemble of multilayer perceptrons

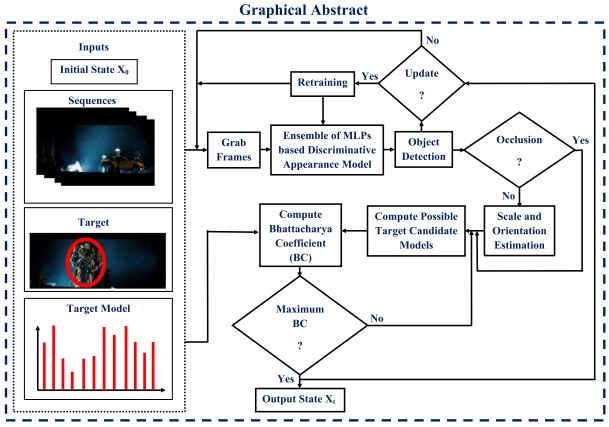
Ajoy Mondal [a,*], Ashish Ghosh [a], Susmita Ghosh [b]

[a] Machine Intelligence Unit, Indian Statistical Institute, 203 B. T. Road, Kolkata 700108, India
[b] Department of Computer Science and Engineering, Jadavpur University, Kolkata 700032, India

HIGHLIGHTS

- An algorithm is proposed to track scaled and oriented object.
- An ensemble of MLP is proposed to detect object from its cluttered background.
- Orientation and enhanced scale of detected object are estimated using binary moments.
- One heuristic based on support value is proposed to reduce drift.
- Another heuristic based on confidence score is proposed to detect full occlusion.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 7 April 2018
Received in revised form 20 August 2018
Accepted 20 September 2018
Available online 9 October 2018

Keywords:
Scaled and oriented object tracking
Ensemble of multilayer perceptron
Estimation of scale and orientation of object
Object detection

ABSTRACT

Major challenging problems in the field of moving object tracking are to handle changing in scale and orientation, background clutter and large variation in pose with occlusion. This article presents an algorithm to track moving object under such complex environment. Here, a discriminative model based on an ensemble of multilayer perceptrons (MLPs) is proposed to detect object from its cluttered background. Orientation and enhanced scale of the detected object is estimated using binary moments. Here, the problem of object tracking is posed as a constrained optimization with respect to location, scale and orientation of the object. Two different heuristics based on support value and confidence score are proposed to reduce drift and to detect full occlusion. Three benchmark datasets are considered for the experimental purpose and the proposed algorithm attains state-of-the-art performance under various conditions.

1. Introduction

In computer vision, moving object tracking has various applications including surveillance, traffic monitoring, vehicle navigation, etc [1–5]. Given a state (e.g., position, scale and orientation) of a target in an initial frame, the estimation of state of the target in the subsequent frames, is the goal of a tracking algorithm [2]. Various conditions like variation in object's scale and orientation, noisy frames, complex object motion, occlusion, background clutter, deformable object's shape, etc. make tracking as a challenging task [1–3,6,7]. In the literature, numerous tracking algorithms have been invented to solve such problems [8–19].

Now a day, tracking is considered as a tracking-by-detection problem. Various statistical model is taken into consideration for

* Corresponding author.
E-mail addresses: ajoy.mondal83@gmail.com (A. Mondal), ash@isical.ac.in (A. Ghosh), susmitaghoshju@gmail.com (S. Ghosh).

https://doi.org/10.1016/j.asoc.2018.09.028
1568-4946/

detecting an object [12]. Among various generative and discriminative appearance models [8–19], discriminative appearance models consider tracking as a two class classification problem. Goal of these models is to maximize the discrimination between the object and its surroundings. Therefore, it basically focuses on discovering the most informative features for visual tracking [3]. Tracking algorithms [12–18] using discriminative appearance models are popular in the literature. In all these approaches, a linear discriminative appearance model is considered as base classifier to detect object from its background. Background clutter in most of the real life video scenes, interposes non-linear decision boundary between the object and the background in the feature space. It is already well established that an ensemble of large number of linear (weak) classifiers can learn non-linear decision boundaries [20].

On the other hand, a multilayer perceptron (MLP) can learn non-linearity present in the decision boundary [21]. To take such advantage, we consider the MLP as a base classifier to distinguish the object from its surroundings. However, performance of an MLP depends on some parameters like its network architecture, learning rate, weighted connection between the neurons, etc [21, 22]. Results thus obtained using an MLP may sometimes alter by changing its architecture [21]. Determination of the network architecture that provides optimal results is a crucial problem [22]. Sometimes, it is easy to combine decisions of individual MLPs to provide near optimal solution and to maximize the generalization capability [20]. In literature, many authors have proposed various techniques for fusing results of different classifiers [20]. Among them, AdaBoost [23] is popularly used for different applications. However, neural network based approaches seem to be able to provide better results [24]. In this regard, the decision combination neural network (DCNN) proposed by Lee and Srihari [25] is a multilayer perceptron which can combine decisions of different classifiers. In this article, to take advantages of both MLP and ensemble technique, we consider ensemble of multilayer perceptrons as a discriminative appearance model to detect object from its complex background.

In case of tracking-by-detection techniques [3], prior knowledge about the object location is available for the initial frame (or a few initial frames). Appearance models need to be updated to cope with the changes in object appearance. Therefore, tracking has been done by finding a position in the current frame, having maximum classification score using sliding window approach in a neighborhood of the object location in the previous frame [3]. If the object changes both its scale and orientation during movement, the tracker needs to be adapted accordingly for tracking the object properly and to reduce the drift problem (due to variation in object's scale and orientation) in the subsequent frames. In ASLA [8], SCM [11] and TGPR [18], variation in scale and orientation of the target is handled by considering affine motion model in Bayesian framework. These techniques are unable to handle large variation both in scale and orientation of the target due to affine motion model. However, in [17], scale of the target is estimated by considering scale pyramid. It searches the best scale of the target among different target candidates (with different scales). But it is unable to estimate the exact scale of the target when object changes both its scale and orientation. Therefore, it is a key issue to properly track the object when it changes its scale and orientation together.

To the best of the authors' knowledge, in the literature, there are limited number of tracking-by-detection methods which explicitly consider adaptation of both scale and orientation together during tracking. In this article, an algorithm is proposed to explicitly adapt changes both in scale and orientation of the object during tracking. A preliminary experiment of this work was reported in [26]. A robust analysis of the results with application in different complex video sequences and comparison with more recent state-of-the-art techniques is reported in the present article. Thus, this article presents two contributions:

(i) Use of MLPs with DCNN to cope with non-linear changes in visual appearance of the moving object.

(ii) A joint estimation of object's scale and orientation during tracking.

In this article, we propose an algorithm to track an object with variation both in its scale and orientation. An ensemble of MLPs based discriminative appearance model is proposed to detect object from its cluttered background. Here, DCNN [25] is considered to combine the decisions of various MLPs (with different architecture) and to provide object background classification map (binary image) for the given video frames. Moment functions of the binary image (classification map) estimate the scale and orientation of the object. One heuristic approach is proposed to enhance the scale estimation using binary moments. Now the problem of object tracking is formulated as a constrained optimization one with respect to location, scale and orientation of the target.

The appearance model needs to be updated to account for the variation in object appearance. Appearance model updating for every upcoming frame is not required as the frames are temporarily coherent [3]. In this work, we propose a heuristic approach based on support value (provided by DCNN) to update the appearance model. Such an update mechanism can reduce the drift as well as reduce the computational cost (due to the retraining of the networks) for tracking. Another heuristic based on relative confidence score, is proposed to detect full occlusion during tracking. Effectiveness of the proposed algorithm is tested on various challenging video sequences containing large variation both in scale and orientation, illumination variation, pose change, fast motion, occlusion, shape deformation, transparency, specularity, low contrast, etc. Tracking results obtained using the proposed method are compared with those of twelve state-of-the-art techniques. To evaluate the performance of the proposed algorithm, two existing evaluation measures: average center location error and average tracking accuracy are considered in this work. Both quantitative and qualitative analysis of results highlight that the proposed method achieves state-of-the-art performance under various complex environments.

Organization of the rest of the article is as follows. Related work is briefly discussed in Section 2. Section 3 presents the proposed algorithm for tracking scaled and oriented object. Results obtained by various methods are analyzed in Section 4. Section 5 includes discussion and future work. Finally conclusive remarks are put in Section 6.

2. Related work

Object tracking is an important topic in computer vision and has been an active research area for several decades. There is an extensive literature on visual object tracking. Here, tracking using generative and discriminative appearance models are reviewed. Moreover, tracking methods based on deep learning, segmentation and particle filter are also available in this section. For more information about the literature on visual tracking, the reader may refer to [1,3,27].

2.1. Tracking based on generative models

Here, an object in the target frame is described by constructing generative appearance model and then search the required target location in the target candidate frame having most similar appearance to the constructed generative model. Among several existing methods in the literature [3], ASLA [8], MTT [9], LSHT [10] and

SCM [11] are more popular due to their better performances under various complex conditions.

ASLA [8], MTT [9] and SCM [11] are based on generative appearance model. Sparse representation is considered to represent the target in all these approaches. Such representations are robust under partial occlusion, pose change of the object. All these methods provide good results under such complex environments. Here, it is also noted that the local sparse representation (considered in ASLA [8] and SCM [11]) is more effective than the global sparse representation (used in MTT [9]). In all these above mentioned techniques, particle filter with affine motion model is considered to locate the object in the subsequent frames. ASLA [8] and SCM [11] can handle scale change of the target during tracking using affine motion model. However, they are unable to estimate the exact scale of the object due to affine motion model. Along with, for very fast moving object, all these methods produce unsatisfactory results due to consideration of poor dynamic models. On the other hand, He et al. [10] proposed a generative model using locality sensitive histogram (LSHT) to take into account for the contributions of every pixel in a frame instead of pixels only inside the local neighborhoods. Sometimes, it may provide good results for sequences having illumination variation, partial occlusion and pose change.

2.2. Tracking based on discriminative models

Visual object tracking is considered as a binary classification problem in these approaches. It finds the target location that can maximize the separability between the object and background [3, 27]. Ensemble tracking [12], MIL [13], Struck [14], TLD [15], KCF [16], DSST [17], TGPR [18] have been developed to track object under various complex environments. All these approaches constract a discriminative appearance model as a base classifier to distinguish the object from its surroundings.

The ensemble tracker [12] poses object tracking as a pixel level two class classification problem. It can handle changes in the object appearance by adding new weak classifiers to the ensemble system. In this direction, various techniques [13,15] have been developed to track objects.

Discriminative appearance models need to be updated to cope with the changes in the object appearance. To update the model, newly generated training set (after tracking the target in the target candidate frame) is not always useful (with respect to the object and the background classes). Therefore, model update with wrongly generated training set reduces the tracking accuracy and hence the tracker may sometimes enter into drift. By avoiding adhoc update strategies, Hare et al. [14] proposed a structure output tracking algorithm with kernels (named as Struck) to reduce drift problem. In this direction, Zhang et al. [28] proposed a multi-expert tracking framework (MEEN) where the base tracker can move backward to correct undesirable effects of bad model update using an entropy regularized restoration scheme. It can able to recover tracker from the drift.

Furthermore, based on fast Fourier transform, Henriques et al. [16] proposed a kernelized correlation filter (KCF) to distinguish the object from its surroundings. It is the fastest among all other existing tracking algorithms with respect to performance. In [17], Danelljan et al. proposed a novel approach for robust scale estimation in a tracking-by-detection framework. The classifier learns discriminative correlation filters based on a scale pyramid representation. It learns separate filters for translation and scale estimation. It provides acceptable results for scaled objects. However, it fails to estimate the exact scale of the target when object changes its scale and orientation jointly.

Robust visual target representation increases the tracking accuracy in several complex scenario. In this regard, Gao et al. [18] proposed a new transfer learning based tracking algorithm with Gaussian process regression (GPR). Since GPR is exploited to make a new objective of the observation model, drift problem is reduced. Experimentally it is shown that it generates better results than many state-of-the-art techniques.

To take advantages of both discriminative and generative appearance models, Cheng et al. proposed a multi-task learning framework for tracking object in [7]. The advantage of this tracking algorithm is that the generative tracker takes the local information of object into account for handling partial occlusions; while the discriminative tracker considers the overall information of object to represent the object appearance.

Features play an important role in visual tracking. Therefore, it is necessary to select more informative features from a set of features for robust tracking. In this direction, Song [29] proposed a visual tracking (RVT-IFS) algorithm which selects most informative features from a set of compressive features online. It takes into account both appearance and spatial layout information. This method can track objects in several complex environments.

2.3. Tracking based on deep learning

Performance of a tracking system depends on visual representation of the object. In this regard, handcrafted features have been considered in the literature. In the recent years, advancement of deep learning architectures is widely considered on visual recognition problems [30–35]. Deep learning has the ability to extract useful features from the image frames. To get such an advantage, Wang and Yeung [30] proposed a deep learning tracker (named as DLT) using a stacked denoising auto-encoder. One of the major advantages of this work is that it does not require any labeled patterns, however it needs large number of images to train the network in an unsupervised manner. Chen et al. proposed a tracking algorithm named as CNNTracker using deep learning concept [31]. In this approach, the authors used deep learning to learn the most discriminative features dynamically and transferred these learned features to task of object tracking. In this approach, the drift problem is handled by updating the model using both the information of ground truth of initial frame and object observation online. On the other hand, Ma et al. [32] proposed a visual tracking algorithm, named as HCF-VT. Here, deep convolutional neural network (CNN) is considered to extract robust feature. Since, this model is trained by object recognition dataset, it improves the tracking performance under various complex conditions. This method is more robust than DLT.

In this direction, Wang et al. [33] proposed a tracking algorithm, named as VT-FCNN using fully convolutional networks. The model is pre-trained on image classification dataset. Features extracted from different convolution layers are considered to represent target and to discriminate the target from its surroundings. Location of the target in the target candidate frame is estimated using Gaussian distribution. Though this method is robust to the target appearance, but it is unable to provide good results for sequences having scaled and oriented object, object with fast motion and shape deformable object. In [34], Wang et al. proposed a tracking algorithm using structured output convolutional neural network (named as SO-DLT) which transfers generic object features for online tracking. ImageNet detection dataset [36] is considered to pre-trained the model and then the model is fine tuned during the online tracking process. Furthermore, Kang et al. [35] proposed a tracking algorithm using combination of deep learning and swarm intelligence. Here, deep learning is considered to extract robust feature for detection and hybrid gravitational search algorithm for optimizing function in particle swarm optimization. This approach is able to track occluded object properly.

All these tracking algorithms based on deep learning architectures need a large amount of image classification dataset to train

the networks in an off-line manner. After that, the pre-trained network is fine-tuned in an online manner by the images of the problem in hand. However, in visual tracking, only one (or sometimes only a few) training frame is given. So training of a CNN using one training frame is next to impossible.

2.4. Segmentation based object tracking

The aim to segmentation is to divide a given image into several homogeneous regions. Homogeneity can be defined with respect to some features like intensity, color, texture, etc [37,38]. In these approaches, the object is divided into a number of patches and then tracking is done by establishing correspondence between patches of the target and the target candidate frames [1]. Several tracking algorithms based on image segmentation are proposed in the literature [39–41]. Since, the object is divided into a number of patches, it can handle some complex problems like partial occlusion, illumination variation, pose change, etc. In this regard, recently the researchers are concentrating more on superpixels. Ren and Malik first introduced the concept of superpixels in [42]. Recently, it is successfully applied in tracking [43–45]. Either high level or low level cues are considered in most of the tracking algorithms [43]. In contrast, Yang et al. [43] developed a tracking algorithm using superpixels (SPT) as mid-level information. Maximum a posterior probability as the object background confidence map indicates target location in the current frame. It can handle drift problem due to occlusion. However, this method is computationally more expensive. Moreover, Cai et al. [44] represented the target as a graph and formulated the tracking task as graph matching problem between a target graph and the target candidate graph. In this approach, each node of the graph is as super-pixel of the target. This approach can handle deformation and occlusion of the object during tracking.

Furthermore, in Locally Orderless Tracking (LOT) [45], the target is segmented into a number of superpixels. Each superpixel is represented by average (hue, saturation and value) HSV. The target state is sampled using a particle filter with a Gaussian weight around the previous position of the object. Due to the updating of appearance model, this technique can handle occlusion.

2.5. Object tracking using particle filter

Particle filter (PF) also known as condensation or sequential Monte Carlo model was introduced for visual tracking [46]. Over the last decades, it has became a very popular tracking framework due to its excellent performance in the presence of nonlinear target motion and flexibility to different object representations [47]. In general more particles are needed for better representation of the object. More likely, particle filter based tracking algorithms perform reliably in occluded, cluttered and noisy environments. However, the computational cost of particle filter based tracking algorithm is linearly dependent on the number of particles. This fact motivated researchers to develop some techniques to speed up the object tracking process in particle filter framework. In [48], objects are characterized using histograms of color and edge orientation. The observation likelihood is computed in a coarse-to-fine manner, which allows the computation to quickly focus on more promising regions. However, Khan et al. [49] proposed tracking framework using particle filter based on subspace representation. This method efficiently represents subspace in particle filter by applying Rao–Blackwellization to integrate the subspace coefficients in the state vector. Furthermore, Zhou et al. [50] proposed an object tracking algorithm in particle filter framework in which samples are adjusted according to an adaptive noise component. On the other hand, Brasnett et al. [51] proposed an object tracking technique using particle filter framework based on multiple cues (color, edge and texture) with adaptive parameters. They experimentally presented that tracking with multiple weighted cues provides more reliable performance than single cue tracking. In [5], Sardari and Moghaddam proposed an algorithm in particle filter framework for tracking occluded object. This approach is robust against changes in the object appearance model by employing Modified Galaxy based Search Algorithm (MGbSA) to reinforce for finding the optimum state in the particle filter state space. However, some generative [46,52–54], discriminative and segmentation [55–57] based trackers also considered particle filter to estimate object location in the current frame.

3. Proposed algorithm for tracking scaled and oriented object

This section presents the proposed scaled and oriented object tracking algorithm in detail.

3.1. Object detection using discriminative appearance model based on ensemble of multilayer perceptrons

Prior knowledge regarding the label of each pixel can be learned from a training (i.e., target[1]) frame, to construct an appearance model for both the target and the background. Let us consider *Mushiake*[2] video sequence. In this work, object (head) in the target frame is approximated by an ellipse. Fig. 1(a) shows the approximation of the object (with red colored ellipse) in the $(t-1)$th frame. Training set is generated with the positive samples (patterns) correspond to the pixels of the object region (inside the ellipse) and the negative samples corresponding to pixels of the background region (outside the ellipse but inside the dotted red colored rectangle which is double in size with respect to the semi-major and semi-minor axes of the ellipse). Here, we assume that the possible movement of the object is within that rectangle (called region of search) in the tth frame where the objects to be tracked [12]. The features namely 8-bin histogram of orientation gradients (HOG) [58] along with R, G and B corresponding to each pixel are extracted. Each labeled sample is represented with extracted 11-dimensional feature vector.

Test set (consisting of unlabeled samples) is generated from the tth frame. Pixels of the tth frame within the corresponding dotted red colored rectangle of the $(t-1)$th frame are considered as unlabeled samples. Fig. 1(b) shows the corresponding dotted rectangle in the tth frame. Extracted 11-dimensional feature (8-bin HOG along with R, G and B) vector represents each unlabeled sample.

In this work, a discriminative appearance model based on ensemble of MLPs is proposed to detect an object from its surrounding. Here, MLP with different architectures are trained using back-propagation algorithm to classify the unlabeled samples of the tth frame. After a number of epochs, when MLPs are stable (i.e., mean square error is optimal), the MLPs are able to classify the unlabeled samples. Each MLP makes decision for each unlabeled sample. Therefore, DCNN is trained with decisions (outputs) of MLPs for training samples using back-propagation algorithm to combine the decisions of MLPs. The stable DCNN is considered to combine the decisions of the MLPs for unlabeled samples. Let C be the number of classes. For our problem, $C = 2$, i.e., obj (object) and bg (background). Let the decision of the ith neuron at the output layer of DCNN be z_j^i for jth pattern $\mathbf{S}_j$, where $\mathbf{S}_j = \{\mathbf{S}_{j1}, \mathbf{S}_{j2}, \ldots, \mathbf{S}_{jT}\}$ is the set of outputs (decisions) of T numbers of MLPs and $\mathbf{S}_{jl} = \{S_{jl}^{obj}, S_{jl}^{bg}\}$ is the output of lth MLP. Here, $z_j^i \in \mathbb{R}$,

[1] In this article, training, target, $(t-1)$th and previous frame are used interchangeably.

[2] http://www.ee.surrey.ac.uk/CVSSP/xm2vtsdb/.

$\forall i, j$. We define $\bar{z}_j^i = \frac{\exp(z_j^i)}{\sum_i \exp(z_j^i)}$, where $i \in \{obj, bg\}$. Therefore, $\bar{z}_j^i \in (0, 1)$, $\forall i, j$ is called the support value [20] of the jth pattern to the ith class. Now, if $\bar{z}_j^{obj} > \bar{z}_j^{bg}$, then jth pattern is classified as object, otherwise background. Therefore, DCNN provides the object background classification map (also called binary image) for the tth frame. Fig. 1(c) shows the object background classification map of the tth frame of *Mushiake* video sequence.

3.1.1. Generation of confidence score

After detecting the object in the tth frame using the proposed ensemble of MLPs based discriminative appearance model, two confidence scores C_t^{obj} and C_t^{bg} for the object and the background regions, respectively, are calculated as $C_t^{obj} = \frac{U_t^{obj}}{U_t^{obj}+U_t^{bg}}$ and $C_t^{bg} = \frac{U_t^{bg}}{U_t^{obj}+U_t^{bg}}$, where U_t^{obj} and U_t^{bg} are the cardinality of pixels (test patterns) classified as the object and the background, respectively, in the tth frame. Here, $0 \leq C_t^{obj}, C_t^{bg} \leq 1$. These scores are considered to deal with occlusion problem (Section 3.4).

3.2. Joint estimation of orientation and enhanced scale

Let X be a random variable, which can take values from the set $\{(x, y)|0 \leq x \leq P-1; 0 \leq y \leq Q-1\}$ and $P \times Q$ is dimension of the image frame. Probability mass function of variable X is defined as

$$f(X) = f(x, y) = \begin{cases} 0 & if \quad (x, y) \in bg, \\ 1 & if \quad (x, y) \in obj. \end{cases}$$

Therefore, (m, n)th order binary raw moment is defined as $M^{m,n} = \sum x^n y^n f(x, y)$. Let $I_t = [f(x_i, y_j)]; f(x_i, y_j) = 0$ or $1; 0 \leq x_i \leq P-1; 0 \leq y_j \leq Q-1$ be a binary image (classification map corresponding to the tth frame using DCNN model) of size $P \times Q$. Considering pixel value as a point mass, I_t is considered to be composed of a set of point masses located at (x_i, y_j). Using moments [59] of the binary image frame, centroid of I_t is a point $(\bar{x}_t, \bar{y}_t)$ defined as

$$\bar{x}_t = \frac{M_t^{10}}{M_t^{00}}; \ \bar{y}_t = \frac{M_t^{01}}{M_t^{00}}. \tag{1}$$

If μ_t^{20}, μ_t^{02} and μ_t^{11} are the second order moments of the (binary) image and λ_t^1, λ_t^2 refer to its principal moment of inertia, then

$$\lambda_t^1 = \frac{(\mu_t^{20} + \mu_t^{02}) + [(\mu_t^{20} - \mu_t^{02})^2 + 4(\mu_t^{11})^2]^{\frac{1}{2}}}{2};$$
$$\lambda_t^2 = \frac{(\mu_t^{20} + \mu_t^{02}) - [(\mu_t^{20} - \mu_t^{02})^2 + 4(\mu_t^{11})^2]^{\frac{1}{2}}}{2}. \tag{2}$$

The orientation angle θ_t^* of one of the principal axes of inertia with the x-axis is given by the equation as $\mu_t^{11} \tan^2 \theta_t^* + (\mu_t^{20} - \mu_t^{02}) \tan \theta_t^* - \mu_t^{11} = 0$. Hence,

$$\theta_t^* = \frac{1}{2} \arctan\left(\frac{2\mu_t^{11}}{\mu_t^{20} - \mu_t^{02}}\right). \tag{3}$$

Eqs. (2) and (3) can be used to define an image ellipse (Fig. 2(a)) which has the same moments of inertia and principal axes direction as the original image frame. The lengths a_t^* and b_t^* of the semi-major and the semi-minor axes, respectively, of the image ellipse are given by

$$a_t^* = 2\left(\frac{\lambda_t^1}{\mu_t^{00}}\right)^{\frac{1}{2}} \text{ and } b_t^* = 2\left(\frac{\lambda_t^2}{\mu_t^{00}}\right)^{\frac{1}{2}}. \tag{4}$$

3.2.1. Enhancement of scale estimation

Let A_1B_1 and C_1D_1 be the major and the minor axes of the estimated (using binary moment functions as discussed above) ellipse that can approximate the object shape in the tth frame. We have $\overline{A_1B_1} = 2a_t^*$ and $\overline{C_1D_1} = 2b_t^*$. Let P_1, Q_1, T_1, and S_1 are four intersection points of object boundary and possible rectangle outside the object region (i.e., classification map I_t) in the tth frame (Fig. 2(b)). Now straight lines A_1B_1 and C_1D_1 are extended to A_2B_2 and C_2D_2, respectively. From the points Q_1 and S_1, two perpendiculars Q_1Q_2 and S_1S_2 are drawn respectively on the extended line C_2D_2. Similarly, two perpendiculars P_1P_2 and T_1T_2 are also drawn, respectively from the points P_1 and T_1 on the extended line A_2B_2. Let $\overline{Q_1Q_2} = a_t'$, $\overline{S_1S_2} = a_t''$, $\overline{P_1P_2} = b_t'$ and $\overline{T_1T_2} = b_t''$. We define

$$a_t^{max} = \max\{a_t^*, a_t', a_t''\}; \ a_t^{min} = \min\{a_t^*, a_t', a_t''\} \tag{5}$$

and

$$b_t^{max} = \max\{b_t^*, b_t', b_t''\}; \ b_t^{min} = \min\{b_t^*, b_t', b_t''\}. \tag{6}$$

Basically, estimation of a_t^* and b_t^* depends on classification map of the tth frame using DCNN. It may not always be possible that the estimated a_t^* and b_t^* are optimal ones. On the other hand, estimation of a_t^{min}, a_t^{max} and b_t^{min}, b_t^{max} may help to estimate the optimal length of semi-major and semi-minor axes of the ellipse for properly tracking scaled object in the tth frame. Therefore, estimation of a_t^{min}, a_t^{max} (Eq. (5)) and b_t^{min}, b_t^{max} (Eq. (6)) may enhance the estimation of a_t^* and b_t^* (Eq. (4)).

3.3. Constrained object tracking

Here, the problem of object tracking is formulated as a constrained optimization one with respect to the following parameters: center location of the target (x_t, y_t), scale of the target (a_t, b_t) and orientation of the target (θ_t). Therefore, the target state X_t in the tth frame is obtained as

$$\hat{X}_t \equiv (\hat{x}_t, \hat{y}_t, \hat{a}_t, \hat{b}_t, \hat{\theta}_t) = \arg\max_{X_t^l} \rho(p(X_{t-1}), q(X_t^l)), \tag{7}$$

where, ρ is the Bhattacharyya coefficient [60] between distribution $p(X_{t-1})$ of the target model in the $(t-1)$th frame and distribution $q(X_t^l)$ of the lth target candidate model in the tth frame. X_t is the target state at the tth frame, defined as $X_t \equiv (X_t^c, X_t^s, X_t^\theta) \equiv (x_t, y_t, a_t, b_t, \theta_t)$, where $X_t^c \equiv (x_t, y_t)$ represents the center location of the target, $X_t^s \equiv (a_t, b_t)$ denotes the scale of the target and $X_t^\theta \equiv (\theta_t)$ represents the orientation of the target along the positive direction of the x-axis, and $X_t^1, X_t^2, \ldots, X_t^l, \ldots, X_t^N$ are N different (possible candidate targets) templates obtained by varying parameters as

$$z_t \in [(u_t - k_i * \sigma(z_{t-1})), (v_t + k_i * \sigma(z_{t-1}))], \tag{8}$$

with $z_t \in \{x_t, y_t, a_t, b_t, \theta_t\}$, $u_t \in \{\bar{x}_t, \bar{y}_t, a_t^{min}, b_t^{min}, \theta_t^*\}$, $v_t \in \{\bar{x}_t, \bar{y}_t, a_t^{max}, b_t^{max}, \theta_t^*\}$ and $k_i \in \mathbb{R}$; for $i = 1, 2, 3, 4, 5$. $(\bar{x}_t, \bar{y}_t)$ is computed as the centroid of the detected object using Eq. (1). a_t^{max} and a_t^{min} are respectively the maximum and minimum length of semi-major axis (Eq. (5)). Similarly b_t^{max} and b_t^{min} are the maximum and minimum lengths of semi-minor axis, respectively (Eq. (6)), and θ_t^* is the orientation angle of the detected object (Eq. (3)). $(\sigma^2(x_{t-1}), \sigma^2(y_{t-1}))$, $(\sigma^2(a_{t-1}), \sigma^2(b_{t-1}))$ and $\sigma^2(\theta_{t-1})$ are the variance of position, scale and orientation of the detected object up to $(t-1)$th frame. The Bhattacharyya coefficient in this case can be defined as

$$\rho\left(p\left(X_{t-1}\right), q\left(X_t^l\right)\right) = \sum_{u=1}^{h} \sqrt{p_u\left(X_{t-1}\right) q_u\left(X_t^l\right)}. \tag{9}$$

Here, the normalized h-bin color histogram is considered as distribution of the target.

Fig. 1. (a) Object represented by ellipse in the target frame, (b) Corresponding dotted rectangle in the target candidate frame, (c) Detected target by DCNN in the target candidate frame.

Table 1
Architectures of considered MLP networks.

Used networks	Hidden layers	Nodes in 1st hidden layer	Nodes in 2nd hidden layer
MLP1	2	9	6
MLP2	2	7	4
MLP3	2	4	3
MLP4	1	8	0
MLP5	1	5	0
DCNN	0	0	0

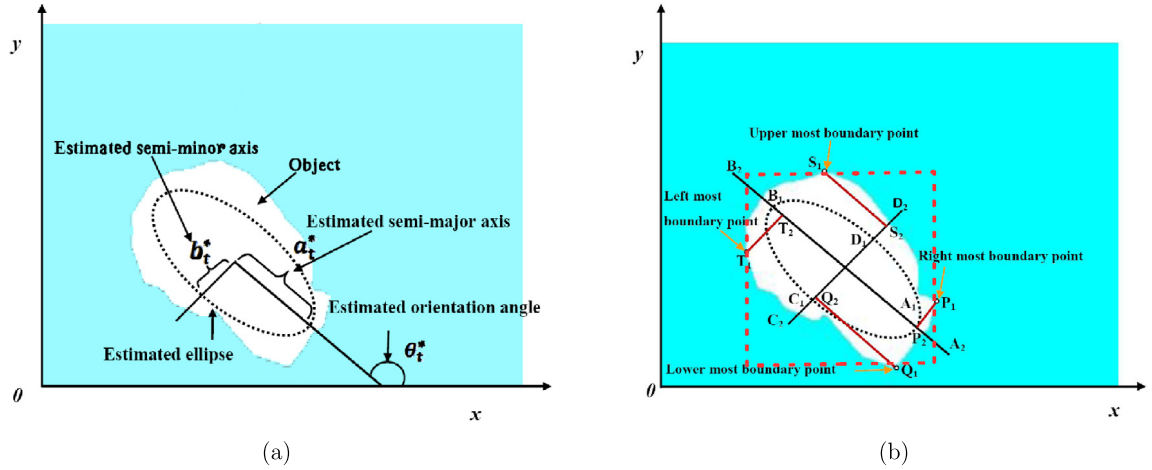


Fig. 2. (a) Estimation of scale and orientation angle and (b) Enhancement of scale estimation.

An ellipse with center $(\hat{x}_t, \hat{y}_t)$, semi-major axis $\hat{a}_t$, semi-minor axis $\hat{b}_t$ and orientation angle $\hat{\theta}_t$ can approximate the target in the tth frame. Hence, the required object is tracked with that ellipse in the tth frame and Fig. 3 displays the tracking result for head of *Mushiake* video sequence. After tracking the object in the tth frame, the mean (μ) position, scale and orientation of the target with new information (position, scale and orientation) of the tth frame is updated as

$$\mu(z_t) = (1 - \gamma_i) * \mu(z_{t-1}) + \gamma_i * \hat{z}_t, \tag{10}$$

where $z_t \in \{x_t, y_t, a_t, b_t, \theta_t\}, \hat{z}_t \in \{\hat{x}_t, \hat{y}_t, \hat{a}_t, \hat{b}_t, \hat{\theta}_t\}$ and $\gamma_i \in [0, 1]$; for $i = 1, 2, 3, 4, 5$. Here, $(\mu(x_{t-1}),\ \mu(y_{t-1}))$, $(\mu(a_{t-1}),\ \mu(b_{t-1}))$ and $\mu(\theta_{t-1})$ are mean position, scale and orientation of the target up to $(t-1)$th frame.

Similarly, variance (σ^2) is also updated as

$$\sigma^2(z_t) = (1 - \gamma_i) * \sigma^2(z_{t-1}) + \gamma_i * (\mu(z_t) - \hat{z}_t)^2, \tag{11}$$

where $z_t \in \{x_t, y_t, a_t, b_t, \theta_t\}, \hat{z}_t \in \{\hat{x}_t, \hat{y}_t, \hat{a}_t, \hat{b}_t, \hat{\theta}_t\}$ and $\gamma_i \in [0, 1]$; for $i = 1, 2, 3, 4, 5$.

During full occlusion, position (Eq. (1)), scale (Eqs. (4), (5) and (6)) and orientation (Eq. (3)) of the target are not explicitly estimated (Section 3.2). The updated means and variances of position, scale and orientation of the target are considered to track the object during full occlusion. Next section describes the object tracking in full occlusion scenario.

3.4. Online update to handle drift and occlusion

The proposed discriminative appearance model needs to be updated to adapt changes in the object appearance due to various factors. As in a video, the frames are temporarily coherent to each other [1]; so no need to update the appearance model for every upcoming frame. In this work, we provide a simple but effective approach to update the appearance model. Let $A = \{\mathbf{S}_j : \bar{z}_j^{obj} > \alpha\}$, $B = \{\mathbf{S}_l : \bar{z}_l^{bg} > \alpha\}$ and $D = A \cup B$, where $\bar{z}_j^{obj}$ and $\bar{z}_l^{bg}$ are the support values (Section 3.1) of the jth and the lth test patterns to be classified and α $(\in [0, 1])$ is a user defined constant. If $|D| < \beta$, where $|.|$ represents the cardinality of a set and $\beta \in \mathbb{N}$ is a user defined constant, then update the discriminative appearance model. Here, the appearance model is updated using retraining (same as the training process described in Section 3.1) with newly generated training set from the tth frame and we maintain a list which contains only the current frame.

In this work, we present a simple but efficient method to detect full occlusion during object tracking. For a state X_t of the target at time t, the confidence scores C_t^{obj} and C_t^{bg} (defined in Section

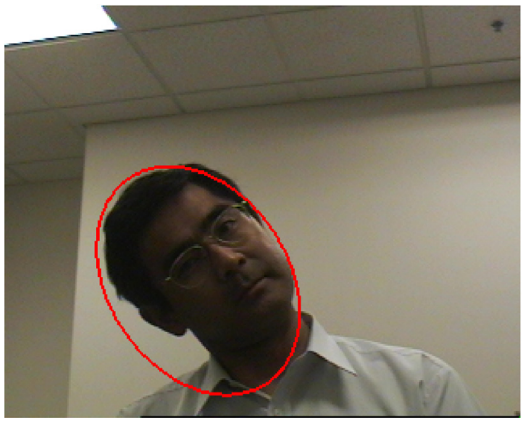


Fig. 3. Tracking result of *Mushiake* video sequence.

3.1.1) for the object and background regions are bounded within the range [0, 1]. The higher value of C_t^{obj} indicates that all pixels in the tth frame belong to the object. The lower bound of C_t^{obj} indicates that all pixels in the tth frame belong to the background. A threshold value Ω ($\in$ [0, 1]) is set to detect full or heavy occlusion: $RCS = \frac{C_t^{obj}}{\mu(C_{t-1}^{obj})+\epsilon} < \Omega$, where ϵ is a small non zero real value (used to avoid the division by 0 issue) and $\mu(C_{t-1}^{obj})$ is the average object confidence score up to $(t-1)$th frame. It is calculated as $\mu(C_{t-1}^{obj}) = \frac{1}{(t-1)} \sum_{l=1}^{t-1} C_l^{obj}$.

Relative confidence score (RCS) of the object (detected by DCNN) less than Ω means the detected object still belongs to the background with high probability and a heavy occlusion is deemed to occur. In such a situation, scale and orientation of the detected object is not estimated as described in Section 3.2, but the target (with respect to position, scale and orientation) in the tth frame is found using Eq. (7), where $X_t^1, X_t^2, \ldots, X_t^l, \ldots, X_t^N$ are N different (possible candidate targets) templates with different parameters:

$$z_t \in [(\mu(z_{t-1}) - k_i * \sigma(z_{t-1})), (\mu(z_{t-1}) + k_i * \sigma(z_{t-1}))], \qquad (12)$$

where $z_t \in \{x_t, y_t, a_t, b_t, \theta_t\}$ and $k_i \in \mathbb{R}$; for $i = 1, 2, 3, 4, 5$. After tracking the occluded object in the tth frame, update the mean (μ) and variance (σ^2) of position, scale and orientation of the target (using Eqs. (10) and (11)), respectively. In case of full occlusion, the considered discriminative appearance model is not updated, only the list is updated by the new one. Fig. 4 displays the flowchart of the proposed tracking algorithm.

4. Experimental results and analysis

4.1. Datasets

The performance of the proposed algorithm is evaluated on various challenging video sequences downloaded from Visual Tracker Benchmark,[3] ALOV++,[4] VOT2015 Challenge.[5] These sequences include most challenging factors in visual tracking like object with large variation both in its scale and orientation, occlusion, background clutter, illumination change, object deformation, etc.

[3] http://cvlab.hanyang.ac.kr/tracker_benchmark/index.html.
[4] http://crcv.ucf.edu/data/ALOV++/.
[5] http://www.votchallenge.net/vot2015/dataset.html.

4.2. Baselines

DLT[6] [30], DSST[7] [17], TLD[8] [15], ASLA[9] [8], TGPR[10] [18], SCM[11] [11], KCF[12] [16], MEEN[13] [28], LOT[14] [45], LSHT[15] [10], MTT[16] [9], SPT[17] [43] are considered as baseline algorithms. Source codes provided by the authors are considered to generate comparative results. Same initialization and parameter as mentioned in the corresponding article are considered to produce baseline results. The proposed method also considered same parameters for all sequences in our experiments.

4.3. Implementation details

We utilized 8-bin HOG along with red, green and blue components of RGB color space as features to represent each pixel of the frame. The given image frame is divided into small overlapping spatial regions of size 5 × 5 pixels, called cells. For each cell, 8-bin HOG, for [0°, 180°] interval, is computed by accumulating weighted votes based on gradient magnitude into bins for each orientation. The center pixel of the cell is represented by 8-bin HOG. It is able to capture shape and appearance of the object. It is also invariant to translation, rotation and global illumination variation [58].

To collect a training set in the initialization step, the target region in the 1st frame is located by manually cropping. Here, the parameters k_i, for $i = 1, 2, 3, 4, 5$ are set to 1. The parameters γ_i in Eqs. (10) and (11), for $i = 1, 2, 3, 4, 5$ are set to 0.5. The threshold Ω, to detect full occlusion, is considered as 0.3. The parameters α is set to 0.75 and β is taken as integer part of 70 percent of training set. For the proposed approach, 10 percent of training set was considered to train the network and it was sufficient to train the T (set to 5) number of considered MLPs. Table 1 provides the architectures of the considered MLPs (with 11 nodes at the input layer and 2 nodes at the output layer) and DCNN (with 10 nodes at the input layer and 2 nodes at the output layer). Weights of each network are initialized randomly within [−1.0, 1.0]. Learning rate is set to 0.001 for all networks. All these parameters are selected based on experiments. During experiments, we selected parameters from a set of parameters which provided best object tracking accuracy for 10 sequences (more complex). We considered these values of parameters to produce results for all considered video sequences.

All simulations have been done on a machine with Intel(R) Core(TM)i3 CPU 3.20 GHz, 4.0 GB RAM and Fedora operating system.

Here, the proposed method provides elliptical tracker whereas all existing trackers are rectangular. Therefore, a rectangular tracker is constructed with the information of the proposed elliptical tracker to make a fair comparison between them.

4.4. Evaluation measures

We considered quantitative measures such as average center location error and average tracking accuracy (ATA) [61] to evaluate the performance of the proposed tracking algorithm.

[6] http://winsty.net/dlt.html.
[7] http://www.cvl.isy.liu.se/en/research/objrec/visualtracking/scalvistrack/index.html.
[8] http://github.com/zk00006/OpenTLD.
[9] http://faculty.ucmerced.edu/mhyang/project/cvpr12_jia_project.htm.
[10] http://www.dabi.temple.edu/~hbling/code/TGPR.htm.
[11] http://faculty.ucmerced.edu/mhyang/project/cvpr12_scm.htm.
[12] http://home.isr.uc.pt/~henriques/circulant/.
[13] http://http://cs-people.bu.edu/jmzhang/MEEM/MEEM.html.
[14] http://www.eng.tau.ac.il/~oron/LOT/LOT.html.
[15] http://www.cs.cityu.edu.hk/~shengfehe2/visual-tracking-via-locality-sensitive-histograms.html.
[16] http://sites.google.com/site/zhangtianzhu2012/publications.
[17] http://www.umiacs.umd.edu/~fyang/spt.html.

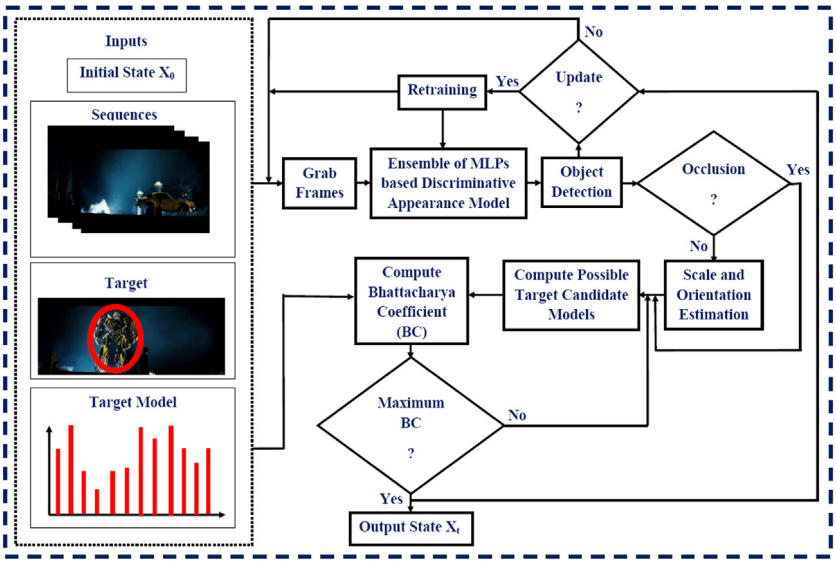


Fig. 4. Flowchart of the proposed algorithm.

Table 2
Average Tracking Accuracy (area overlap) of all the considered sequences. Red: indicates best value, pink: indicates second best, green: indicates third best.

Sequences	DLT	DSST	TLD	ASLA	TGPR	SCM	KCF
200	0.709	0.733	0.659	0.617	0.570	0.613	0.582
	MEEN	LOT	LSHT	MTT	SPT	PM	
	0.503	0.612	0.492	0.491	0.562	0.769	

Table 3
Average center location error of all the considered sequences. Red: indicates best value, pink: indicates second best, green: indicates third best.

Sequences	DLT	DSST	TLD	ASLA	TGPR	SCM	KCF
200	15.60	13.64	32.08	25.07	50.69	40.53	36.84
	MEEN	LOT	LSHT	MTT	SPT	PM	
	21.13	28.65	36.72	30.49	29.42	10.17	

Table 4
Average execution time (seconds) needed for tracking objects in a frame using different methods. Red: indicates best value, pink: indicates second best, green: indicates third best.

Sequences	DLT	DSST	TLD	ASLA	TGPR	SCM	KCF
200	4.166	0.404	0.036	0.445	4.885	4.144	0.005
	MEEN	LOT	LSHT	MTT	SPT	PM	
	0.344	1.429	0.065	1.043	2.574	0.420	

4.5. Overall performance evaluation

4.5.1. Quantitative evaluation

Average area overlap and average center location error over all the considered sequences are summarized in Tables 2 and 3, respectively. Higher value of average tracking accuracy indicates the better tracking performance. From Table 2, it is seen that the proposed method attains highest value for average tracking accuracy among all the considered techniques. The table also highlights that DSST and DLT are second and third highest among all these techniques. DSST and DLT both considered deep learning to detect object, however, they not considered any algorithm to explicitly estimate the scale and orientation of the object. Due to this reason, they are unable to provide best results for scaled and oriented object. On the other hand, the proposed method explicitly estimates the scale and orientation of the detected object, therefore it obtained best results.

On the other hand, the lower value of the average center location error indicates better tracking performance. Table 3 highlights that the proposed method obtains lower value of average center location error. Similarly, DSST and DST attain second and third lowest values of average center location error. Both these tables highlight that the proposed method obtained better tracking results due to adaptation of scale and orientation of the object than the considered existing techniques.

4.5.2. Qualitative evaluation

In this section, tracking results obtained using different techniques are analyzed based on their visual qualities. Due to page constraint, we only discuss the performances of the algorithms only on a few sequences and similar findings are obtained for other sequences also.

The Bag sequence contains abrupt motion with large scale and orientation changes. Tracking results of the frames 31, 55 and 195 are displayed in Fig. 5(a). All the existing techniques are unable to provide optimal trackers due to large variation both in scale and orientation of the object. On the other hand, the proposed method jointly estimates the scale and orientation of the object during tracking, therefore it is able to obtain optimal tracker for such a sequence.

Results of the Book sequence are displayed in Fig. 5(b). Most of the trackers enter into drift when object changes its scale and orientation together, whereas the proposed method can handle changes both in scale and orientation of the object, very well. Compared with the existing techniques, the proposed method shows much better performance establishing the superiority of it.

In Octopus sequence, the Octopus changes its shape during movement. Tracking results in few frames of this sequence are given in Fig. 5(d). All the existing trackers properly track the Octopus over a number of frames. After frame 170, most of the existing trackers enter into drift due to abrupt shape deformation. Whereas, the proposed method is able to track the Octopus throughout the sequence as shown in Fig. 5(d).

Various factors like occlusion, scale change and fast motion create difficulty to track Woman. Fig. 6(a) displays tracking results of Woman sequence. LSHT, MTT, ASLA and TLD approaches fail to track Woman after few number of frames due to occlusion. On the other hand, SPT, LOT, DLT, MEEN, DSST, TGPR algorithms track the Woman successfully up to 560 frames. After that all these techniques enter into drift due to large scale change (see

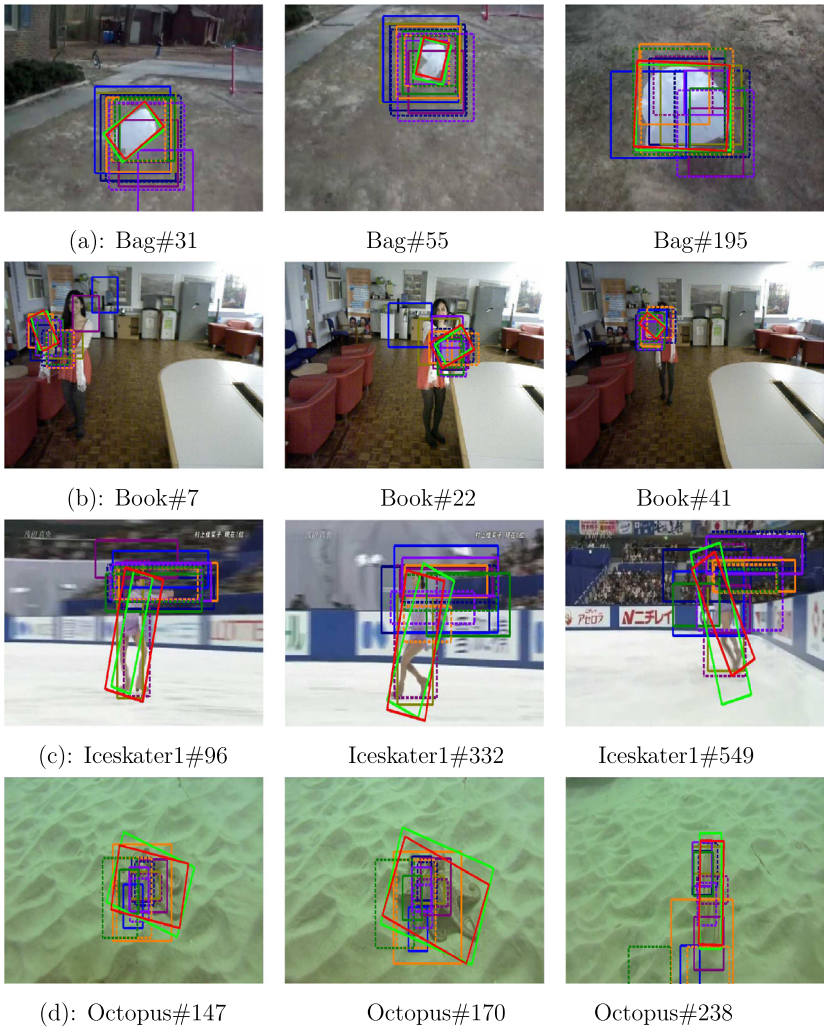


Fig. 5. Tracking results of 13 trackers on 4 sequences of VOT2015.

frame number 573). Only the proposed method is able to track the Woman properly throughout the sequence.

It is very difficult to track player properly in the Basketball sequence due to abrupt motion, illumination variation, occlusion and more similar objects. TLD, ASLA, LOT, TLD, LHST methods enter into drift for most of the frames (e.g., 284, 648 and 723 frames in Fig. 6(b)), due to the closer of several similar objects to the actual object. In such scenario, SPT, KCF, SCM, DLT, MEEN and TGPR techniques also enter into drift. In contract, the proposed algorithm is able to track object over the full sequence (see Fig. 6(b)).

In this experiment, Faceocc1 sequence is considered to evaluate the robustness of the proposed algorithm for tracking an occluded object. Fig. 6(c) displays the tracking results of 213, 703 and 873 frames. This figure highlights that most of the existing methods try to track the face accurately. Only ASLA and LOT trackers encounter drift due to occlusion. Whereas, SPT also encounters minor drift for few number of frames (e.g., 703, 741 and 873).

Due to fast motion, partial occlusion and large scale variation, Carscale sequence is considered for our experiments. Fig. 6(d) displays tracking results obtained using different methods. All trackers try to track the object properly. But after a few frames (e.g., 200), all state-of-the-art methods enter into drift due to large scale variation. In contrast, the proposed method tracks the Car properly during large scale variation (e.g., frame number 244 in Fig. 6(d)).

Similar kind of observations is found in the results presented in Fig. 7.

4.5.3. Critical analysis

To evaluate the performance of the proposed algorithm over the existing techniques, tracking results of two sequences using different methods are critically analyzed. Tracking results for few frames (e.g., 91, 191 and 246) of the Bag sequence are provided in Fig. 8. ASLA is able to track the Bag. Though affine motion model is considered in ASLA technique, but it is unable to provide optimal tracker due to abrupt changes in scale and orientation of the Bag (see Fig. 8(a)). KCF also tracks the Bag properly over a few frames. It provides better result than ALSA over some frames. When large variation in scale occurs, KCF is unable to adapt such large variation and unable to generate the optimal tracker (see frame number 246 in Fig. 8(b)). However, DLT and DSST (in Fig. 8(c) and (d), respectively) can better adapt the scale change of the Bag and provide better results than other existing techniques. On the other hand, MEEN, SCM, TGPR and TLD are unable to adapt the large scale variation in the object during tracking. The tracking results of these techniques individually compared with the proposed technique and ground truth tracker, are provided in Fig. 8(e), (f), (g) and (h), respectively. The visual results provided in Fig. 8 highlights that the proposed method can better estimate the scale and orientation of the object jointly and hence produces better tracking results than state-of-the-art-techniques.

In case of the Book sequence, the performances of the state-of-the-art-techniques are individually compared with the proposed method and visual results are displayed in Fig. 9. Fig. 9(a) shows

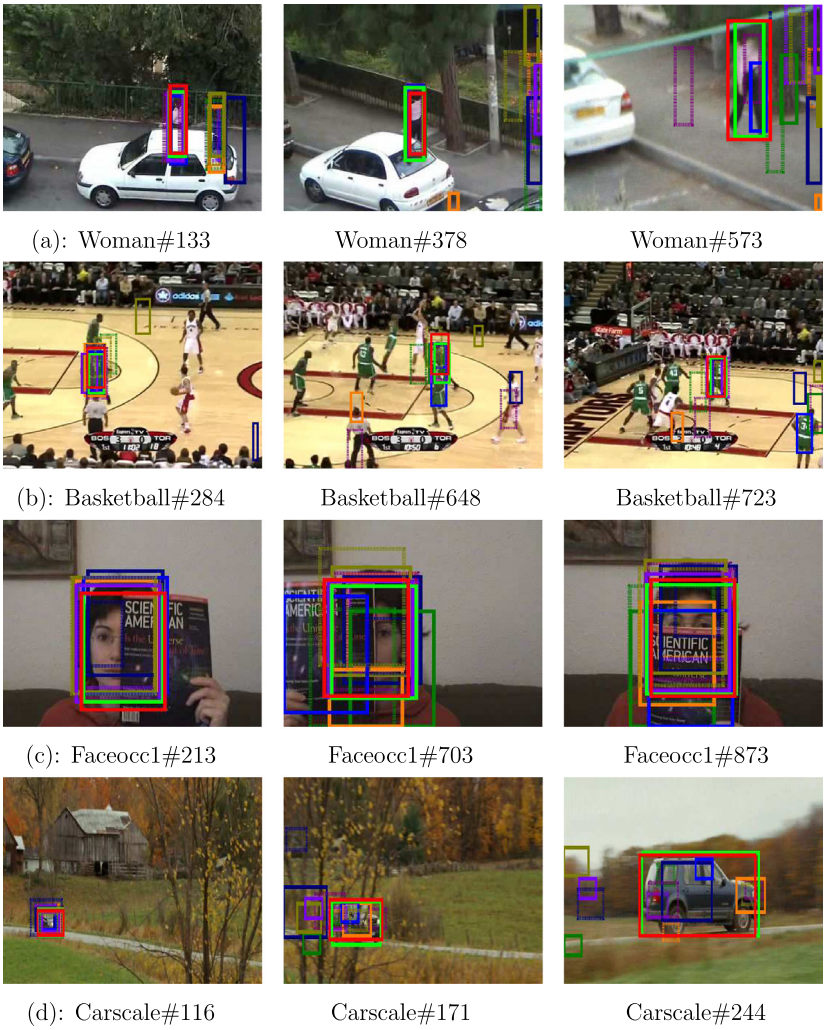


Fig. 6. Tracking results of 13 trackers on 4 sequences of Visual Tracker Benchmark.

that ASLA fails to properly track the Book in most of the frames. However, KCF can better track the Book than other existing methods for most of the frames. But for a few frames, where object changes its scale by a large amount, KCF enters into drift (see Fig. 9(b)). Moreover, DLT, DSST, MEEN, SCM, TGPR and TLD are all unable to properly track the Book and after few frames, they enter drift. Fig. 9(c), (d), (e), (f), (g) and (h), respectively show the failure of these techniques for the Book sequence. This figure highlights that the proposed method can properly track the Book throughout the sequence.

4.6. Execution time analysis

Average execution time required for tracking an object in a single frame using various methods are summarized in Table 4. This table highlights that KCF takes least amount of time whereas TGPR needs maximum amount of time among all techniques for tracking a single frame. From this table, it also observed that DSST, ASLA, MEEN and the proposed method take moderate amount of time for tracking the object in a single frame.

5. Discussion and future work

In the proposed approach, ensemble of MLP based discriminative appearance model is able to distinguish the object from its surrounding and, provides object background classification map (binary image). Then, orientation angle and scale change are jointly estimated from the detected object using second order moments of the binary classification map. As the estimations of scale and orientation in the proposed method rely on the binary classification map, rapid scale and orientation changes do not affect the accuracy of tracking algorithm. The proposed algorithm is robust to the background clutter and pose change. The proposed method can handle occlusion by searching the appropriate target among all possible candidate targets.

Though the proposed method provides better tracking results than state-of-the-art techniques however, DLT [30], HCF-VT [32], VT-FCNN [33] and SO-DLT [34] based on deep learning can better detect object than the ensemble of MLPs. But in case of these techniques, scale and orientation of the object is not explicitly estimated and they are unable to provide good results for tracking scaled and oriented object. On the other hand, various experiments illustrate that the proposed method provides better tracking results than DLT. One may consider the proposed scale and orientation estimation approach into DLT or some other techniques based on deep learning to obtain better tracking results for handling scale and oriented object in various complex environments.

5.1. Failure of the proposed method

The proposed method fails to track an object under more complex environments (like sequences with confusion, large shape

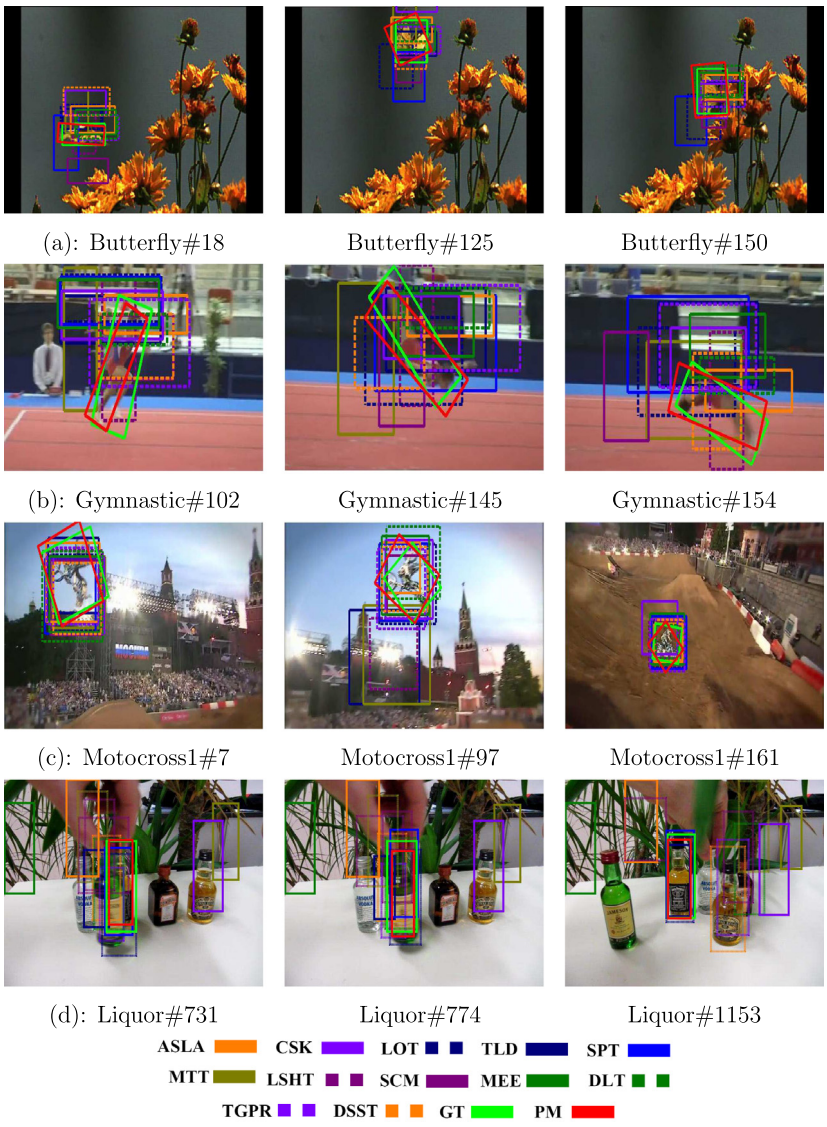


Fig. 7. Tracking results of 13 trackers on 4 sequences of ALOV++.

deformation, transparency, low contrast, specularity, long time full occlusion with large object motion). Fig. 10 shows the sequences where the proposed method enters into drift.

In the Confusion sequence, after some number of frames, the visual content of this frame is faded out and visual content of another sequence is put in. In this sequence, the proposed method tracks object properly for a few frames (see Fig. 10(a)). When frames of a new sequence are superimposed into the frames of original sequence, the proposed method fails to track the original object and enters into drift (see results of 110 and 118 frames). This is due to large change of appearance of the frame. The updated model is unable to properly detect object (with large appearance change) in the current frame. Due to erroneous detection of object, the proposed method is unable to estimate scale and orientation of the actual object, hence tracker is unable to properly track the object and enters into drift. In case of Shape sequence, the shape of the object is certainly changed and view of the camera is also certainly changed. The proposed method is unable to adapt certain scale and view change. Fig. 10(b) shows the tracking result using the proposed method. From this figure, it is seen that after frame number 90, scale of the object and view of the camera are certainly changed with a large amount. Due to this reason, the proposed method fails to adapt to such a large amount of scale and view change. In case of Transparency sequence, one scene is transparent to another scene. Due to change in visual content of the scene, the proposed method is unable to track the object after the frames where transparency occurs. Fig. 10(c) displays the tracking results of Transparency sequence. From this figure, it is seen that the content of frame number 237 is totally different from the content of frame number 205, due to difference in visual content, the proposed tracker enters into drift. Since, the visual content of the current frame is totally changed from the previous frame, the updated ensemble model with information of the previous frame is not sufficient to detect the object in the current frame, where transparency is occurred. As object is not detected in the current frame, the proposed method fails to estimate scale and orientation of the object. In such cases (i.e., failure of object detection), the proposed method considered the estimated scale and orientation of the object in the previous frame as tracker information in the current frame. Therefore, using tracker information of the previous frame, the proposed method is unable to track object in the current frame in case of transparency.

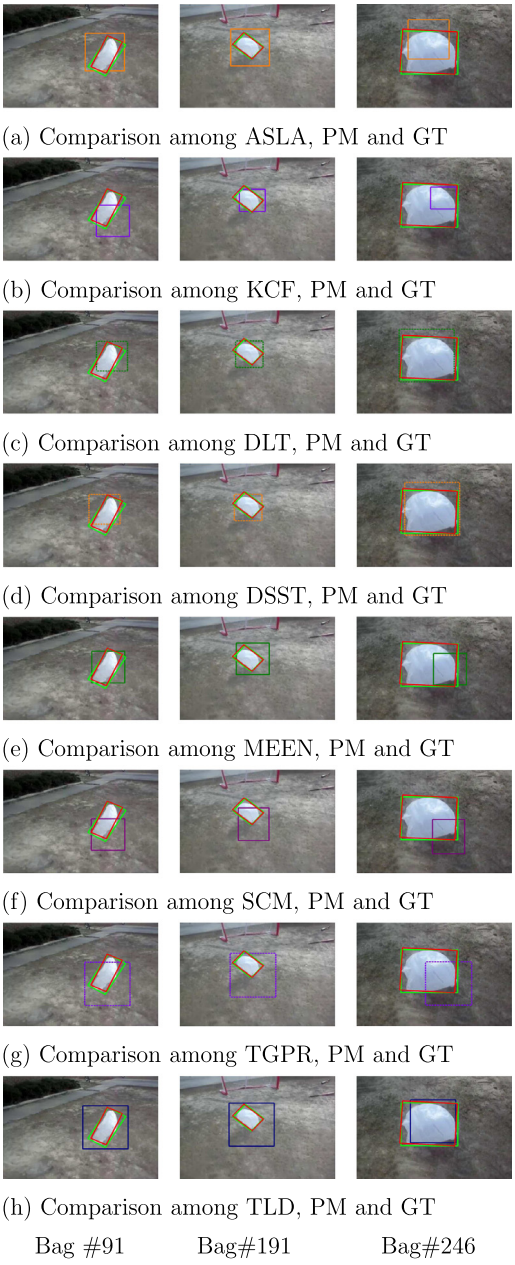


Fig. 8. Tracking results of 8 trackers on Bag sequence.

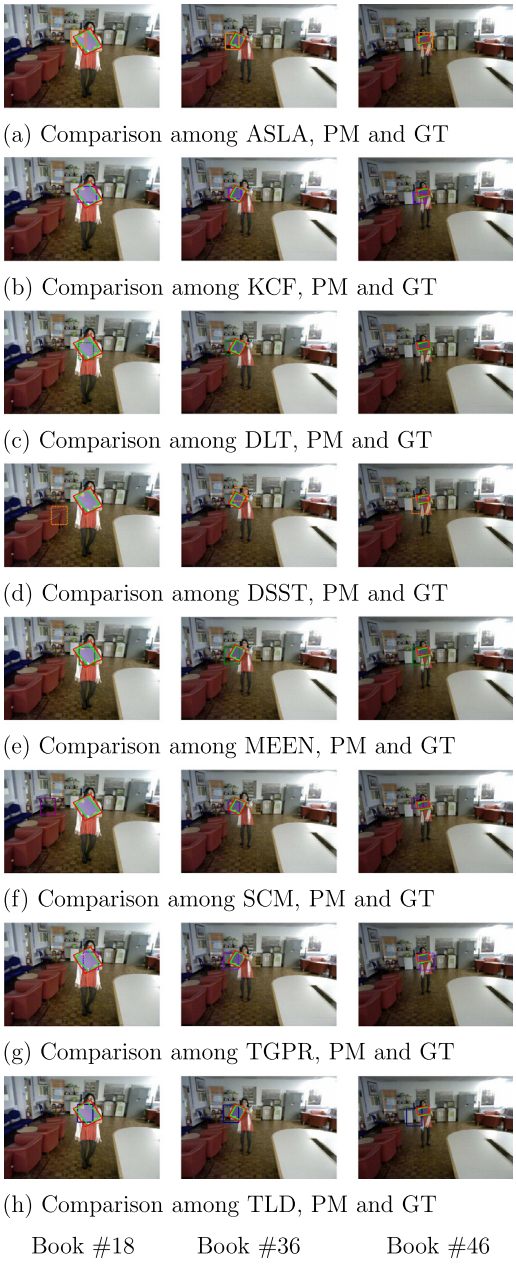


Fig. 9. Tracking results of 8 trackers on Book sequence.

5.2. Future work

In this article, we have considered pixel information to detect an object. For more complex sequences, pixel information is not sufficient to detect object. In future, patch or region based information may be taken into consideration to increase detection accuracy. In the present work, two threshold values are needed to update the appearance model and detect full occlusion. These threshold values are set experimentally. In future, we may provide some adaptive approaches to adjust these threshold values. In future, to make the proposed algorithm more robust against such complex sequences, we will consider tracklet concept and some data association rules. The tracklet concept will help us to analyze the already visited path by the moving object, whereas, data association will help us to establish correspondence among objects. Based on tracklet and data association rule, the location of the object in the current frame can be estimated.

6. Conclusions

In this article, an algorithm is proposed using ensemble of MLPs based discriminative appearance model and binary moments. The proposed technique effectively tracks an object with large variation both in scale and orientation and several other conditions like occlusion, drift, etc. Experimental results and evaluations demonstrate that the proposed method outperforms twelve state-of-the-art techniques for tracking objects in some complex environments.

Acknowledgments

The authors like to thank the editor and the reviewers for their thorough and constructive comments, which helped a lot to enhance the quality of the manuscript. Funding by U.S. Army, USA through the project “Processing and Analysis of Aircraft Images with Machine Learning Techniques for Locating Objects of Interest"

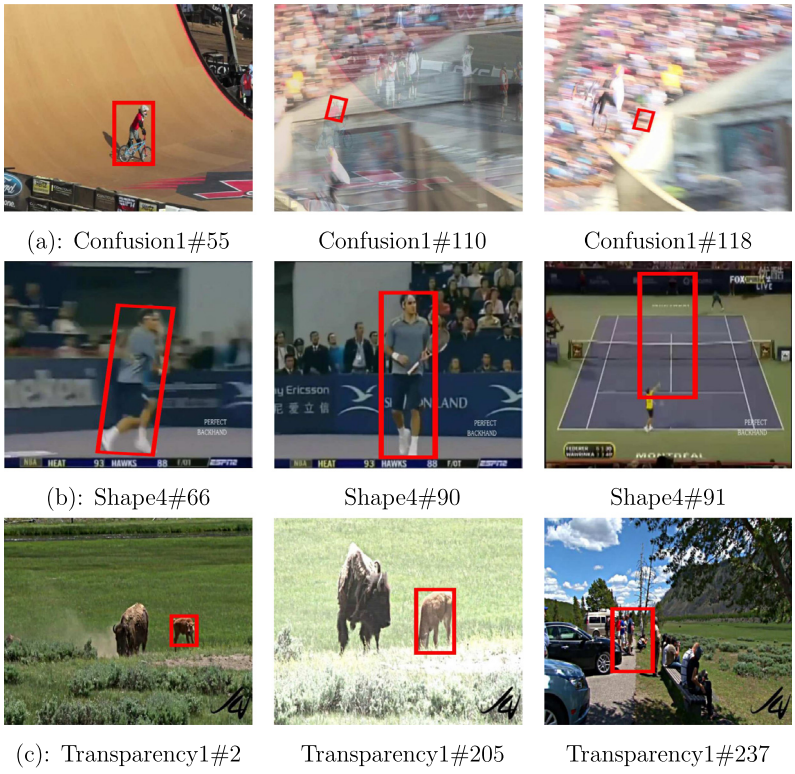


Fig. 10. Sequences where the proposed tracker enters into drift.

(Contract No. FA5209-08-P-0241) is also gratefully acknowledged. The authors also like to thank Prof. C. V. Jawahor, CVIT, IIIT, Hyderabad, India for his support during revision of this article.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.asoc.2018.09.028.

References

[1] A. Yilmaz, O. Javed, M. Shah, Object tracking: A survey, ACM Comput. Surv. 38 (4) (2006) 1–45.
[2] E. Maggio, A. Cavallaro, Video Tracking: Theory and Practice, John Wiley and Sons, Ltd., West Sussex, U. K., 2011.
[3] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, A.V.D. Hengel, A survey of appearance models in visual object tracking, ACM Trans. Intell. Syst. Technol. 4 (4) (2013) 1–48.
[4] M. Kaushal, B.S. Khehra, A. Sharma, Soft computing based object detection and tracking approaches: State-of-the-art survey, Appl. Soft Comput. 70 (2018) 423–464.
[5] F. Sardari, M.E. Moghaddam, A hybrid occlusion free object tracking method using particle filter and modified galaxy based search meta-heuristic algorithm, Appl. Soft Comput. 50 (2017) 280–299.
[6] J. Cruz-Mota, M. Bierlaire, J.-P. Thiran, Sample and pixel weighting strategies for robust incremental visual tracking, IEEE Trans. Circuits Syst. Video Technol. 23 (5) (2013) 898–911.
[7] X. Cheng, N. Li, T. Zhou, L. Zhou, Z. Wu, Object tracking via collaborative multi-task learning and appearance model updating, Appl. Soft Comput. 31 (2015) 81–90.
[8] X. Jia, H. Lu, M.-H. Yang, Visual tracking via adaptive structural local sparse appearance model, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, 2012, pp. 1822–1829.
[9] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, Robust visual tracking via structured multi-task sparse learning, Int. J. Comput. Vis. 101 (2) (2013) 367–383.
[10] S. He, Q. Yang, R. Lau, J. Wang, M.-H. Yang, Visual tracking via locality sensitive histograms, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, 2013, pp. 2427–2434.
[11] W. Zhong, H. Lu, M.-H. Yang, Robust object tracking via sparse collaborative appearance model, IEEE Trans. Image Process. 23 (5) (2014) 2356–2368.
[12] S. Avidan, Ensemble tracking, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2) (2007) 261–271.
[13] B. Babenko, M.- H. Yang, S. Belongie, Robust object tracking with online multiple instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1619–1632.
[14] S. Hare, A. Saffari, P.H. Torr, Struck: structured output tracking with kernels, in: IEEE International Conference on Computer Vision, ICCV, IEEE, 2011, pp. 263–270.
[15] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-learning-detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1409–1422.
[16] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 583–596.
[17] M. Danelljan, G. Häger, F. Khan, M. Felsberg, Accurate scale estimation for robust visual tracking, in: British Machine Vision Conference, BMVA, BMVA Press, 2014, pp. 1–11.
[18] J. Gao, H. Ling, W. Hu, J. Xing, Transfer learning based visual tracking with Gaussian processes regression, in: European Conference on Computer Vision, ECCV, Springer, 2014, pp. 188–203.
[19] K. Zhang, L. Zhang, M.-H. Yang, Fast compressive tracking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (10) (2014) 2002–2015.
[20] L.I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, John Wiley, New Jersey, 2004.
[21] S. Haykin, Neural Networks A Comprehensive Foundation, Prentice Hall, Inc., New Jersey, 1999.
[22] H. Kurokawa, C.-Y. Ho, S. Mori, A novel back propagation algorithm with optimal number of hidden units, in: Proceedings of the International Conference on Artificial Neural Networks, ICANN, Springer, 1993, pp. 783–783.
[23] Y. Freund, R.E. Schapire, A short introduction to boosting, J. Japan. Soc. Artif. Intell. 14 (5) (1999) 771–780.
[24] R. Benmokhtar, B. Huet, Neural network combining classifier based on Dempster-Shafer theory for semantic indexing in video content, in: International Conference on Multimedia Modeling, Springer, 2006, pp. 196–205.
[25] D.-S. Lee, S.N. Srihari, A theory of classifier combination: the neural network approach, in: Proceedings of 3rd IEEE International Conference on Document Analysis and Recognition, vol. 1, IEEE, 1995, pp. 42–45.
[26] A. Mondal, A. Ghosh, S. Ghosh, Neural approach for object tracking in complex environment, in: Procedings of IEEE International Conference on Systems, Man, and Cybernetics, SCM, IEEE, 2016, pp. 003516–003521.
[27] Y. Wu, J. Lim, M.-H. Yang, Online object tracking: A benchmark, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, 2013, pp. 2411–2418.

[28] J. Zhang, S. Ma, S. Sclaroff, MEEM: robust tracking via multiple experts using entropy minimization, in: European Conference on Computer Vision, ECCV, Springer, 2014, pp. 188–203.
[29] H. Song, Robust visual tracking via online informative feature selection, Electron. Lett. 50 (25) (2014) 1931–1933.
[30] N. Wang, D.-Y. Yeung, Learning a deep compact image representation for visual tracking, in: Advances in Neural Information Processing Systems, NIPS, Curran Associates, Inc., 2013, pp. 809–817.
[31] Y. Chen, X. Yang, B. Zhong, S. Pan, D. Chen, H. Zhang, CNNTracker: Online discriminative object tracking via deep convolutional neural network, Appl. Soft Comput. 38 (2016) 1088–1098.
[32] C. Ma, J.-B. Huang, X. Yang, M.-H. Yang, Hierarchical convolutional features for visual tracking, in: Proceedings of the IEEE International Conference on Computer Vision, ICCV, IEEE, 2015, pp. 3074–3082.
[33] L. Wang, W. Ouyang, X. Wang, H. Lu, Visual tracking with fully convolutional networks, in: Proceedings of the IEEE International Conference on Computer Vision, ICCV, IEEE, 2015, pp. 3119–3127.
[34] N. Wang, S. Li, A. Gupta, D.-Y. Yeung, Transferring rich feature hierarchies for robust visual tracking, 2015, arXiv preprint arXiv:1501.04587.
[35] K. Kang, C. Bae, H.W.F. Yeung, Y.Y. Chung, A hybrid gravitational search algorithm with swarm intelligence and deep convolutional feature for object tracking optimization, Appl. Soft Comput. 66 (2018) 319–329.
[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, 2009, pp. 248–255.
[37] R.F. Gonzalez, R.E. Woods, Digital Image Processing, Pearson Education, Singapore, 2008.
[38] A. Mondal, S. Ghosh, A. Ghosh, Robust global and local fuzzy energy based active contour for image segmentation, Appl. Soft Comput. 47 (2016) 191–215.
[39] C. Kim, J.-N. Hwang, Fast and automatic video object segmentation and tracking for content-based applications, IEEE Trans. Circuits Syst. Video Technol. 12 (2) (2002) 122–129.
[40] C. Wang, M. de La Gorce, N. Paragios, Segmentation, ordering and multi-object tracking using graphical models., in: IEEE International Conference on Computer Vision, ICCV, IEEE, 2009, pp. 747–754.
[41] V. Belagiannis, F. Schubert, N. Navab, S. Ilic, Segmentation based particle filtering for real-time 2d object tracking, in: Europian Conference on Computer Vision, ECCV, Springer, 2012, pp. 842–855.
[42] X. Ren, J. Malik, Learning a classification model for segmentation, in: IEEE International Conference on Computer Vision, ICCV, IEEE, 2003, pp. 10–17.
[43] F. Yang, H. Lu, M.-H. Yang, Robust superpixel tracking, IEEE Trans. Image Process. 23 (4) (2014) 1639–1651.
[44] Z. Cai, L. Wen, Z. Lei, N. Vasconcelos, S.Z. Li, Robust deformable and occluded object tracking with dynamic graph, IEEE Trans. Image Process. 23 (12) (2014) 5497–5509.
[45] S. Oron, A. Bar-Hillel, D. Levi, S. Avidan, Locally orderless tracking, Int. J. Comput. Vis. 111 (2) (2015) 213–228.
[46] M. Isard, A. Blake, Condensation conditional density propagation for visual tracking, Int. J. Comput. Vis. 29 (1) (1998) 5–28.
[47] Y. Wu, T.S. Huang, Robust visual tracking by integrating multiple cues based on co-inference learning, Int. J. Comput. Vis. 58 (1) (2004) 55–71.
[48] C. Yang, R. Duraiswami, L. Davis, Fast multiple object tracking via a hierarchical particle filter, in: IEEE International Conference on Computer Vision, ICCV, vol. 1, 2005, pp. 212–219.
[49] Z. Khan, T. Balch, F. Dellaert, A Rao-Blackwellized particle filter for eigentracking, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, vol. 2, 2004, pp. 974–980.
[50] S.K. Zhou, R. Chellappa, B. Moghaddam, Visual tracking and recognition using appearance-adaptive models in particle filters, IEEE Trans. Image Process. 13 (11) (2004) 1491–1506.
[51] P. Brasnett, L. Mihaylova, D. Bull, N. Canagarajah, Sequential Monte Carlo tracking by fusing multiple cues in video sequences, Image Vis. Comput. 25 (8) (2007) 1217–1227.
[52] X. Jia, H. Lu, M.-H. Yang, Visual tracking via adaptive structural local sparse appearance model, in: IEEE Conference on Computer vision and pattern recognition, CVPR, 2012, pp. 1822–1829.
[53] K. Zhang, L. Zhang, M.-H. Yang, Q. Hu, Robust object tracking via active feature selection, IEEE Trans. Circuits Syst. Video Technol. 23 (11) (2013) 1957–1967.
[54] W. Zhong, H. Lu, M.-H. Yang, Robust object tracking via sparse collaborative appearance model, IEEE Trans. Image Process. 23 (5) (2014) 2356–2368.
[55] V. Belagiannis, F. Schubert, N. Navab, S. Ilic, Segmentation based particle filtering for real-time 2d object tracking, in: Europian Conference on Computer Vision, ECCV, 2012, pp. 842–855.
[56] F. Yang, H. Lu, M.-H. Yang, Robust superpixel tracking, IEEE Trans. Image Process. 23 (4) (2014) 1639–1651.
[57] S. Oron, A. Bar-Hillel, D. Levi, S. Avidan, Locally orderless tracking, Int. J. Comput. Vis. 111 (2) (2015) 213–228.
[58] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR, vol. 1, 2005, pp. 886–893.
[59] R. Mukundan, K.R. Ramakrishnan, Moment Functions in Image Analysis: Theory and Applications, World Scientific, Singapore, 1998.
[60] A. Bhattacharyya, On a measure of divergence between two statistical populations defined by their probability distribution, Bull. Calcutta Math. Soc. 35 (1) (1943) 99–109.
[61] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, J. Zhang, Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics and protocol, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 319–336.