Structurally Enhanced Correlation Tracking

Mayur Rajaram Parate[1], Kishor M. Bhurchandi[2]
[1,2] Visvesvaraya National Institute of Technology, Nagpur
Maharashtra, India.
[1][e-mail: mrparate1787@gmail.com]
[2] [e-mail: bhurchandikm@yahoo.co.in]
*Corresponding author: Mayur Rajaram Parate

Received November 23, 2016; revised March 22, 2017; accepted May 14, 2017;
published October 31, 2017

Abstract

In visual object tracking, Correlation Filter-based Tracking (CFT) systems have arouse recently to be the most accurate and efficient methods. The CFT's circularly shifts the larger search window to find most likely position of the target. The need of larger search window to cover both background and object make an algorithm sensitive to the background and the target occlusions. Further, the use of fixed-sized windows for training makes them incapable to handle scale variations during tracking. To address these problems, we propose two layer target representation in which both global and local appearances of the target is considered. Multiple local patches in the local layer provide robustness to the background changes and the target occlusion. The target representation is enhanced by employing additional reversed RGB channels to prevent the loss of black objects in background during tracking. The final target position is obtained by the adaptive weighted average of confidence maps from global and local layers. Furthermore, the target scale variation in tracking is handled by the statistical model, which is governed by adaptive constraints to ensure reliability and accuracy in scale estimation. The proposed structural enhancement is tested on VTBv1.0 benchmark for its accuracy and robustness.

Keywords: Visual tracking, object tracking, correlation filter, computer vision, target representation, target scale estimation

https://doi.org/10.3837/tiis.2017.10.013 **ISSN : 1976-7277**

1. Introduction

Visual object tracking is a predominant part of computer vision due to its various applications in real life senarios like surveillance, crowd understanding, motion detection, classification, recognition and human-computer interfaces. Designing a tracking algorithm to handle all situations is a difficult task bacause of the challanges involved in tracking, like scale variation, occlusion, illumination variation, motion blur, fast motion, etc. [1, 2].

Various generative and discriminative tracking algorithms have been proposed over the past decade to overcome these challanges. Discriminative approaches [3-5] consider tracking as a classification that distinguish the target from the background whereas generative approaches [6-12] find the best-matching window to the tracked target. In [13], Hare et al. proposed the discriminative approach using structured output support vector machine (S-SVM) in which the highest discriminant score corresponds to the new target location irrespective to the scale of the target. Babenko *et al.* [7] proposed to estimate the target location using a discriminative classifier trained using multiple instance learning. In [14], a binary classifier is trained on labeled and structured unlabeled data samples to detect the traget during tracking. In [15], the target appearance and motion are modeled based on the timely components; descriptive, discriminative and regressive. Recently, correlation based discriminative trackers are found to be more promissing in achieving better accuracy at higher frames per second (FPS) [16]. In [17], Bolme *et al*. proposed the correlation filter with the adaptive training approach to efficiently track the target, with the use of kernels it is further improved in [18]. In [19], the target representation in tracker is improved further using the color attributes for better efficiency. Despite of their efficiency and speed, the correlation filter based trackers suffer from inherent inability to handle target scale variations and heavy occlusions.

In this paper, we address the discussed problems by structurally improving the correlation filter using the global and the local layers of target representation. The global and local layers combinely captures the holistic and local appearance of the target which is improved by the addition of the reverse RGB channels. A strategy to adaptivly combine the confidence maps of global and local layer is developed. This strategy provides the robustness to background changes and heavy occlusion. Furthermore, the target scale variations are effectively handled by developing a statistical model which is governed by two adaptive constraints. Extensive experiments are conducted on VTBv1.0 [20] to test the structural enhancement in correlation filter based tracker.

Remainder of this article is organized as follows; in Section 2, we review the previous work in correlation filter based tracking. Section 3 presents conventional correlation tracking. Section 4 introduces the structural enhancement for handling heavy occlusion and target scale variation. Section 5 presents extensive experimentation to evaluate performance of the proposed algorithm and its benchmarking with the existing state-of-the-art trackers. We conclude the manuscript in Section 6.

2. Related Work

The correlation filter based tracking methods have proven to be competative with far more complicated tracking methods but they are inefficient in handling heavy occlusion and estimating the scale of the target during tracking.Various algorithms based on part based representation of the target have been proposed to address the issue of heavy occlusion [21-25]. In correlation filter based tracking, [26, 27] have made attempts to use part based tracking stategy. In [26], the target is decomposed into five parts and each part is independently tracked by KCF to generate respective confidence maps. The particle filter is used to combine five confidence maps to get the final confidence map. In [27], the local context of the target is captured by sampling patches and calculating their reliabilities. A patch which is trackable and sticking to the target is considered as reliable. The new state of the target is obtained by combining the tracking results of the reliable patches using Hough Voting-like scheme. It is found that particle filter is effective in combining the multiple confidence maps but itself suffers from various issues [28].

Furthermore, conventional CFTs use fixed-sized windows for tracking and they are unable to handle scale changes of the target. However, many attempts for handling scale have been proposed in the recent years [26, 27, 29-33]. In [29-32], a scaling pool method has been employed to find the scale of the target. The scaling pool involves sampling of different sized windows around the target and is correlated with the trained correlation filter. Further, the window with the maximum correlation score is selected as new state of the target. Let s_0 be the template window size, so we can have $s_i=a_is_0$ where, $\Upsilon=\{a_{1,}\ a_{2,}\ a_{3\ .....}\ a_N\}$ is scaling pool and N is positive numbers. In [29], Υ is set by constant values 0.985 to 1.015 with increment of 0.005 and in [30], Υ is given by (1).

$$\Upsilon=\left\{a_n \mid n=\left\lfloor -\frac{N-1}{2}\right\rfloor,\dots,\left\lfloor\frac{N-1}{2}\right\rfloor\right\} \tag{1}$$

The scaling pool is computationally expensive and does not ensure accuracy in scale estimation of the target. Further, in [26], the joint correlation map is used in Bayesian inference framework to estimate the target candidate with maximum posterior probability and subsequently used to estimate the target scale. Whereas, [33] estimates the scale $s_t^{'}$ at time t by

$$s_t^{'}=\sqrt{\frac{l((p_0)_t)}{l((p_0)_{t-1})}} \tag{2}$$

where p_o is new estimated center of the target and $l(p_o)$ is its computed confidence score. But scale $s_t^{'}$ becomes very large and gets unstable when denominator is close to zero.

3. Correlation Filter based Tracking

According to the published correlation filter based tracking methods, set of training image patches from the given target position and the training outputs are required to train the correlation filters. The training output is usually Gaussian with its peak centered at the target center in the training image. From the set of training patches, various features are extracted and a cosine window is applied for smoothing boundary effects. The target is tracked by correlating filter over a search window. Efficient correlation operations are performed using element wise multiplications in Discrete Fourier Transform (DFT) domain instead of computationally exhaustive convolutions in spatial domain. Generally, DFT is efficiently computed using Fast Fourier Transform (FFT) algorithm. The spatial position corresponding to the maximum value in confidence map obtained using Inverse Fast Fourier Transform (IFFT) is predicted as the new target position. Appearance update is performed in frequency domain at the predicted position as only DFT of the correlation filter is required for detection, training and updating.

We have used Kernelized Correlation Filter (KCF) [34] as a base for addressing the mentioned lacunae in correlation filter based trackers. It represents that the ridge regression principle and circulant matrix can be effectively used to kernelize the correlation filters. Considering the correlation filters as classifiers, they can be trained with i^{th} input x_i and its label y_i using ridge regression. Let X be a two dimensional data matrix containing the patch features corresponding to the first patch in terms of Histogram of Gradients (HOG) [35] and y be the Gaussian regression labels with its center at the target position in training patch. The goal of training is to find the function $f(x_i)$ so as to minimize the squared error over x_i and its regression label y_i as presented in (3).

$$\min_{w} \sum_{i}^{n} (y_i - f(w, x_i))^2 + \lambda \| w \|^2 \tag{3}$$

Where, where λ is regularization parameter to restrict the over training of correlation filters. Solving (3) according to [36], the weight w is given as (4).

$$w = (X^T X + \lambda I)^{-1} X^T y \tag{4}$$

Where, y is a matrix of corresponding regression labels and I is an identity matrix.

In KCF, the samples for training the regression are obtained using the circulant matrix $X = C(x)$. Where $x = (x_0, x_1, ..., x_{n-1})$ is a base sample which is the first row of circulant matrix. The circulant matrix X collects all the translated samples around the target without sacrificing much speed. It exhibits the property of representing the matrix X using $\text{diag}(\hat{x})$ as presented in (5). We use '^' to represent DFT of a vector and F is a constant Discrete Fourier Transform (DFT) matrix.

$$X = F\, diag(\hat{x}) F^H \tag{5}$$

where, H represents the Hermitian transpose of a matrix. With the ability of representing a matrix using its diagonal, all the operations can be performed element-wise on diagonals.

To improve performance, the 'kernel trick' [37] is introduced which represents the optimization problem in different set of variables (the dual space) while keeping it linear. In kernel trick, the kernel function is used with non-linear feature space $\varphi(x)$ and w is expressed as linear combination of inputs, $w = \sum_i \alpha_i \varphi(x_i)$ making α_i to be the variable under optimization. Then $f(x_i)$ takes the form,

$$f(x_i) = \sum_{j=1}^{n} \alpha_i k(x_i, x_j) \tag{6}$$

where, $k(x_i, x_j) = \langle \varphi(x_i) \varphi(x_j) \rangle$ is the kernel function. Considering K is a kernel matrix with its elements $K_{ij} = k(x_i, x_j)$, the solution of (3); the kernelized version of ridge regression is given in [36] as (6). In a new frame, it is possible to compute the regression function for all candidate patches using (6). Further, to compute regression function efficiently it is represented in terms of kernel correlation $\hat{k}^{xz}$ between the maintained base sample x and a new sample z in Fourier domain given by [34] as (7).

$$\hat{f}(z) = \hat{k}^{xz} \Theta \hat{\alpha} \tag{7}$$

where, $\hat{\alpha} = \dfrac{\hat{y}}{\hat{k} + \lambda}$ and $k^{xz} = \exp\left(-\frac{1}{\sigma^2}(\| x \|^2 + \| z \|^2 - 2F^{-1}(\hat{x}^* \Theta \hat{z}))\right)$

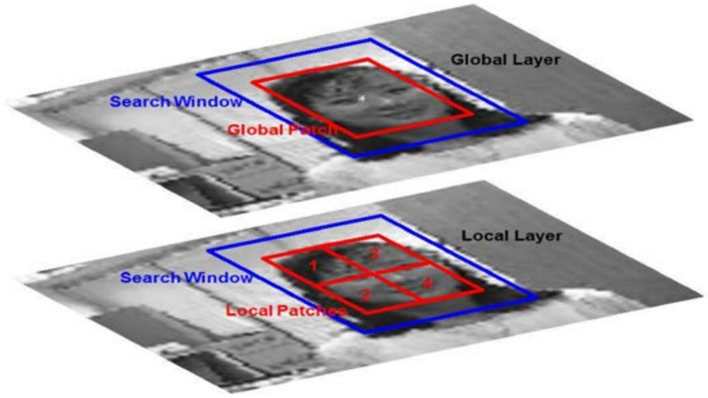


Fig. 1. The global and local layer representation of the target. Search window is 1.5 times the target size (i.e. height and width).

So, for a new sample z, the confidence map y i.e full detection response over all cyclic shifts of z in spatial domain is given by inverse DFT of (7).

$$y = F^{-1}(\hat{f}(z)) = F^{-1}(\hat{k}^{xz} \Theta \hat{\alpha}) \tag{8}$$

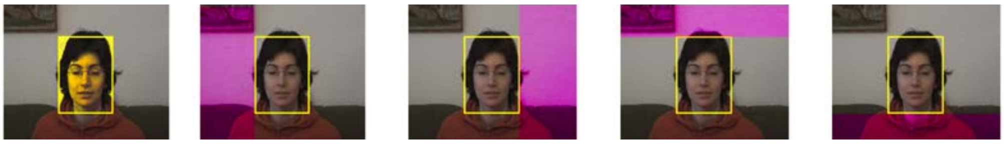
The spatial position corresponding to the maximum value in y can be considered as new position of the target. The detailed derivations of equations (6-8) can be found in [34] .

4. Proposed Structured Correlation Tracking

We aim to develop an online tracking algorithm that can handle heavy occlusions without being prone to drifting and also estimate the scale of target during tracking. This is achieved by structurally enhancing the correlation filter to use both global and local appearances of the target. For this, we decompose the target into a global layer and a local layer. The global layer consist of complete representation of the target, which is essential in fast motion, in-plane rotation and camera motion. Whereas, the local layer have four local patches extracted from the target and plays an important role in heavy occlusions and background changes. The global and local layer representation is shown in **Fig. 1**. Furthermore, the target representation is enhanced by embedding reverse RGB channels along with original RGB channels to prevent loss of object in dark background. A new position of the target is estimated using the adaptively weighted sum of confidence maps corresponding to global and local layers. The scale estimation is based on statistical model, utilizing the relative position displacement between the patches of global and local layers.

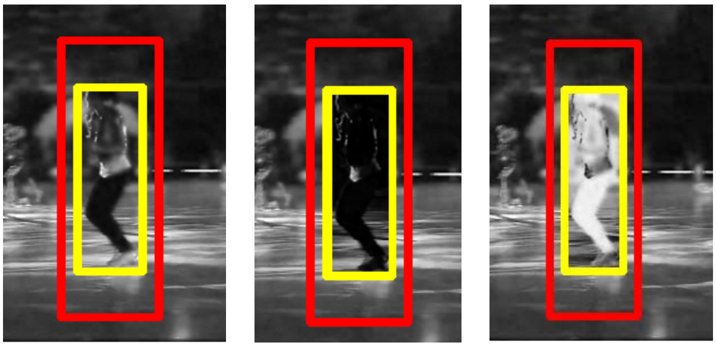
4.1 Target Enhancement and Background Suppression

In conventional circularly shifted correlation tracking methods, the searching window needs to be large enough to track the target with motion. However, the background context in the search window changes frequently and becomes more and more chaotic after circular shifting, which may cancel out the contribution of the target when applying the regression. Therefore, in order to enhance the target representation, a search window is preprocessed. In this, we suppress background and enhance the target using a likelihood that a pixel belongs to the target employing color histogram based Bayes classifier on a search window S. Let B represent the background surrounding the target T and $H_R(b)$ represents b^{th} bin of the color histogram over a given region R.



Target Left Right Top Bottom

Fig. 2 (a). Layout of the target and the surrounding used for estimating joint probability distributions for the target enhancement and background suppression.



Original image Enhanced without reverse RGB channels Enhanced with reverse RGB channels

Fig. 2 (b). Target enhancement using reverse RGB channels in a dark background situations.

Then, the likelihood of a pixel at position p with bin-index b_p that belongs to the target is calculated using (9).

$$P(p \in T \mid b_p) \approx \frac{P(b_p, p \in T)}{P(b_p, p \in B) + P(b_p, p \in T)} \tag{9}$$

Where, $P(b_p, p \in T)$ is estimated using $H_T(b_p)P(p \in T)$. And, for the background B; estimaion of joint probability distribution $P(b_p, p \in B)$ is not straight forward, as the background has four parts and are relatively away from each other. So, to estimate the joint probability distribution $P(b_p, p \in B)$, the background is divided into four rectangular parts as presented in **Fig. 2 (a)** and maximum of the four probability distributions is considered as defined in (10).

$$P(b_p, p \in B) \approx \max_{i=1:4} H_{B_i}(b_p)P(p \in B_i) \tag{10}$$

Therefour, the likelihood of a pixel that belongs to the background is given as;

$$P(p \in B \mid b_p) \approx \frac{P(b_p, p \in B)}{P(b_p, p \in B) + P(b_p, p \in T)} \tag{11}$$

Furthermore, the prior probability of a pixel that belongs to the given region R can be approximated as $P(p \in R) = |R| / |S|$, $R = T, B_1, B_2, B_3, B_4$. Where, |.| represents cardinality.

Multiplying the search window with the likelihood map estimated using (9), the background pixels in a serch window can be suppressed which yields enhanced target representation. This facilitates the development of robust correlation filter based tracker. Further, in tracking with dark background; it is obseved that the tracker often drifts while tracking dark object. To address this, we encorporate reverse RGB channels i.e (255-R, 255-G, 255-B) along with the original RGB channels in the color histogram to enhance the target representation as shown in **Fig. 2 (b)**. It can be seen that, for a dark target on dark background, the use of only RGB channels for enhancing the representation dose not yield prominent boundaries. Whereas, the use of reverse RGB induces visible boundary between the target bounding box and the background. Then, the search window is multiplied by the target likelihood given by (9). This gives prominent distinguishing boundary between the target and background surrounding it, even if the background is dark, which is very important in tracking objects especially when using HOG features.

4.2 Target Position and Scale Estimation

The conventional correlation filter based trackers perform tracking considering only global appearance of the target. However, it is also beneficial to track the target based on its local appearance. To take advantage of local as well as global appearance both, we represent the target using global layer and local layer as already presented in **Fig. 1**. The size of each patch in local layer is one fourth of the global patch in global layer. Spatially, the sizes and positions of the global and local patches are kept fixed such that, each quadrant ($Q_i \mid i$=1:4) will have one local patch, considering origin O as center of the target. We apply correlation filter based tracker defined by (8) independently to track position of each of the four local patches and the global patch. In the successive video frame, the correlation filter computes the confidence maps of the global patch and all the four local patches. The position and scale estimation is explained further and its flow is presented in **Fig. 3.**

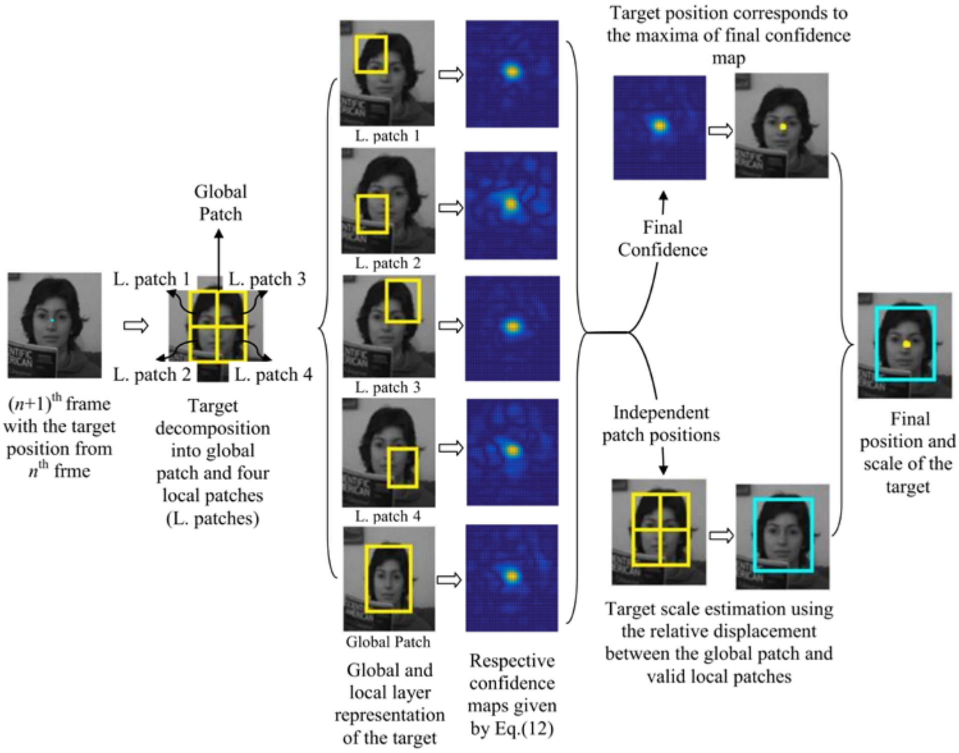

Target position corresponds to the maxima of final confidence map
Global Patch
L. patch 1
L. patch 3
L. patch 2
L. patch 4
(n+1)th frame with the target position from nth frme
Target decomposition into global patch and four local patches (L. patches)
L. patch 1
L. patch 2
L. patch 3
L. patch 4
Global Patch
Global and local layer representation of the target
Respective confidence maps given by Eq.(12)
Final Confidence
Independent patch positions
Target scale estimation using the relative displacement between the global patch and valid local patches
Final position and scale of the target

Fig. 3. Flowchart of the proposed tracking system. The candidate patch having the target location from the previous frame as its center is decomposed into global and local layer. The final confidence map estimated as adaptively weighted mean of independent confidence maps calculated using (8).
The scale of the target is derived utilizing the relative displacement between the global patch and valid local patches.

The spatial position corresponding to the maximum value in final confidence map is the new target position. The difficulty lies in developing an appropriate framework to combine these confidence maps from independently tracked parts of the target. As there may be relative displacement between the tracked positions of global and local patches, it is reasonable to fuse the confidence maps from global and local layers and generate a final map. Thus, to estimate the final confidence map, we adaptively weight the confidence maps from global and local layers. This is achieved by offering higher weights to the patches those are having higher confidence and those with lower confidence i.e. occluded parts get lower weights. Thus, the final confidence map y_w to locate the target in given search window is defined by (12).

$$y_w = \sum_{i=1}^{5} \psi_i y_i \qquad (12)$$

where, ψ_i is an adaptive weight obtained by (13).

$$\psi_i = \frac{\max(y_i)\tau_i}{\sum_{i=1}^{5} \max(y_i)\tau_i} \qquad (13)$$

where, τ_i is set to 1 for global patch (i=5) while for the local quarters (i=1:4) it is set at 0.25. This is because, each layer i.e. global and local is having equal importance and the number of patches in the global layer and the local layer are one and four respectively. Thus, the spatial position corresponding to the maxima of y_w is considered as the new position of the target.

While tracking in a video sequence, when the target moves closer to the camera its size increases and results in expansion of local appearance distribution over the video frame. As the trackers corresponding to the local patches track respective local appearance of the target, the estimated positions of local patches move away from each other continuing to track the local part of the target. This relative displacement between the positions of global patch and local patches is used to estimate the scale of the target as described further.

Let $Qtpos_n^1$, $Qtpos_n^2$, $Qtpos_n^3$, $Qtpos_n^4$ and C_1, C_2, C_3 and C_4 represent the centers and corners of the four local patches as shown in **Fig. 4 (a)**. Assuming that, the target does not move more than half of its height and width in one frame, we draw an extreme virtual outer bounding box (outer BB) with height and width 1.5 times than that of the template window (i.e height and width). The scale of the target depends on the relative spatial positions of the global patch and the four local patches. It can be further illustrated using vector algebra for local patch centered at $Qtpos_n^3$.

Consider vectors $\overline{C}_{Th3}$ and $\overline{C}_3$ corresponding to corner $C_{Th3}(x_{Th3}, y_{Th3})$ of the outer bounding box and corner $C_3(x_3, y_3)$ of a local patch in quadrant Q_3 respectively. As the target is not moving more than half of its height or width in one frame; the center $Qtpos_n^3$ of local patch must lie inside the quarter curve with radius $\| \overline{C}_{Th3} \|$ as shown in **Fig. 4 (b)**. (||.|| represents norm of a vector.) If it moves outside the outer bounding box, it can be considered as drifted and should not be involved in scale estimation of the target. Further, it must lie inside the quadrant Q_3; this is because, the spatial positions of local patches in local layer are fixed with respect to the quadrants.

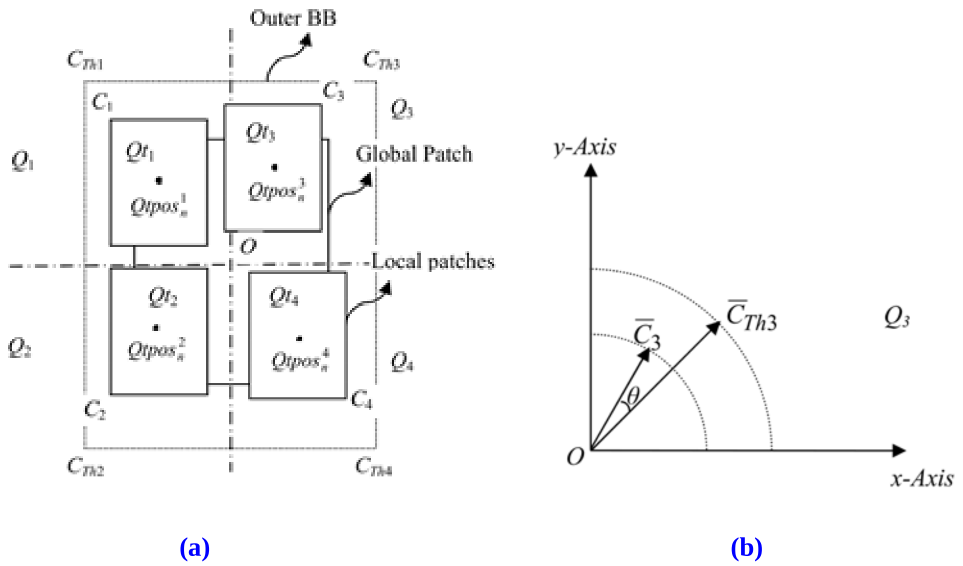


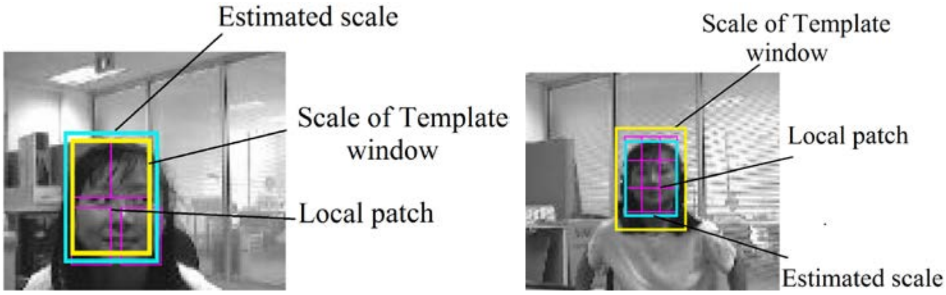
Fig. 4. (a) Relative positions between the global patch and the detected local patches during tracking. **(b)** Illustration of the constraints for scale estimation using vector algebra for local patch Qt_3 in quadrant Q_3 . (BB-Bounding Box).

Thus, the adaptive constraints for the local patch centered at $Qtpos_n^3$ to be valid are 1) the magnitude of $\overline{C}_3$ should be less than magnitude of $\overline{C}_{Th3}$ and 2) angle between $\overline{C}_3$ and $\overline{C}_{Th3}$ i.e. θ should be less than or equal to |π/4|. So, for local patches (Qt_i | i=1:4) the validity criteria is given by adaptive constrains conjointly represented as,

$$v_i = \begin{cases} 1 & if \quad \|\overline{C}_i\| \le \|\overline{C}_{Ti}\| \; and \; \theta_i \le |\pi/4| \\ 0 & otherwise \end{cases} \tag{14}$$

where, v_i is a validity flag for i^{th} local quarter and ||.|| represents the magnitude of a vector. As the size of each local patch is fixed for a video sequence, the corners C_1, C_2, C_3 and C_4 corresponding to each local patch are estimated easily. Thus, for the valid local patches, we have their corners estimated from their centers as indicated in **Fig. 4 (a)**. Therefore, the width or height of a target can be found out as maximum spatial difference between the corners and is presented in terms of Δx and Δy respectively. Depending on how many local patches are valid, there are four cases; any one local patch is valid, two local patches are valid, three local patches are valid and all the four local patches are valid. The height, width and the scale ($scale_{n+1}$) of the target in the next frame is estimated by (15). Whereas, in (15) the computations corresponding to only valid patches are considered. The process of the target scale estimation slightly varies depending on the number of valid local patches. In case of two non-diagonal valid quarters, based on outer corner coordinates, we can estimate either height or width. Then the other parameter is estimated using the known aspect ratio (A_r) from the previous frame. While, in case of single valid patch, the scale of the target from previous frame is maintained.

$$\left. \begin{aligned} &\Delta x = \max(v_1v_3\,|x_1 - x_3|,\, v_2v_4\,|x_2 - x_4|,\, v_1v_4\,|x_1 - x_4|,\, v_2v_3\,|x_2 - x_3|) \\ &\Delta y = \max(v_1v_2\,|y_1 - y_2|,\, v_3v_4\,|y_3 - y_4|,\, v_1v_4\,|y_1 - y_4|,\, v_3v_2\,|y_3 - y_2|) \\ &\Delta x = A_r \times \Delta y \end{aligned} \right\} \tag{15}$$



(a) (b)

Fig. 5. Target scale changes (a) When the target comes closer to camera, local patches move away from each other. (b) The target is moving far from the camera; local patches come closer to each other and overlap.

It is observed during tracking that when the target moves closer to the camera, its size increases and the trackers corresponding to the local patches track respective local appearance of the target. Therefore, the estimated positions of the local patches move away from each other continuing to track the local part of the target. This results in the increase in estimation of the scale as shown in the **Fig. 5 (a)**. Similarly, when the target moves away from the camera, its size decreases resulting in position of the local patches to come closer to each other and overlap. This decreases the estimated scale of the target as shown in the **Fig. 5 (b)**.

The complete process of the proposed tracking algorithm along with occlusion handling and scale estimation is presented in **Algorithm 1**.

```
Algorithm 1: Structurally enhanced correlation tracking
Input: Current frame $n$, Target position $Pos_n$, Target scale $scale_n$ and Video frame $frame_n$.
1. Enhance target representation and suppress background using Eq. (9) and Eq.(11)
2. Decompose the target into global and local layers w.r.t. $Pos_n$ and $scale_n$ from $frame_n$
3. Get center position of global patch $Pos_n$ and local patches $Qtpos_n^i \mid i = 1:4$
4. For $i$=1 to 4 do
       Train using Eq.(3) w.r.t. $Qtpos_n^i$ and get $\hat{\alpha}_i$ .
   end For
5. Train using Eq.(3) w.r.t. $Pos_n$ and get $\hat{\alpha}_5$ .
6. Read Frame $frame_{n+1}$
7. For $i$=1 to 5 do
       Find confidence map $y_i$ using Eq.(8) and $\hat{\alpha}_i$
       Estimate spatial positions corresponding to max($y_i$).
   End For
8. Find adaptive weights using Eq.(13)
9. Find final target position $Pos_{n+1}$ using Eq.(12).
10. Estimate $v_i$ using Eq.(14).
11. Calculate $scale_{n+1}$ as spatial difference between valid local patches using Eq.(15).
Output: Target position $Pos_{n+1}$ and Target scale $scale_{n+1}$ for frame ($n$+1).
```

5. Experimental Results and Discussions

The proposed enhancements in correlation filter based tracker are extensively tested using Visual Tracker Benchmark v1.0 (VTBv1.0) [20] dataset having 50 fully annotated video sequences and total 29,522 video frames. In our implementation, KCF is used as a base tracker and search window sizes for the global and local patches are set to 1.5 times the target patch (i.e. height and width). The correlation filters are trained based on the Histogram of Oriented Gradients (HoG) with cell size of 4 pixels and 9 orientations which yield a feature matrix of size (N/4 x M/4 x 32). Here NxM is size of search window. The regression labels are Gaussian of size (N/4 x M/4) with its peak at the center of the target in previous frame. The performance of the proposed method is compared with recently published tracking algorithms namely KCF [34], CN [19], STC [33], Struck [13], TLD [38], ASLA [25], MIL [7], Frag [39], RACF [26] and CNT[40] using area under the precision and success curves [20]. The tracking results of the compared methods are obtained using the codes provided by the respective authors with recommended default parameters and using the suggested initial settings. The experiments were conducted on Intel(R) Core(TM)i7-4770 CPU @ 3.4GHz with 32GB RAM system.

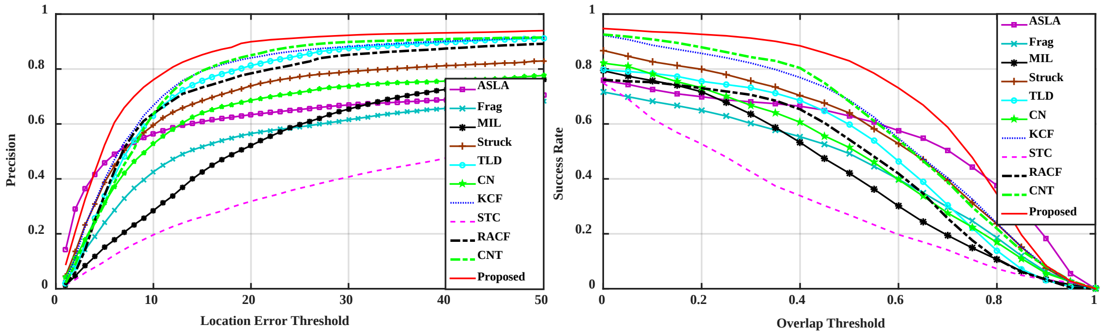
5.1 Performance Evaluation Methodology

We have used two evaluation parameters i.e. area under the precision and success curves [20] to rank the algorithms**.** The precision is a performance measure for a tracker to evaluate its ability to localize the target in a frame with respect to the given ground truth location. Precision is the percentage of number of frames whose location is within the given location error threshold compared to the ground truth. Precision curve is a plot between the precision and the location error threshold varying from 0 to 50 in pixels. The Location error threshold is the acceptable error between the spatial centers of the detected target and the center of the target according to the ground truth for each frame. The area under the precision curve is used to rank the trackers robustly instead of using precision value at one particular threshold.

Success curve is used to evaluate tracker, for its ability to determine the scale of the target to be tracked. The success curve is a plot between the success rate and the overlap threshold varying from 0 to 1. The overlap score S is given by $\frac{|BB_t \cap BB_{gt}|}{|BB_t \cup BB_{gt}|}$, where, BB_t is tracked bounding box area and BB_{gt} is ground truth bounding box area. Further, $\cap$, $\cup$ and $|\ .\ |$ represent intersection, union and number of pixels in the region. Then the percentage of frames whose overlap score S is equal to or larger than the overlap threshold is estimated to obtain the success rate. The area under the success curve is used to rank the trackers robustly instead of using success rate at one particular threshold value.

5.2 One Pass Evaluation (OPE)

This is a conventional way of evaluating tracking algorithms by initializing them in the first frame and allowing them to run throughout the sequence. OPE gives the overall performance of the tracker in presence of all the mentioned challenges and can be used for ranking the tracking algorithms. The ranking is based on area under the precision and success curves. **Fig. 6** shows the precision and success curves for OPE on VTBv1.0 database. It can be observed in **Fig. 6 (a)** that the proposed structural enhancement results in achieving higher



(a) (b)

Fig. 6. One Pass Evaluation (OPE) on VTBv1.0 database; (a) Precision Curves (b) Success Curves

Table 1. One Pass Evaluation; Area under Precision Curves (APC), Area under Success Curves (ASC) and Average Frames per second (FPS) are tabulated for each algorithm on VTBv1.0 database. The highest values are shown bold.

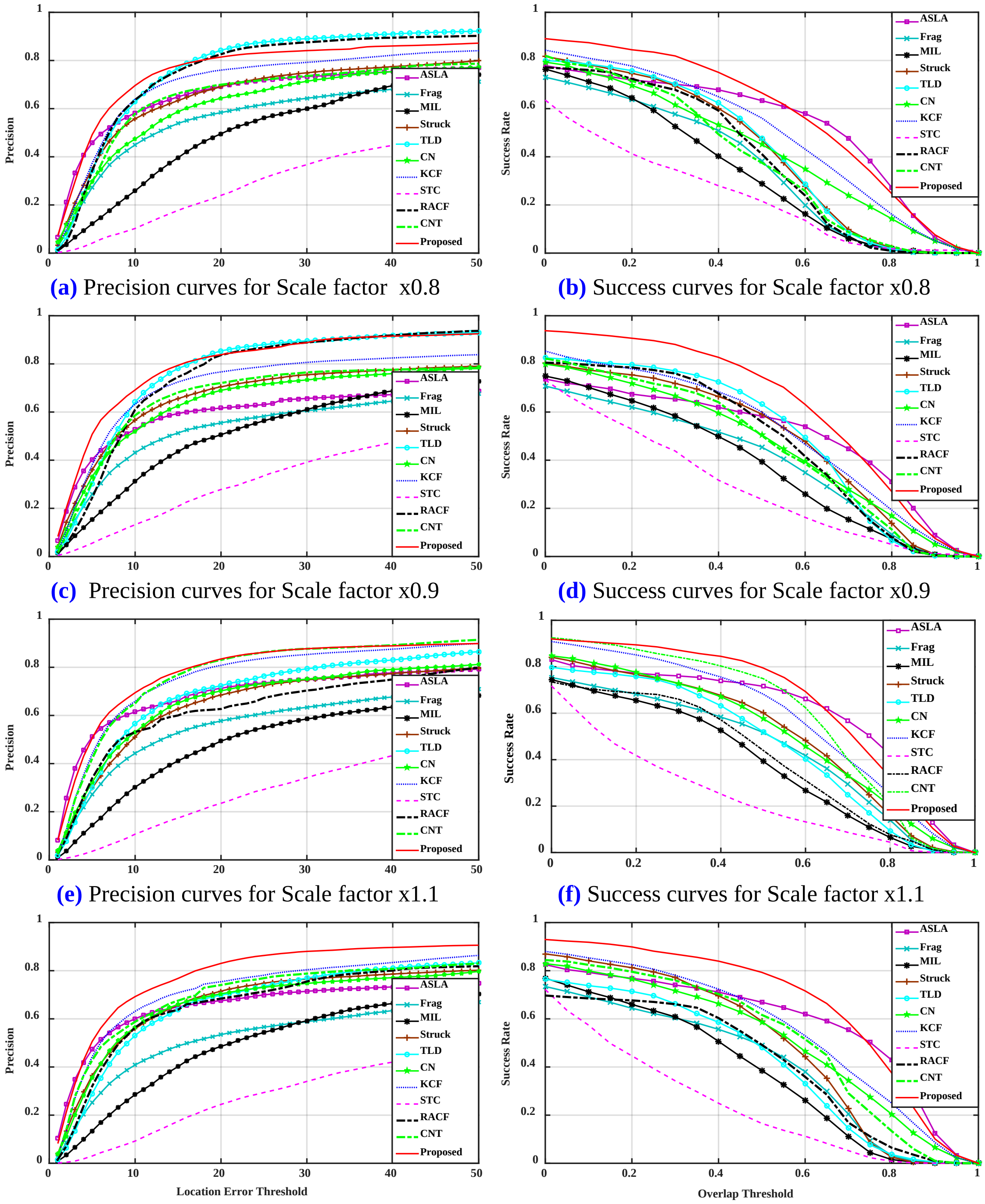
	STC	MIL	Frag	ALSA	CN	Struck	TLD	KCF	RACF	CNT	Proposed
APC	0.3350	0.5181	0.5268	0.6062	0.6286	0.6828	0.7404	0.7609	0.7233	0.7560	**0.8159**
ASC	0.2995	0.4090	0.4282	0.5414	0.4612	0.5388	0.4890	0.5750	0.4575	0.5799	**0.6707**
FPS	183	38	19	7.48	132	20.4	33.3	**191**	38.7	5.42	46.2

precision at lower thresholds compared to the KCF and other algorithms. This indicates the increase in tracking accuracy by the proposed algorithm eliminating small drifts due to partial occlusions. Also, it can be seen that the proposed algorithm achieves higher area under the precision curve with larger margin than the second rank KCF. Further, from **Fig. 6 (b)**, we observe that, the proposed scale estimation method efficiently handles the target scale variations over a wide range. This results in high success rate and subsequently yields higher area under the success curve by the proposed algorithm curve compared to the state-of-the-art algorithms. **Table 1** presents the area under the precision curve (APC), area under the success curves (ASC) and average frames per second for OPE. It can be observed that STC yields the lowest performance; this is because of instability in the scale estimation of STC.

5.3 Spatial Robustness Evaluation (SRE)

Apart from the OPE, in Spatial Robustness Evaluation (SRE), the trackers are tested for the target scale variations during their initialization. In SRE, the trackers are initialized with a bounding box scaled by scaling the original target bounding box by factors 0.8, 0.9, 1.1 and 1.2. With the scaled bounding box initialization, trackers are allowed to run throughout the sequences. The scaling of bounding box creates perturbations in initialization of the trackers. Thus, SRE helps in evaluating the response of trackers for the perturbations during their initialization. The precision and success curves for SRE are shown in **Fig. 7**. **Table 2** presents the area under the precision and success curves on all the video sequences from VTBv1.0 database corresponding to SRE. The tracking results i.e. precision and success rates are obtained using estimated target positions and corresponding scaled ground truth data from the database.

We have observed that the ability of STC, MIL, ASLA, Struck and TLD trackers is affected by the change in scale during initialization. STC, MIL, TLD and RACF show reduction in area under the precision curve for higher scale factors whereas, ASLA shows increase in area under the precision curve at higher scale factors.



(a) Precision curves for Scale factor x0.8

(b) Success curves for Scale factor x0.8

(c) Precision curves for Scale factor x0.9

(d) Success curves for Scale factor x0.9

(e) Precision curves for Scale factor x1.1

(f) Success curves for Scale factor x1.1

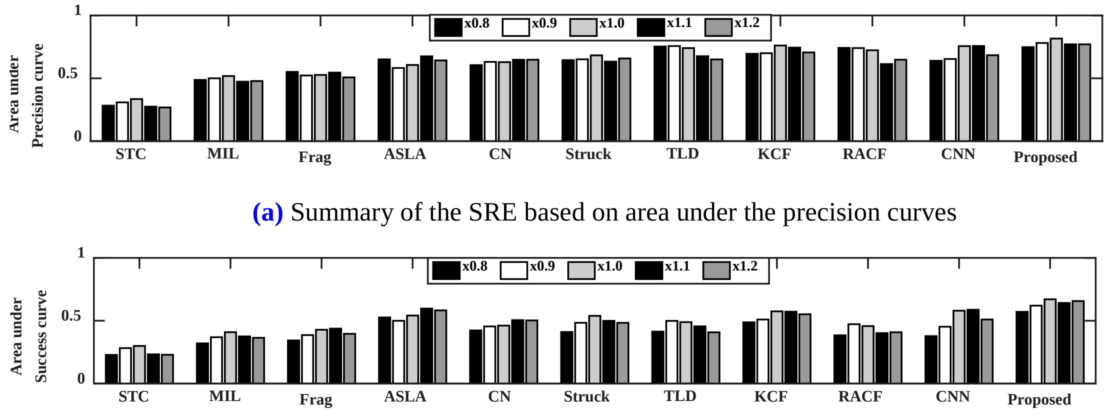
(g) Precision curves for Scale factor x1.2

(h) Success curves for Scale factor x1.2

Fig. 7. Spatial Robustness Evaluation (SRE) based on precision and success curves. The trackers are initialized with different sizes of bounding box with scale factors 0.8, 0.9, 1.1 and 1.2. (a), (c), (e) and (g) show precision curves and (b), (d), (f) and (h) show success curves for scale factors 0.8, 0.9, 1.1 and 1.2 respectively.

Table 2. Spatial Robustness Evaluation (SRE) based on Area under the Precision Curve (APC) and Area under the Success Curves (ASC) for VTBv1.0 database.

Tracker /	x0.8		x0.9		x1.1		x1.2	
Scale	APC	ASC	APC	ASC	APC	ASC	APC	ASC
ASLA	0.6507	0.526	0.5821	0.4996	0.6749	0.5968	0.6423	0.5826
Frag	0.5508	0.3432	0.5225	0.3858	0.5453	0.4373	0.5075	0.3962
MIL	0.4870	0.3200	0.5001	0.3690	0.4731	0.3754	0.4783	0.3643
Struck	0.6444	0.4104	0.6513	0.4837	0.6328	0.4992	0.6576	0.4834
TLD	**0.7528**	0.4134	0.7566	0.4989	0.6757	0.4555	0.6502	0.4080
CN	0.6048	0.4228	0.6305	0.4547	0.6476	0.5048	0.6477	0.5026
KCF	0.6949	0.4877	0.6998	0.5098	0.7441	0.5720	0.7060	0.5517
STC	0.2826	0.2281	0.3088	0.2824	0.2746	0.2328	0.2683	0.2294
RACF	0.7416	0.3841	0.7405	0.4727	0.6132	0.4017	0.6480	0.4084
CNT	0.6392	0.3775	0.6536	0.4522	0.7568	0.5874	0.6838	0.5100
Proposed	0.7479	**0.5705**	**0.7809**	**0.6200**	**0.7705**	**0.6413**	**0.7705**	**0.6564**



(a) Summary of the SRE based on area under the precision curves

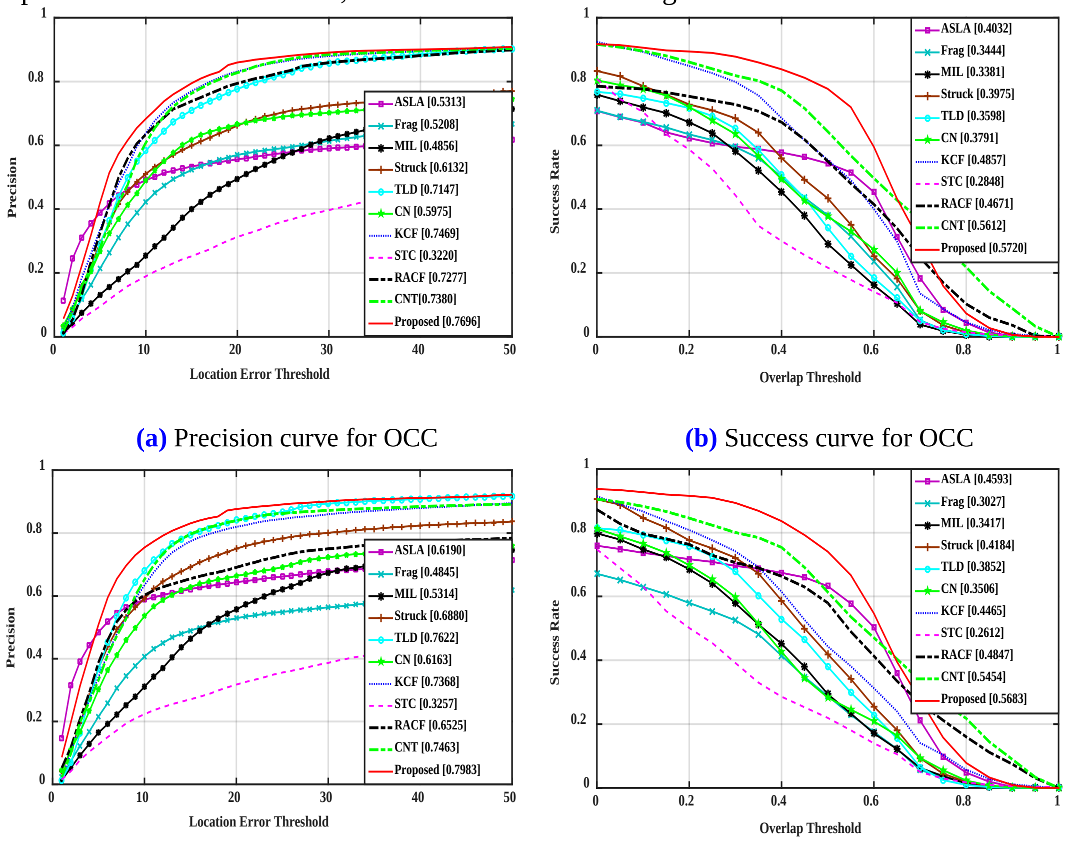
(b) Summary of the SRE based on area under the success curves

Fig. 8. Summary of Spatial Robustness Evaluation (SRE) of the trackers initialized with different size of bounding box with scale factor 0.8, 0.9, 1, 1.1 and 1.2. (a) Performance is compared using area under the precision curve over all the video sequences from VTBv1.0 database. (b) Performance is compared using area under the success curve over all the video sequences from VTBv1.0 database.

In MIL and TLD, the insertion of background pixels in target appearance drifts them during heavy occlusion and background clutters. Area under the precision curve in case of CN, KCF and the proposed method remains almost unaffected by target scale variations. It can also be observed that, STC, Frag, Struck, KCF, CNT and the proposed method show reduction in area under the success curve at lower scale factors. Whereas, at higher scale factors, the performance of ASLA, KCF and the proposed method either slightly increases or remains stable as that of the original scale (x1.0). At lower scaling factors, it is observed that the local patches in the proposed method drifts and becomes invalid for scale estimation which results in slightly reduced accuracy in the target scale estimation. The performance summary of all the tracking algorithms in terms of area under the precision and success curves against the bounding box scale is presented in **Fig. 8**. It is clear that the proposed algorithm outperforms all the contemporary published algorithms.

5.4 Attribute Specific Performance Evaluation (ASPE)

The performance of the trackers on the sequences with specific attributes is estimated using the annotated video sequences from VTBv1.0. It gives a subset of video sequences with respective dominant attributes, which indicate the challenges that a tracker will face while tracking the target in each sequence. As the enhancements are made primarily to meet the robustness against heavy occlusions and target scale variations; we test and compare the performance of the proposed method for occlusion (OCC) and target scale variations (SV). **Fig. 9** shows the precision and success curves for video sequence subsets attributing occlusion (OCC) and target scale variation (SV) each having 29 and 28 video sequences respectively. The areas under precision and success curves are shown in legends of the corresponding figures. It can be observed that, the proposed method in presence of OCC and SV outperforms all the state-of-the-art algorithms in terms of area under the precision and the success curves both. **Fig. 10** shows snapshots of the tracking results representing bounding boxes for the proposed algorithm and other three best performing trackers.



(a) Precision curve for OCC

(b) Success curve for OCC

(c) Precision curve for SV

(d) Success curve for SV

Fig. 9. Attribute Specific Performance Evaluation (ASPE). Performance of the trackers for the attributes OCC and SV. (a), (c) and (b), (d) show the precision and success curves respectively. The areas under the curves are indicated in legends of the respective figures.

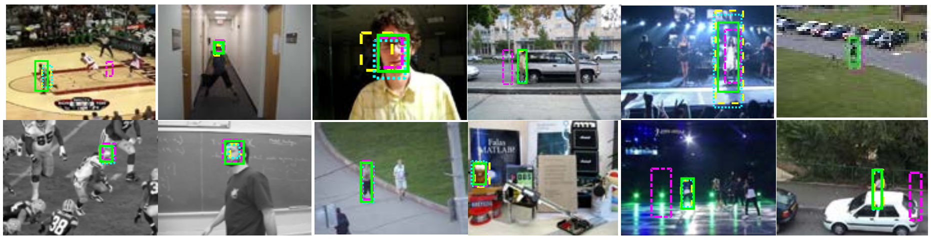


Fig. 10. Snapshots of the tracking results on different VBT v1.0 video frames. Each frame shows four bounding boxes for different state of the art algorithms including the proposed one. Bounding boxes' line styles are captioned as '---' for Color Names (CN), '....' for Kernelized Correlation filters (KCF), '.-.-.' for Tracking Learning Detection (TLD), '__ ' for the Proposed method.

6. Conclusion

The inherent lacunae in correlation filter based trackers are addressed by structurally enhancing the correlation filter. The heavy occlusions are handled using local appearance along with the global one. The taraget appearance is improved by embedding reverse RGB channels in its appearance. The final confidence map corresponding to the new target position is estimated as an adaptively weighted mean of the individual confidence maps of global and local patches. The adaptive weights have improved the reliability of tracking in case of heavy occlusions. Further, the relative displacement between the independent locations of global and local patches during tracking is used to calculate the scale of the target. The involvement of the local patch in scale estimation is governed by the validity flag which is set by two adaptive constraints ensuring reliable scale estimation. The OPE is presented for overall comparison of trackers on VTBv1.0 database, also the structural enhancements in correlation filter based tracker are extensively tested for their performance using video subsets for occlusion and scale variations from VTBv1.0 database.

References

[1] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834-1848, 2015. Article (CrossRef Link).

[2] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: an experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol. 36, no. 7, pp. 1442-1468, 2014. Article (CrossRef Link).

[3] S. Lucey, "Enforcing non-positive weights for stable support vector tracking," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska, USA, pp. 1-8, 2008. Article (CrossRef Link).

[4] X. Wang, G. Hua, and T. X. Han, "Discriminative Tracking by Metric Learning," in *Proc. of the European Conference on Computer Vision (ECCV)*, Berlin, Heidelberg, 2010. Article (CrossRef Link).

[5] Z. Zuo, G. Wang, B. Shuai, L. Zhao, Q. Yang, and X. Jiang, "Learning discriminative and shareable features for scene classification," in *Proc. of European Conference on Computer Vision (ECCV)*, Zurich,Switzerland., pp. 552-568, 2014. Article (CrossRef Link).

[6] S. Avidan, "Ensemble Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 261-271, 2007. Article (CrossRef Link).

[7] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619-1632, 2011. Article (CrossRef Link).

[8] K. Junseok and L. Kyoung Mu, "Tracking by Sampling Trackers," in *Proc. of IEEE International Conference on Computer Vision (ICCV)* Barcelona, Spain, pp. 1195-1202, 2011. Article (CrossRef Link).

[9] H. Weiming, L. Xi, L. Wenhan, Z. Xiaoqin, S. Maybank, and Z. Zhongfei, "Single and Multiple Object Tracking Using Log-Euclidean Riemannian Subspace and Block-Division Appearance Model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 12, pp. 2420-2440, 2012. Article (CrossRef Link).

[10] T. Liu, G. Wang, L. Wang, and K. L. Chan, "Visual Tracking via Temporally Smooth Sparse Coding," *IEEE Signal Processing Letters*, vol. 22, no. 9, pp. 1452-1456, 2015. Article (CrossRef Link).

[11] J. Xing, J. Gao, B. Li, W. Hu, and S. Yan, "Robust object tracking with online multi-lifespan dictionary learning," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, Sydney, Australia, pp. 665-672, 2013. Article (CrossRef Link).

[12] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," in *Proc. of European Conference on Computer Vision*, pp. 864-877, 2012. Article (CrossRef Link).

[13] S. Hare, A. Saffari, and P. H. Torr, "Struck: Structured output tracking with kernels," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, pp. 263-270, 2011. Article (CrossRef Link).

[14] Z. Kalal, J. Matas, and K. Mikolajczyk, "Pn learning: Bootstrapping binary classifiers by structural constraints," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, pp. 49-56, 2010. Article (CrossRef Link).

[15] D. Chen, Z. Yuan, G. Hua, J. Wang, and N. Zheng, "Multi-timescale Collaborative Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1-1, 2016. Article (CrossRef Link).

[16] Z. Chen, Z. Hong, and D. Tao, "An Experimental Survey on Correlation Filter-based Tracking," *arXiv preprint arXiv:1509.05520*, 2015. Article (CrossRef Link).

[17] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, pp. 2544-2550, 2010. Article (CrossRef Link).

[18] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. of European Conference on Computer Vision (ECCV)* ed Florence, Italy: Springer Berlin Heidelberg, pp. 702-715, 2012. Article (CrossRef Link).

[19] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, Ohio, pp. 1090-1097, 2014. Article (CrossRef Link).

[20] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. of IEEE Conference on Computer vision and pattern recognition (CVPR)* Portland, OR, pp. 2411-2418, 2013. Article (CrossRef Link).

[21] L. Cehovin, M. Kristan, and A. Leonardis, "An adaptive coupled-layer visual model for robust visual tracking," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, pp. 1363-1370, 2011. Article (CrossRef Link).

[22] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental Learning for Robust Visual Tracking," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125-141, 2008/05/01 2008. Article (CrossRef Link).

[23] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* Rhode Island, pp. 1815-1821, 2012. Article (CrossRef Link).

[24] B. Yang and R. Nevatia, "Online Learned Discriminative Part-Based Appearance Models for Multi-human Tracking," in *Proc. of European Conference on Computer Vision (ECCV)* A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds., ed Florence, Heidelberg: Springer Berlin Heidelberg, pp. 484-498, 2012. Article (CrossRef Link).

[25] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. of IEEE Conference on Computer vision and pattern recognition (CVPR)*, Providence, RI, pp. 1822-1829, 2012. Article (CrossRef Link).

[26] T. Liu, G. Wang, and Q. Yang, "Real-time part-based visual tracking via adaptive correlation filters," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, Massachusetts, pp. 4902-4912, 2015. Article (CrossRef Link).

[27] Y. Li, J. Zhu, and S. C. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, Massachusetts, pp. 353-361, 2015. Article (CrossRef Link)

[28] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, 2002. Article (CrossRef Link).

[29] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. of European Conference on Computer Vision (ECCV)*, Zurich, pp. 254-265, 2014. Article (CrossRef Link).

[30] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. of British Machine Vision Conference (BMVC)* Nottingham, September 1-5, 2014. Article (CrossRef Link).

[31] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, "Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, Massachusetts, pp. 749-758, 2015. Article (CrossRef Link).

[32] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, Massachusetts, pp. 5388-5396, 2015. Article (CrossRef Link).

[33] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *Proc. of European Conference on Computer Vision*, Zurich, pp. 127-141, 2014. Article (CrossRef Link).

[34] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-Speed Tracking with Kernelized Correlation Filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583-596, 2015. Article (CrossRef Link).

[35] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627-1645, 2010. Article (CrossRef Link).

[36] R. Rifkin, G. Yeo, and T. Poggio, "Regularized least-squares classification," *Nato Science Series Sub Series III Computer and Systems Sciences*, vol. 190, pp. 131-154, 2003. Article (CrossRef Link).

[37] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*: MIT press, 2002. Article (CrossRef Link).

[38] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-Learning-Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409-1422, 2012. Article (CrossRef Link).

[39] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. of IEEE Conference on Computer vision and pattern recognition (CVPR)*, New York, NY, USA., pp. 798-805, 2006. Article (CrossRef Link).

[40] K. Zhang, Q. Liu, Y. Wu, and M. H. Yang, "Robust Visual Tracking via Convolutional Networks Without Training," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1779-1792, 2016. Article (CrossRef Link)

Mayur Rajaram Parate received his B.Eng. degree in Electronics and Tele-communication Engineering and M.Eng. degree in Digital Electronics from SGBAU, Amravati, India in 2009 and 2013 respectively. He is currently pursuing his Ph.D. degree from Visvesvaraya National Institute of Technology, Nagpur, India. His areas of interest are image processing, computer vision and embedded system design.

Kishor M. Bhurchandi received his B.Eng. and M. Eng. degrees in electronics engineering in 1990 and 1992. He further obtained his Ph.D. degree from Visvesvaraya Regional College of Engineering, Nagpur University, Nagpur, India in 2002, where he is currently working as a Professor. He is principal investigator of two major funded research projects in the field of signal processing and embedded systems. He has more than 50 publications to his credit. He is the co-author of a popular book titled 'Advanced Microprocessors and Peripherals' published by McGraw Hill, India. His research interests include color image processing, computer vision, digital signal processing.