IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Structured group local sparse tracker

ISSN 1751-9659
Received on 30th November 2018
Revised 8th March 2019
Accepted on 1st April 2019
E-First on 6th June 2019
doi: 10.1049/iet-ipr.2018.6578
www.ietdl.org

Mohammadreza Javanmardi[1] ✉, *Xiaojun Qi*[1]

[1]*Department of Computer Science, Utah State University, Logan, UT 84322-4205, USA*

✉ *E-mail: javanmardi@aggiemail.usu.edu*

Abstract: Sparse representation is considered as a viable solution to visual tracking. In this study, the authors proposed a structured group local sparse tracker (SGLST), which exploits the local patches inside target candidates in the particle filter framework. Unlike the conventional local sparse trackers, the proposed optimisation model in SGLST not only adopts local and spatial information of the target candidates but also attains the spatial layout structure among them by employing a group-sparsity regularisation term. To solve the optimisation model, the authors proposed an efficient numerical algorithm consisting of two subproblems with the closed-form solutions. Both qualitative and quantitative evaluations on the benchmarks of challenging image sequences demonstrate the superior performance of the proposed tracker against several state-of-the-art trackers.

1 Introduction

Visual tracking is the process of estimating states of a moving object in a dynamic frame sequence. It has been considered as one of the most paramount and challenging topics in computer vision with various applications in human motion analysis, surveillance, smart vehicles transportation, navigation, etc. Although numerous tracking methods [1–8] have been introduced in the recent years, developing a robust algorithm that can handle different challenges such as occlusion (OCC), illumination variations (IVs), deformation (DEF), fast motion (FM), camera motion, and background clutter (BC) still remains unsolved.

Visual tracking algorithms can be roughly classified into discriminative and generative categories. Discriminative approaches cast the tracking problem as a binary classification and formulate a decision boundary to separate the target from backgrounds. Representative discriminative approaches include the ensemble tracker [9], the online boosting [10, 11], the multiple instance learning [12], the PN-learning [13], and correlation filter (CF)-based trackers [14–16]. In contrast, generative approaches adopt a model to represent the target and cast the tracking as a searching procedure to find the most similar region to the target model. Representative generative tracking methods include Eigentracking [17], mean shift [18], Frag-track [19], incremental learning [20], visual tracking decomposition [21], and adaptive colour tracking [22].

Sparse representation-based trackers (sparse trackers) are considered as generative tracking methods, as they sparsely express the target candidates using a few templates (bases). Generally, sparse representation has played a dominant role in computer vision applications such as face recognition [23], image denoising and restoration [24], image segmentation [25, 26], image pansharpening [27], etc. Most sparse trackers utilise a convex optimisation model to represent the global appearance of target candidates in the particle filter framework. As one of the pioneer work, Mei and Ling[28] represented the global information of target candidates by a set of templates using ℓ_1 minimisation. Bao *et al.* [29] presented an accelerated proximal gradient descent method to increase the efficiency of solving ℓ_1 minimisation. In order to obtain the relationship among target candidates, Zhang *et al.* [30] proposed to jointly learn the global information of all target candidates. Later, Hong *et al.* [31] cast tracking as a multi-task multi-view sparse learning problem in terms of the least square. To handle the data possibly contaminated by outliers and noise, Mei *et al.* [32] used the least absolute deviation in their optimisation model. In general, these global sparse trackers achieve good performance. However, they model each target region as a single entity and may fail when targets undergo heavy OCCs in a frame sequence.

Unlike global sparse trackers, local sparse trackers represent local patches inside target candidates together with local patches inside each template set. Liu *et al.* [33] introduced a local sparse tracker, which adopts the histogram of sparse coefficients and a sparse constrained regularised mean-shift algorithm, to robustly track the object. This method is based on a static local sparse dictionary and therefore, fails in the cases when similar objects appear in the scene. Jia *et al.* [34] exploit both partial and spatial information of target candidates and represent them in a dynamic local sparse dictionary. Recently, Jia *et al.* [35] proposed trackers to extract coarse and fine local image patches inside each target candidate. Despite the favourable performance, these local sparse trackers [34, 35] do not consider the spatial layout structure among local patches inside a target candidate. As a result, the sparse vectors of local patches exhibit a random pattern rather than a similar structure on the non-zero elements.

To further improve the tracking performance, recent sparse trackers consider both global and local information of all target candidates in their optimisation models. Zhang *et al.* [5] represented local patches inside all the target candidates along with the global information using $\ell_{1,2}$ norm regularisation on the sparse representative matrix. They assume that the same local patches of all target candidates are similar. However, this assumption does not hold well in practice due to outlier candidates and OCC in tracking. To address this shortcoming, Zhang *et al.* [36] took into account both factors to design an optimal target region searching method. These recent sparse trackers achieve improved performance. However, considering the relationship of all target candidates, degrades the performance when drifting occurs. In addition, using $\ell_{1,2}$ norm regularisation in the optimisation model to integrate both local and global information of target candidates lessens the tracking accuracy in the cases of heavy OCCs.

In this study, we propose a structured group local sparse tracker (SGLST), which exploits the local patches inside a target candidate and represent them in a novel convex optimisation model. The proposed optimisation model not only adopts local and spatial information of the target candidates but also attains the spatial layout structure among them by employing a group-sparsity regularisation term. The main contributions of the proposed work are summarised as follows:

- Proposing a local sparse tracker, which employs local and spatial information of the target candidate and attains the spatial structure among different local patches inside a target candidate.
- Developing a convex optimisation model, which introduces a group-sparsity regularisation term to motivate the tracker to select the corresponding local patches of the same small subset of templates to represent the local patches of each target candidate.
- Designing a fast and parallel numerical algorithm based on the alternating direction method of multiplier (ADMM), which consists of two subproblems with the closed-form solutions.

The remainder of this study is organised as follows: Section 2 introduces the notations. Section 3 presents the SGLST together with its novel convex optimisation model solved by the proposed ADMM-based numerical solution. Section 4 demonstrates the experimental results on 16 publicly available challenging image sequences, the OTB50 and the OTB100 tracking benchmarks and compares the SGLST with several state-of-the-art trackers. Section 5 draws the conclusions.

2 Notations

Throughout this study, matrices, vectors, and scalars are denoted by bolditalic uppercase, bolditalic lowercase, and italic lowercase letters, respectively. For a given matrix $\boldsymbol{X}$, $\boldsymbol{X}_{i,j}$ is the element at the ith row and jth column, $\| \boldsymbol{X} \|_{\mathrm{F}}$ is the Frobenius norm, $\| \boldsymbol{X} \|_{p,q}$ is the ℓ_p norm of ℓ_q norm of the rows in $\boldsymbol{X}$, and $\boldsymbol{X}(:)$ is the vectorised form of $\boldsymbol{X}$. For a given column vector $\boldsymbol{x}$, $\mathrm{diag}(\boldsymbol{x})$, and $\boldsymbol{x}_i$ are the diagonal matrix formed by the elements of $\boldsymbol{x}$ and the ith element of $\boldsymbol{x}$, respectively. The symbol $\mathrm{tr}(\cdot)$ stands for the trace operator, $\boldsymbol{X} \otimes \boldsymbol{Y}$ is the Kronecker product on two matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ of arbitrary sizes, $\boldsymbol{1}_l$ is the column vector of all ones with the dimension of l, and $\boldsymbol{I}_k$ is the $k \times k$ identity matrix.

3 Proposed method

This section provides detailed information about the proposed SGLST method. Specifically, Section 3.1 formulates a local sparse appearance model in SGLST and explains how this convex optimisation model addresses the drawbacks of conventional local sparse trackers [34, 35]. Section 3.2 presents an efficient numerical algorithm to solve the convex optimisation problem presented in Section 3.1.

3.1 Structured group local sparse tracker

The proposed SGLST utilises both local and spatial information in the particle filter framework and employs a new optimisation model, which addresses the drawback of conventional local sparse trackers by attaining the spatial layout structure among different local patches inside a target candidate.

Conventional local sparse trackers [34, 35], individually represent local patches without considering their spatial layout structure. For instance, local patches in [34] are separately represented by solving the Lasso problem. As a consequence, local patches inside the jth target candidate may be sparsely represented by the corresponding local patches inside *different* dictionary templates, as illustrated in Fig. 1*a*, two local patches of the jth target candidate, shown in the red and blue bounding boxes, may be represented by the corresponding local patches in different dictionary templates.

In this study, we propose a novel SGLST that adopts both local and spatial information of the target candidates for tracking. The proposed tracker employs a novel optimisation model to solve the aforementioned issues associated with conventional local sparse trackers [34, 35]. Specifically, SGLST formulates an optimisation problem to impose a structure on the achieved sparse vectors for different local patches inside each target candidate and attain the spatial layout structure among the local patches. To solve the proposed model, we developed an efficient numerical algorithm consisting of two subproblems with closed-form solutions by adopting the ADMM within each target candidate in the optimisation function. To maintain the spatial layout structure among local patches, we jointly represent all the local patches of a target candidate in a new convex optimisation model. In other words, if the rth local patch of the jth target candidate is best represented by the rth local patch of the qth template, the sth local patch of the jth target candidate should also be best represented by the sth local patch of the qth template. As shown in Fig. 1*b*, we aim to represent both local patches of the jth target candidate, shown in the red and blue bounding boxes, by their corresponding patches in the *same* dictionary templates (e.g. the first and the tenth templates).

To do so, we first use k target templates and extract l overlapping d dimensional local patches inside each template to construct the dictionary $\boldsymbol{D}$. Such a representation generates the local dictionary matrix $\boldsymbol{D} = [\boldsymbol{D}_1, \ldots, \boldsymbol{D}_k] \in \mathbb{R}^{d \times (lk)}$, where $\boldsymbol{D}_i \in \mathbb{R}^{d \times l}$. Then, we construct a matrix $\boldsymbol{X} = [\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n] \in \mathbb{R}^{d \times (ln)}$, which contains the local patches of all the target candidates, where n is the number of particles. Next, we define the sparse matrix coefficients $\boldsymbol{C}$ corresponding to the jth target candidate as $\boldsymbol{C} \triangleq [\boldsymbol{C}_1 \ \cdots \ \boldsymbol{C}_k]^{\top} \in \mathbb{R}^{(lk) \times l}$, where $\{\boldsymbol{C}_q\}_{q=1}^{k}$ is a $l \times l$ matrix indicating the group sparse representation of l local patches of the jth target candidate using l local patches of the qth template. Finally, we formulate the following convex model:

$$\underset{\boldsymbol{C} \in \mathbb{R}^{(lk) \times l}}{\text{minimize}} \quad \| \boldsymbol{X}_j - \boldsymbol{D}\boldsymbol{C} \|_{\mathrm{F}}^2 + \lambda \| \boldsymbol{C}_1(:) \ldots \boldsymbol{C}_k(:)^{\top} \|_{1,\infty} \tag{1a}$$

$$\text{subject to} \quad \boldsymbol{C} \geq 0, \tag{1b}$$

$$\boldsymbol{1}_{lk}^{\top}\boldsymbol{C} = \boldsymbol{1}_l^{\top}, \tag{1c}$$

where the first term corresponds to the total cost of representing feature matrix $\boldsymbol{X}_j$ using the dictionary matrix $\boldsymbol{D}$ and the second

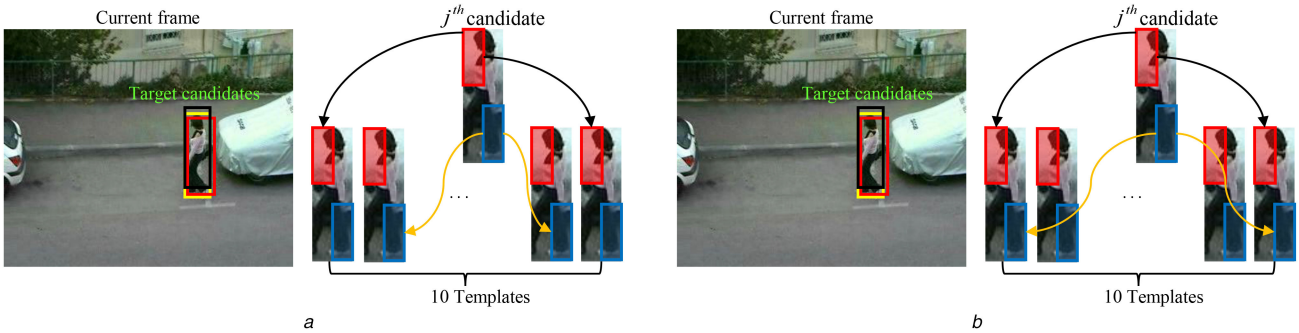

Fig. 1 *Illustration of the sparse representation of two sample local patches of the jth target candidate in*
(a) Conventional local sparse trackers [34, 35]. One local patch of the jth target candidate, shown in the red bounding box, is represented by its corresponding patch in the first and the tenth templates, while another local patch of this candidate, shown in the blue bounding box, is represented by its corresponding patch in two different templates (e.g. the second and the ninth templates), ***(b)*** The proposed SGLST. Both local patches of the jth target candidate, shown in red and blue bounding boxes, are represented by their corresponding patches in the same templates (e.g. the first and the tenth templates)

term is a group-sparsity regularisation term, which penalises the objective function in proportion with the number of selected templates (dictionary words). Moreover, the group-sparsity regularisation term imposes all the local patches to jointly select similar few templates by simultaneously establishing the $\|\cdot\|_{1,\infty}$ minimisation on matrix $\boldsymbol{C}$. To achieve this goal, the regularisation term is imposed to make the columns of $\boldsymbol{C}$ to be sparse (i.e. select few templates for representation) by performing the l_1 norm minimisation on the columns of $\boldsymbol{C}$. In addition, this regularisation term motivates the group of the local patches to jointly select similar few templates by imposing l_∞ on the rows of $\boldsymbol{C}$. The regularisation parameter $\lambda > 0$ balances the trade-off between the two terms. The constraint (1c) ensures that each local patch in $\boldsymbol{X}_j$ is expressed by at least one selected local patch of the dictionary $\boldsymbol{D}$ and the sum of a linear combination of coefficients is constrained.

For each target candidate, we find the sparse matrix $\boldsymbol{C}$ using the numerical algorithm as presented in Section 3.2. We then perform an averaging process along with an alignment pooling strategy [34] to find a representative vector. Finally, we calculate the summation of this representative vector as the likelihood value. The candidate with the highest likelihood value is selected as the tracking result. We also update the templates throughout the sequence using the same strategy as proposed in [34] to handle the appearance variations of the target region.

3.2 Numerical algorithm

This section presents a numerical algorithm based on the ADMM [37] to efficiently solve the proposed model (1). The idea of the ADMM is to utilise auxiliary variables to convert a complicated convex problem to smaller subproblems, where each one is efficiently solvable via an explicit formula. The ADMM iteratively solves the subproblems until convergence. To do so, we first define vector $\boldsymbol{m} \in \mathbb{R}^k$ such that $\boldsymbol{m}_i = \arg\max |\boldsymbol{C}_i(:)|$ and rewrite (1) as

$$\underset{\substack{\boldsymbol{C}\in\mathbb{R}^{(lk)\times l}\\ \boldsymbol{m}\in\mathbb{R}^k}}{\text{minimize}} \| \boldsymbol{X}_j - \boldsymbol{D}\boldsymbol{C} \|_{\mathrm{F}}^2 + \lambda \mathbf{1}_k^\top \boldsymbol{m} \tag{2a}$$

$$\text{subject to} \quad \boldsymbol{C} \geq 0, \tag{2b}$$

$$\mathbf{1}_{(lk)}^\top \boldsymbol{C} = \mathbf{1}_l^\top \tag{2c}$$

$$\boldsymbol{m} \otimes \mathbf{1}_l \mathbf{1}_l^\top \geq \boldsymbol{C}. \tag{2d}$$

It should be noted that the constraint (2d) is imposed in the above reformulation to ensure the equivalence between (1) and (2). This inequality constraint can be transformed into an equality one by introducing a non-negative slack matrix $\boldsymbol{U} \in \mathbb{R}^{(lk)\times l}$, which compensates for the difference between $\boldsymbol{m} \otimes \mathbf{1}_l \mathbf{1}_l^\top$ and $\boldsymbol{C}$. Using the resultant equality constraint, $\mathbf{1}_k^\top \boldsymbol{m}$ can be equivalently written as $(1/l^2)\mathbf{1}_{(lk)}^\top (\boldsymbol{C}+\boldsymbol{U})\mathbf{1}_l$. Moreover, this equality constraint implies that the columns of $\boldsymbol{C}+\boldsymbol{U}$ are regulated to be identical. Hence, one can simply replace it by a linear constraint independent of $\boldsymbol{m}$ as presented in (3d). Therefore, we rewrite (2) independent of $\boldsymbol{m}$ as

$$\underset{\boldsymbol{C},\boldsymbol{U}\in\mathbb{R}^{(lk)\times l}}{\text{minimize}} \| \boldsymbol{X}_j - \boldsymbol{D}\boldsymbol{C} \|_{\mathrm{F}}^2 + \frac{\lambda}{l^2}\mathbf{1}_{(lk)}^\top (\boldsymbol{C}+\boldsymbol{U})\mathbf{1}_l \tag{3a}$$

$$\text{subject to} \quad \boldsymbol{C} \geq 0, \tag{3b}$$

$$\mathbf{1}_{(lk)}^\top \boldsymbol{C} = \mathbf{1}_l^\top \tag{3c}$$

$$\boldsymbol{E}(\boldsymbol{C}+\boldsymbol{U}) = \frac{\boldsymbol{I}_k \otimes \mathbf{1}_l \mathbf{1}_l^\top}{l}(\boldsymbol{C}+\boldsymbol{U}) \tag{3d}$$

$$\boldsymbol{U} \geq 0, \tag{3e}$$

where matrix $\boldsymbol{E}$ serves as the right circular shift operator on the rows of $\boldsymbol{C}+\boldsymbol{U}$. To construct the ADMM formulation, whose subproblems possess closed-form solutions, we define auxiliary variables $\hat{\boldsymbol{C}}, \hat{\boldsymbol{U}} \in \mathbb{R}^{(lk)\times l}$ and reformulate (3) as

$$\begin{aligned}\underset{\boldsymbol{C},\hat{\boldsymbol{C}},\boldsymbol{U},\hat{\boldsymbol{U}}\in\mathbb{R}^{(lk)\times l}}{\text{minimize}} \quad & \| \boldsymbol{X}_j - \boldsymbol{D}\boldsymbol{C} \|_{\mathrm{F}}^2 + \frac{\lambda}{l^2}\mathbf{1}_{(lk)}^\top (\boldsymbol{C}+\boldsymbol{U})\mathbf{1}_l \\ & + \frac{\mu_1}{2}\| \boldsymbol{C} - \hat{\boldsymbol{C}} \|_{\mathrm{F}}^2 + \frac{\mu_2}{2}\| \boldsymbol{U} - \hat{\boldsymbol{U}} \|_{\mathrm{F}}^2\end{aligned} \tag{4a}$$

$$\text{subject to} \quad \hat{\boldsymbol{C}} \geq 0, \tag{4b}$$

$$\mathbf{1}_{(lk)}^\top \hat{\boldsymbol{C}} = \mathbf{1}_l^\top \tag{4c}$$

$$\boldsymbol{E}(\boldsymbol{C}+\boldsymbol{U}) = \frac{\boldsymbol{I}_k \otimes \mathbf{1}_l \mathbf{1}_l^\top}{l}(\boldsymbol{C}+\boldsymbol{U}), \tag{4d}$$

$$\hat{\boldsymbol{U}} \geq 0, \tag{4e}$$

$$\boldsymbol{C} = \hat{\boldsymbol{C}}, \quad \boldsymbol{U} = \hat{\boldsymbol{U}}. \tag{4f}$$

where $\mu_1, \mu_2 > 0$ are the augmented Lagrangian parameters. Without loss of generality, we assume $\mu = \mu_1 = \mu_2$ [37]. The last two terms in the objective function (4a) are later vanished for any feasible solutions, which implies (3) and (4) are equivalent. We further configure the augmented Lagrangian function to solve (4) as follows: (see (5)) , where $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2 \in \mathbb{R}^{(lk)\times l}$ are the Lagrangian multipliers corresponding to the equations in (4f).

Given initialisation for $\hat{\boldsymbol{C}}$, $\hat{\boldsymbol{U}}$, $\boldsymbol{\Lambda}_1$, and $\boldsymbol{\Lambda}_2$ at the time $t = 0$ (e.g. $\hat{\boldsymbol{C}}^0, \hat{\boldsymbol{U}}^0, \boldsymbol{\Lambda}_1^0, \boldsymbol{\Lambda}_2^0$), (5) is solved through the ADMM iterations. At the next iteration, $\boldsymbol{C}$ and $\boldsymbol{U}$ are updated by minimising (5) under the constraint (4d). To do so, we first define $\{z_i\}_{i=1}^{lk}$, where $z_i \in \mathbb{R}^{2l}$ is obtained by stacking the ith rows of $\boldsymbol{C}$ and $\boldsymbol{U}$. We then divide this minimisation problem into lk equality constrained quadratic programs, where each program has its analytical solution. Using the updated $\boldsymbol{C}$ and $\boldsymbol{U}$, we compute $\hat{\boldsymbol{C}}$ and $\hat{\boldsymbol{U}}$ by minimising (5) with the constraints (4b), (4c), and (4e). To this end, we split the problem into two separate subproblems with closed-form solutions over $\hat{\boldsymbol{C}}$ and $\hat{\boldsymbol{U}}$, where the first subproblem consists of l independent Euclidean norm projections onto the probability simplex constraints and the second subproblem consists of l independent Euclidean norm projections onto the non-negative orthant. Finally, we update $\boldsymbol{\Lambda}_1$ and $\boldsymbol{\Lambda}_2$ by performing l parallel updates over their respective columns. All these iterative updates can be quickly performed due to closed-form solutions.

4 Experimental results

In this section, we evaluate the performance of the proposed SGLST on 16 publicly available frame sequences, the OTB50 [38] and the OTB100 [39] tracking benchmarks.

We resize each target region to 32×32 pixels and extract overlapping local patches of 16×16 pixels inside the target region using the step size of 8 pixels. This leads to $l = 9$ local patches. For each local patch, we extract two sets of features, namely, grey-level intensity features and histogram of oriented gradients (HOG)

$$\begin{aligned}\mathscr{L}_\mu(\boldsymbol{C},\boldsymbol{U},\hat{\boldsymbol{C}},\hat{\boldsymbol{U}},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2) = & \| \boldsymbol{X}_j - \boldsymbol{D}\boldsymbol{C} \|_{\mathrm{F}}^2 + \frac{\lambda}{l^2}\mathbf{1}_{(lk)}^\top (\boldsymbol{C}+\boldsymbol{U})\mathbf{1}_l \\ & + \frac{\mu}{2}\| \boldsymbol{C} - \hat{\boldsymbol{C}} + \frac{\boldsymbol{\Lambda}_1}{\mu} \|_{\mathrm{F}}^2 + \frac{\mu}{2}\| \boldsymbol{U} - \hat{\boldsymbol{U}} + \frac{\boldsymbol{\Lambda}_2}{\mu} \|_{\mathrm{F}}^2,\end{aligned} \tag{5}$$

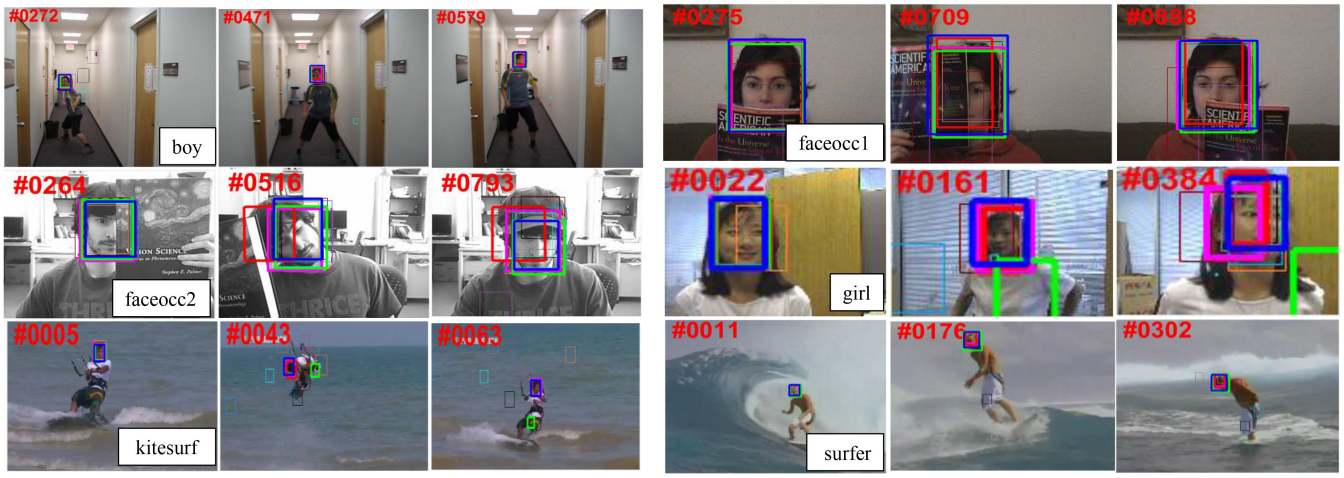

Fig. 2 *Comparison of the tracking results of 11 state-of-the-art trackers and the two variants of the proposed SGLST on boy, faceocc1, faceocc2, girl, kitesurf, and surfer image sequences. Frame indices are shown at the top left corner of representative frames. Results are best viewed on high-resolution displays. The tracking results of the top four trackers (i.e. SGLST_HOG, RSST_HOG, SLGST_Color, and MEEM) are highlighted by thicker lines*
(—L1T, - - -Struck, —IVT, —MTT, —MIL, —VTD, —Frag, —ASLA, - - -KCF, —MEEM, —RSST_HOG, —SGLST_Color, —SGLST_HOG)

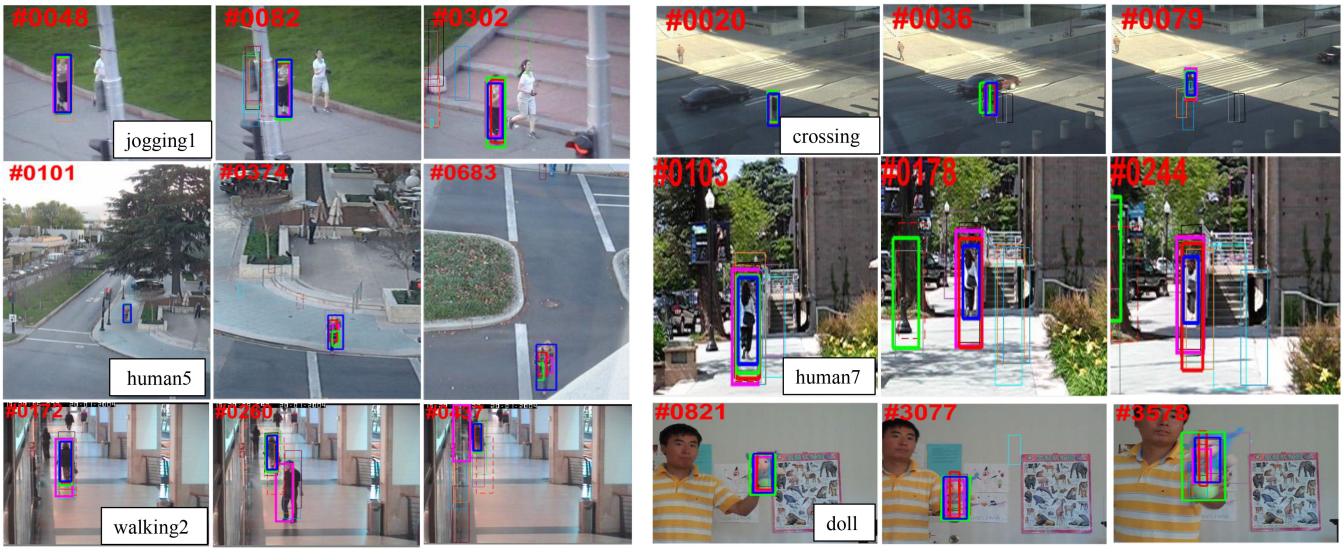

Fig. 3 *Comparison of the tracking results of 11 state-of-the-art trackers and the two variants of the proposed SGLST on jogging1, crossing, human5, human7, walking2, and doll image sequences. Frame indices are shown at the top left corner of representative frames. Results are best viewed on high-resolution displays. The tracking results of the top four trackers (i.e. SGLST_HOG, RSST_HOG, SLGST_Color, and MEEM) are highlighted by thicker lines*
(—L1T, - - -Struck, —IVT, —MTT, —MIL, —VTD, —Frag, —ASLA, - - -KCF, —MEEM, —RSST_HOG, —SGLST_Color, —SGLST_HOG)

features, to represent its characteristics from two perspectives. Both features have shown promising tracking results in different trackers and HOG features [40] have demonstrated significant improvement in visual tracking [32, 36, 41]. The proposed SGLST, therefore, has two variants: SGLST_Color and SGLST_HOG. For the HOG features, we resize the target candidates to 64×64 pixels and exploit 196 dimensional HOG features for each of the 32×32 local patches to capture relatively high-resolution edge information. For all the experiments, we set $\lambda = 0.1$, $\mu_1 = \mu_2 = \mu = 0.1$, the number of particles $n = 400$, and the number of target templates $k = 10$. We adopt the same setting as used in [34] to update templates.

4.1 Experimental results on publicly available sequences

We conducted extensive experiments on 16 publicly available challenging frame sequences, which are used to evaluate trackers including IVT [20], VTD [21], L1T [28], and RSST [36]. These sequences contain various challenges such as FM, OCC, DEF, and scale variation (SV) and therefore, are commonly used to evaluate the qualitative performance of different trackers. We compare SGLST_Color and SGLST_HOG with 11 state-of-the-art trackers, namely, L1T [28], Struck [42], IVT [20], MTT [43], MIL [44], VTD [21], Frag [19], ASLA [34], KCF [45], MEEM [46], and RSST_HOG [36]. To ensure a fair comparison, we use the available source code or the binary code together with the optimal parameters provided by the respective authors to produce the tracking results. Figs. 2–4 demonstrate the tracking results of the 13 aforementioned methods compared on three representative frames of each of the 16 sequences. The tracking results of the top four trackers (i.e. SGLST_HOG, RSST_HOG, SLGST_Color, and MEEM) are highlighted by thicker lines.

Here, we briefly analyse the tracking performance of each compared tracker under different challenging scenarios. The L1T tracker fails when the target undergoes FM and rotation as shown in *Kitesurf* and *surfer* sequences, OCC as shown in the *jogging1* sequence, or SV as shown in the *board* sequence. Struck cannot track the target when OCC (*jogging1* and *box*) or FM (*surfer*) occurs. IVT drifts from the target in the frame sequences containing the out-of-view (OV) challenge (*girl* and *jogging*), FM (*boy*), or SV (*human5*). MTT loses the target having large motions between consecutive frames (*board* and *crossing*). MIL fails to track the target when there is SV (*car4* and *car2*) or when OCC (*walking2*) happens. VTD and Frag lead to the drift of the target under FM and DEF circumstances as shown in *crossing* and *human7* sequences. In addition, they cannot adequately handle SV as shown in the *box* sequence. ASLA does not yield good performance in the cases of heavy OCCs (*faceocc1*, *jogging*, and

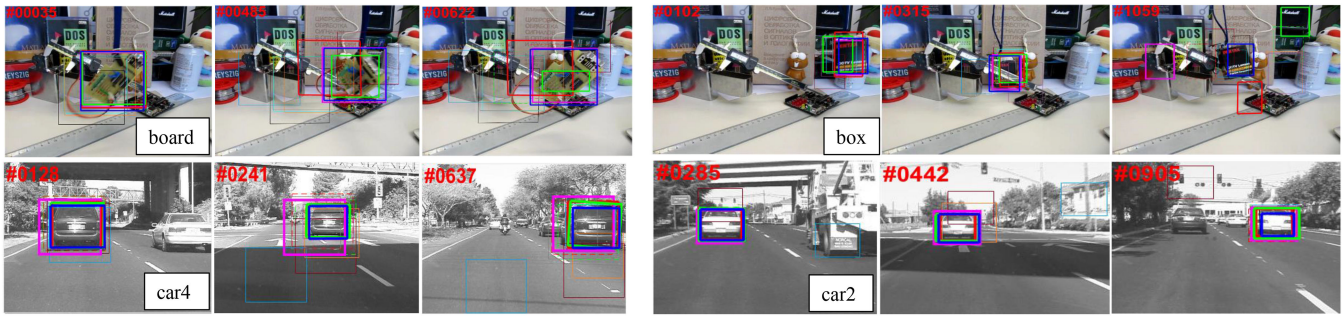

Fig. 4 *Comparison of the tracking results of 11 state-of-the-art trackers and the two variants of the proposed SGLST on board, box, car4, and car2 image sequences. Frame indices are shown at the top left corner of representative frames. Results are best viewed on high-resolution displays. The tracking results of the top four trackers (i.e. SGLST_HOG, RSST_HOG, SLGST_Color, and MEEM) are highlighted by thicker lines*
(—L1T, - - -Struck, —IVT, MTT, —MIL, —VTD, —Frag, —ASLA, - - -KCF, —MEEM, —RSST_HOG, —SGLST_Color, —SGLST_HOG)

Table 1 Summary of the average overlap scores of 13 compared methods on 16 sequences

Seq	L1T	Struck	IVT	MTT	MIL	VTD	Frag	ASLA	KCF	MEEM	RSST_HOG	SGLST_Color	SGLST_HOG
boy	0.73	0.76	0.26	0.49	0.49	0.62	0.38	0.36	0.77	*0.79*	0.76	**0.81**	0.78
faceocc1	0.74	0.73	0.72	0.70	0.60	0.69	0.82	0.41	*0.76*	0.75	0.70	0.74	**0.78**
faceocc2	0.68	*0.76*	0.72	0.74	0.67	0.71	0.65	0.65	0.74	**0.78**	0.67	0.75	0.72
girl	0.73	*0.74*	0.16	0.66	0.39	0.60	0.43	0.72	0.58	0.69	**0.75**	0.21	0.70
kitesurf	0.25	0.64	0.30	0.35	0.31	0.15	0.19	0.25	0.48	*0.67*	**0.71**	0.22	0.47
surfer	0.04	0.41	0.06	0.10	0.25	0.31	0.22	0.41	0.47	0.51	0.61	**0.71**	*0.65*
jogging1	0.14	0.17	0.17	0.17	0.18	0.21	0.52	0.22	0.25	0.67	0.71	**0.78**	*0.74*
crossing	0.24	0.67	0.29	0.19	0.72	0.31	0.31	*0.77*	0.71	0.71	0.76	0.72	**0.79**
human5	0.38	0.35	0.18	0.45	0.21	0.28	0.03	*0.68*	0.21	0.28	0.51	0.35	**0.72**
human7	0.51	0.48	0.23	0.28	0.48	0.28	0.27	0.29	0.29	0.48	*0.58*	0.40	**0.83**
walking2	0.75	0.51	0.79	0.78	0.29	0.40	0.35	0.37	0.39	0.31	0.76	*0.80*	**0.83**
doll	0.46	0.55	0.43	0.39	0.42	0.66	0.61	*0.78*	0.59	0.60	0.48	**0.84**	0.75
board	0.13	0.66	0.19	0.19	0.40	0.28	0.52	0.30	0.65	*0.68*	0.42	0.56	**0.75**
box	*0.55*	0.21	0.51	0.21	0.27	0.42	0.46	0.34	0.35	0.31	0.33	0.21	**0.62**
car4	0.72	0.48	0.82	0.75	0.25	0.36	0.19	0.75	0.48	0.45	**0.87**	0.82	*0.85*
car2	0.86	0.68	0.89	0.87	0.16	0.80	0.25	0.85	0.68	0.68	**0.91**	0.81	*0.88*
average	0.49	0.55	0.42	0.46	0.38	0.44	0.39	0.51	0.53	0.59	*0.66*	0.61	**0.74**

The bold numbers indicate the best performance, while the italic numbers indicate the second best.

walking2). KCF is incapable of dealing with SV (*car4* and *walking2*), OCC *(jogging1)*, or OV challenges (*box*). MEEM achieves good overall performance. However, it drifts from the target when the scale varies (*car4*) and does not sufficiently address the challenge of partial OCC (*walking2* and *box*). RSST_HOG performs well in most sequences, but it drifts away in the sequences with SVs (*doll*, *board*, and *box*). SGLST_Color also demonstrates favourable performance in most of the sequences. However, it encounters problems when illumination changes happen (*kitesurf* and *box*). Among all the compared methods, SGLST_HOG performs well in tracking human faces, human bodies, objects, and vehicles in the 16 challenging sequences. The favourable performance of the proposed SGLST reflects the advantages of adopting local patches within the target and keeping the spatial structure among local patches. In addition, using HOG features in SGLST helps to improve the tracking performance yielded by using intensity features.

For quantitative comparison, we compute the average overlap score across all frames of each image sequence for each compared method. It is worth mentioning that the overlap score between the tracked bounding box r_t and the ground truth bounding box r_g is defined as $S = (|r_t \cap r_g| / |r_t \cup r_g|)$, where $|\cdot|$ is the number of pixels in the bounding box, $\cap$ represents the intersection of the two bounding boxes, and $\cup$ represents the union of the two bounding boxes. Table 1 summarises the average overlap scores across all frames of each of the 16 sequences for the compared methods. It is clear that the two proposed trackers, SGLST_Color and SGLST_HOG, achieve overall favourable tracking performance for the tested sequences. On average, SGLST_Color drastically improves the average overlap scores of L1T, IVT, MTT, MIL, VTD, and Frag by 24.49, 45.24, 32.61, 60.53, 38.64, and 56.41%, respectively. It also outperforms Struck, ASLA, KCF, and MEEM by improving their average overlap scores by 10.91, 19.61, 15.09, and 3.39%, respectively. RSST_HOG is the only tracker that outperforms SGLST_Color by 8.2% mainly due to the use of HOG features. The proposed SGLST_HOG achieves the best average overlap score and significantly outperforms SGLST_Color and RSST_HOG by 21.31 and 12.12%, respectively. In summary, the qualitative results shown in Figs. 2–4 and the quantitative results shown in Table 1 demonstrate that SGLST_HOG achieves the best tracking performance and SGLST_Color achieves the third best tracking performance, inferior to RSST_HOG that uses HOG features instead of intensity features. Both variants of the proposed SGLST can successfully track the targets in a majority of frames in all 16 tested sequences with different challenging conditions such as FM, rotation and SVs, OCCs, and illumination changes.

4.2 Experimental results on the OTB50 benchmark

We conduct the experiments on the OTB50 tracking benchmark [38] to evaluate the overall performance of the proposed SGLST_Color and SGLST_HOG under different challenges. This benchmark consists of 50 annotated sequences, where 49 sequences have one annotated target and one sequence (*jogging*) has two annotated targets. Each sequence is also labelled with attributes specifying the presence of different challenges including IV, SV, OCC, DEF, motion blur (MB), FM, in-plane rotation (IPR), out-of-plane rotation (OPR), OV, BC, and low resolution (LR). The sequences are categorised based on the attributes and 11 challenge subsets are generated. These subsets are utilised to evaluate the performance of trackers in different challenge categories.

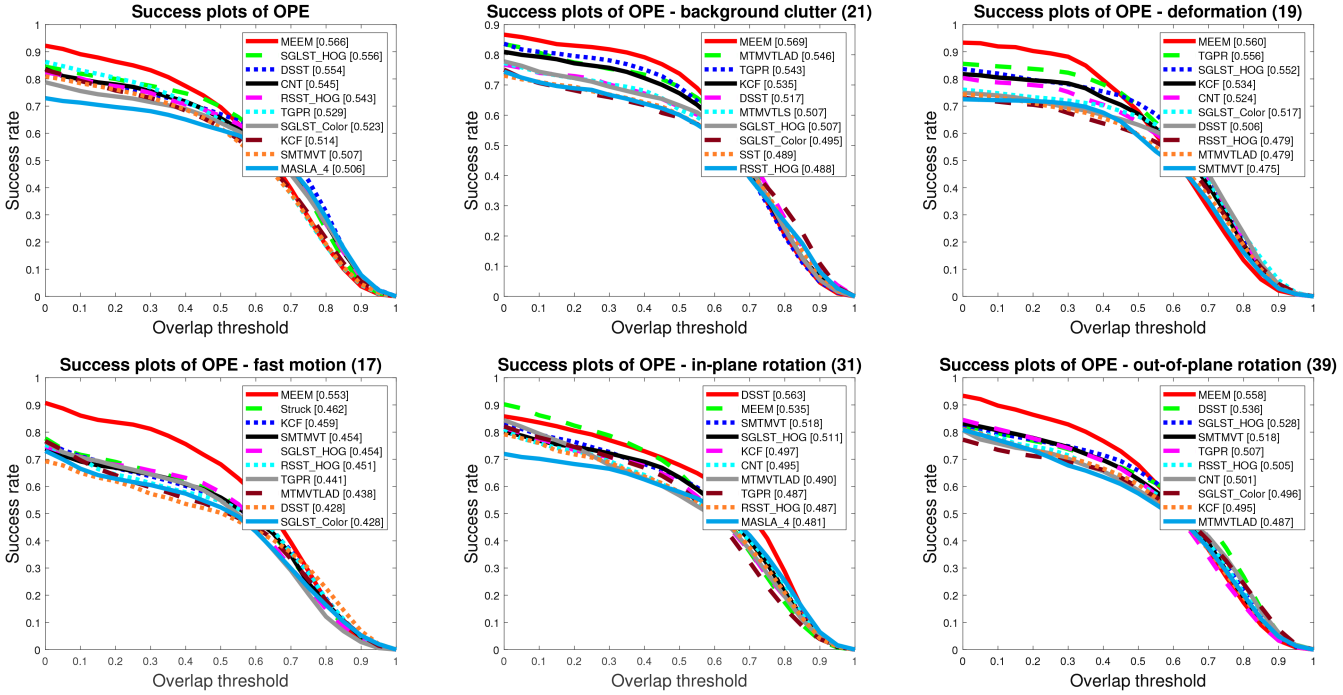

Fig. 5 *OTB50 overall OPE success plots and the OPE success plot BC, DEF, FM, IPR, and OPR challenge subsets. The value appearing in the title is the number of sequences in the specific subset. The values appearing in the legend are the AUC scores. Only the top ten trackers are presented, while the results of the other trackers can be found in [38]*

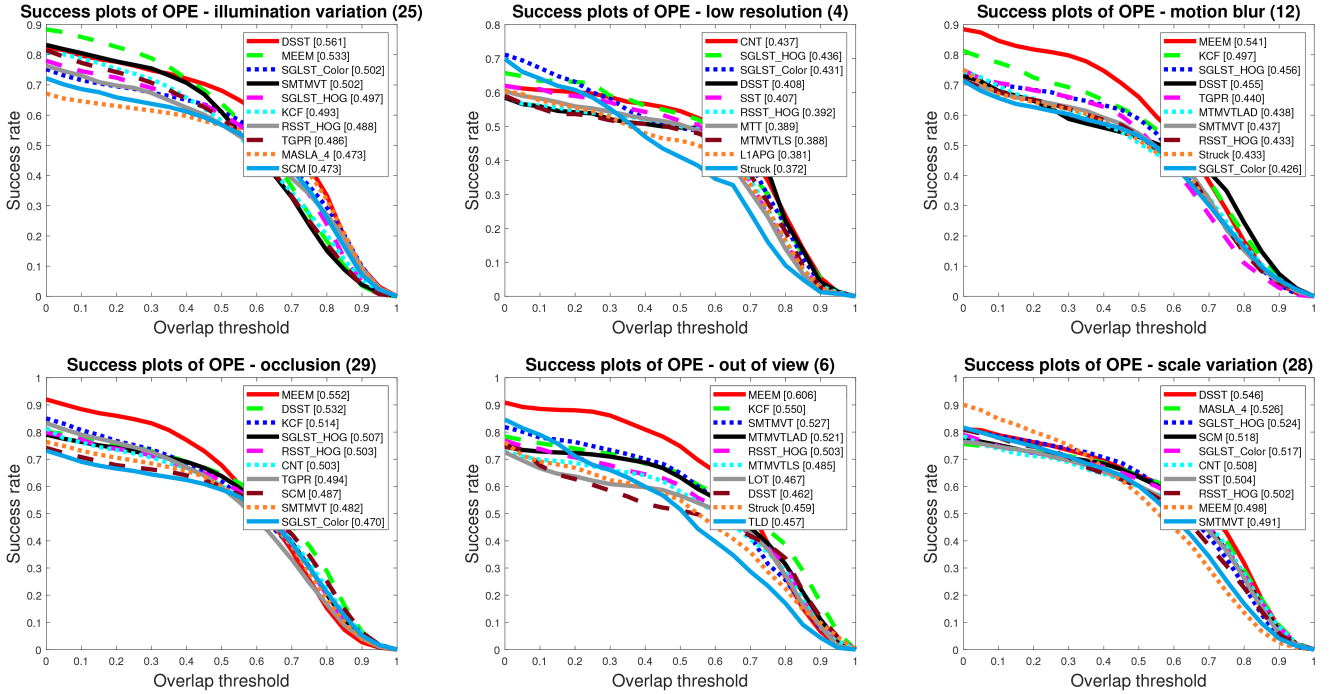

Fig. 6 *OTB50 OPE success plots for IV, LR, MB, OCC, OV, and SV challenge subsets. The value appearing in the title is the number of sequences in the specific subset. The values appearing in the legend are the AUC scores. Only the top ten trackers are presented, while the results of the other trackers can be found in [38]*

For this benchmark data set, there are online available tracking results for 29 trackers [38]. In addition, we include the tracking results of additional 12 recent trackers, namely, MTMVTLS [31], MTMVTLAD [32], MSLA-4 [35] (the recent version of ASLA [34]), SST [5], SMTMVT [47], CNT [48], TGPR [49], DSST [14], PCOM [50], KCF [45], MEEM [46], and RSST [36]. Following the protocol proposed in [38], we use the same parameters for SGLST_Color and SGLST_HOG on all the sequences to obtain the one-pass evaluation (OPE) results, which are conventionally used to evaluate trackers by initialising them using the ground truth location in the first frame. We present the overall OPE success plot and the OPE success plots for BC, DEF, FM, IPR, and OPR challenge subsets in Fig. 5 and the OPE success plots for IV, LR, MB, OCC, OV, and SV challenge subsets in Fig. 6. These success plots show the percentage of successful frames at the overlap thresholds ranging from zero to one, where the successful frames are the ones who have overlapping scores larger than a given threshold. For a fair comparison, we use the area under the curve (AUC) of each success plot to rank the trackers. For the convenience of the reader, we only include the top ten of the 43 compared trackers in each plot. The values in the parenthesis alongside the legends are AUC scores. The values in the parenthesis alongside the titles for 11 challenge subsets are the number of video sequences in the respective subset.

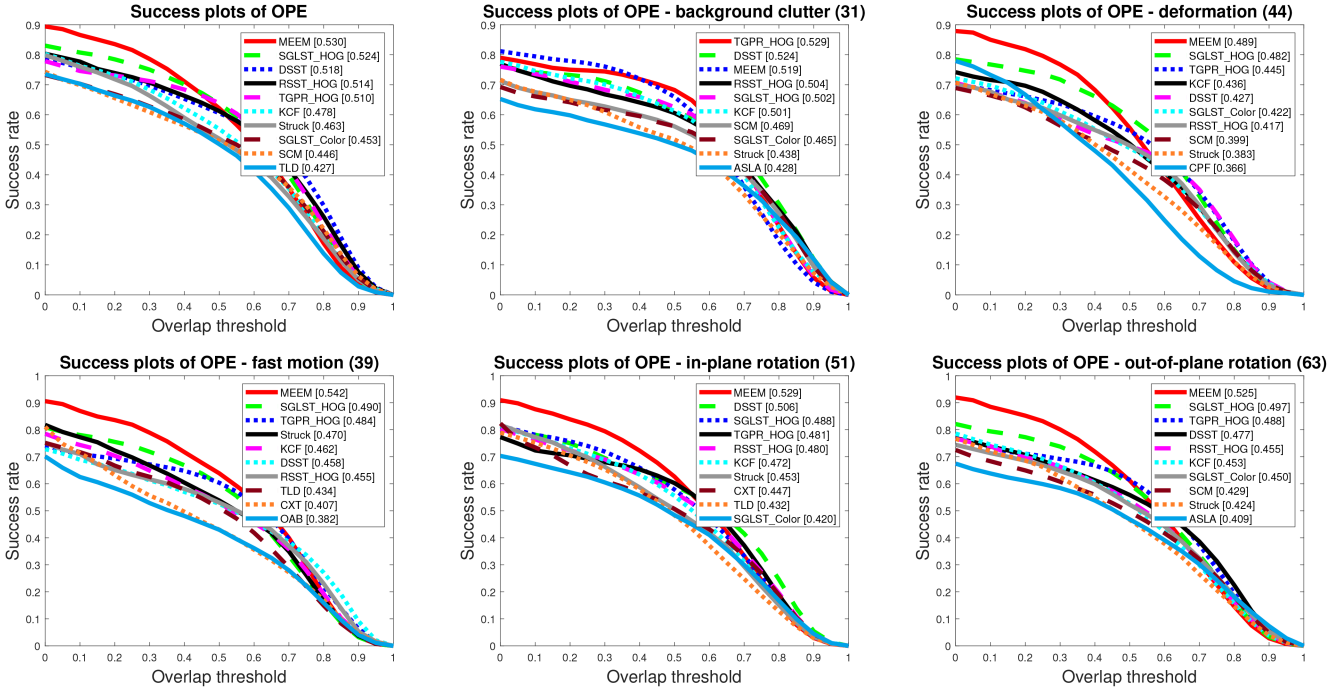

Fig. 7 *OTB100 overall OPE success plots and the OPE success plots BC, DEF, FM, IPR, and OPR challenge subsets. The value appearing in the title is the number of sequences in the specific subset. The values appearing in the legend are the AUC scores. Only the top ten trackers are presented, while the results of the other trackers can be found in [38]*

It is clear from the overall success plot in Fig. 5 that SGLST_HOG (i.e. incorporating HOG features in SGLST) improves the tracking performance (i.e. the AUC score) of SGLST_Color (i.e. incorporating intensity features in SGLST) by 6.31% due to the incorporation of the HOG features instead of the intensity features. The similar improvement trends are also observed in [36, 41]. Among the 29 baseline trackers employed in [38], SCM achieves the most favourable performance. SGLST_HOG outperforms SCM by 11.42% in terms of the AUC score. Compared with the 12 additional recent trackers, SGLST_HOG outperforms MSLA-4, SMTMVT, KCF, TGPR, RSST_HOG, CNT, and DSST by 9.88, 9.66, 8.17, 5.10, 2.39, 2.02, and 0.36%, respectively. It achieves an overall performance that is short of 1.80% on comparing with the performance of the best tracker, MEEM. It should be mentioned that the two variants of SGLST (i.e. SGLST_Color and SGLST_HOG) slightly outperform the two variants of RSST (i.e. RSST_Color and RSST_HOG) by 0.53 and 2.39%, respectively. These slight improvements indicate that the proposed optimisation model is better than its counterpart in RSST due to its employment of a group-sparsity regularisation term to adopt local and spatial information of the target candidates and attain the spatial layout structure among them.

The proposed SGLST_HOG performs significantly better than traditional sparse trackers such as L1APG [29], LRST [30], ASLA [34], MTT [43], and MTMVTLS [31]. It outperforms most recent sparse trackers such as MTMVTLAD [32], SST [5], MSLA-4 [35], SMTMVT [47], and RSST_HOG [36]. SGLST_HOG, which yields the AUC score of 0.556, also achieves better performance than some CF-based methods such as KCF (AUC score of 0.514) and DSST (AUC score of 0.554). Moreover, it outperforms some deep learning-based methods such as CNT (AUC score of 0.545) and GOTURN (AUC score of 0.444) [51]. However, the proposed SGLST_HOG yields lower performance than some deep learning-based methods such as FCNT [52] (AUC score of 0.599), DLSSVM [53] (AUC score of 0.589), and RSST_Deep [36] (AUC score of 0.590). We believe that SGLST can be further improved by incorporating the deep features, as similar improvement trends are clearly shown in RSST [36].

We further evaluate the performance of SGLST on 11 challenge subsets. As demonstrated in Figs. 5 and 6, SGLST_HOG ranks as one of the top three trackers in five subsets with DEF, OPR, LR, MB, and SV challenges and SGLST_Color ranks as one of the top three trackers in two subsets with IV and LR challenges. SGLST_HOG achieves the fourth rank on two subsets with IPR and OCC challenges and the fifth rank on two subsets with FM and IV challenges. SGLST_Color achieves the fifth rank on one subset with the SV challenge. However, SGLST is not in the list of the top ten trackers for the subset with the OV challenge. In overall, the proposed SGLST ranks as one of the top five trackers on nine out of 11 subsets (e.g. 22 out of 50 image sequences) with DEF, OPR, LR, MB, SV, IV, IPR, OCC, and SV challenges.

4.3 Experimental results on the OTB100 benchmark

We conduct the experiments on the OTB100 tracking benchmark [39] to evaluate the overall performance of the proposed SGLST_Color and SGLST_HOG under different challenges. This benchmark is the extension of OTB50 [38], which consists of 100 annotated sequences. Each sequence is labelled with attributes specifying the presence of different challenges. The two sequences, *jogging* and *Skating*, have two annotated targets. The rest of 98 sequences have one annotated target. We evaluate the proposed SGLST against 29 baseline trackers used in [39] and seven recent trackers including DSST [14], PCOM [50], KCF [45], MEEM [46], TGPR [49], and RSST [36]. The other six trackers compared in the OTB50 benchmark do not provide their results on the OTB100 benchmark. Therefore, they are excluded from this experiment.

Fig. 7 presents the overall OPE success plot and the OPE success plots for BC, DEF, FM, IPR, and OPR challenge subsets and Fig. 8 provides the OPE success plots for IV, LR, MB, OCC, OV, and SV challenge subsets. Top ten trackers are included in each plot. The overall success plotted in Fig. 7 clearly demonstrates that the best tracker MEEM, a multi-expert tracker employing an online linear SVM and an explicit feature mapping method, has a slightly better AUC score than the second best tracker, the proposed SGLST_HOG. The difference in terms of the AUC score is only 0.006. SGLST_HOG improves its variant, SGLST_Color, by 15.67% due to the use of HOG features over intensity features. It also improves the fourth-ranked tracker RSST_HOG, the most recent sparse tracker, by 1.95% due to its novel optimisation model. Compared to the third-ranked tracker DSST, a discriminative CF-based tracker, it improves the AUC score of DSST by 1.16%.

Similar to the tracking results obtained on the OTB50 tracking benchmark, the proposed SGLST_HOG performs significantly better than traditional sparse trackers such as L1APG [29], LRST

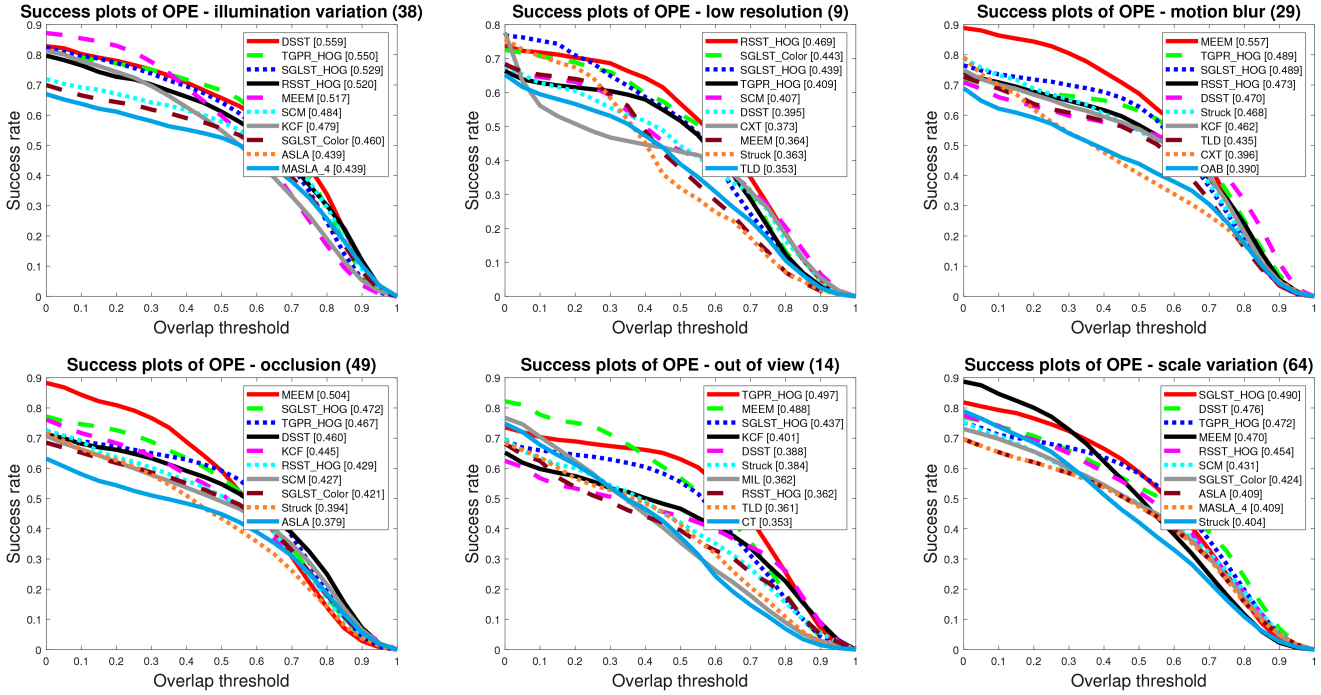

Fig. 8 *OTB100 OPE success plots for IV, LR, MB, OCC, OV, and SV challenge subsets. The value appearing in the title is the number of sequences in the specific subset. The values appearing in the legend are the AUC scores. Only the top ten trackers are presented, while the results of the other trackers can be found in [39]*

[30], ASLA [34], and MTT [43]. It also outperforms RSST_HOG [36], one of the most recent sparse trackers that provided the results on the OTB100 tracking benchmark. SGLST_HOG, which yields the AUC score of 0.524, also achieves better performance than some CF and deep learning-based methods such as KCF (AUC score of 0.478), DSST (AUC score of 0.518), and GOTURN (AUC score of 0.427) [51]. However, it yields lower performance than some deep learning-based methods such as CNN-SVM (AUC of 0.554), CF2 (AUC of 0.562) [54], and RSST_Deep (AUC of 0.583). We believe that incorporating the deep features in SGLST can further improve its tracking performance to be more comparable with the other deep learning-based trackers.

We further evaluate the performance of SGLST on 11 challenge subsets in the OTB100 benchmark. As demonstrated in Figs. 7 and 8, SGLST_HOG ranks as one of the top three trackers in all 11 subsets except one subset with the BC challenge and SGLST_Color ranks as one of the top three trackers in one subset with the LR challenge. SGLST_HOG achieves the fifth rank on the subset with the BC challenges. It achieves better performance than the best tracker, MEEM, in three subsets with IV, LR, and SV challenges. Overall, the proposed SGLST ranks as the top three trackers on ten out of 11 subsets (e.g. 69 out of 100 image sequences) with DEF, FM, IPR, OPR, IV, SV, LR, MB, OCC, and OV challenges.

5 Conclusions

In this study, we propose a novel tracker, called SGLST, which exploits local patches within target candidates in the particle filter framework. Unlike conventional local sparse trackers, SGLST employs a new convex optimisation model to preserve spatial layout structure among the local patches. To solve the proposed optimisation model, we develop an efficient numerical algorithm consisting of two subproblems with closed-form solutions based on ADMM. We test the performance of the proposed tracker with two types of features including grey-level intensity features and HOG features.

The qualitative and quantitative results on 16 publicly available frame sequences demonstrate that SGLST_HOG outperforms all compared state-of-the-art trackers in terms of the overlap score. The experimental results on OTB50 and OTB100 tracking benchmarks demonstrate that SGLST_HOG outperforms all compared state-of-the-art trackers except the MEEM tracker in terms of the average AUC score. The inferior performance of MEEM in handling the SV is shown in the *car4*, *box*, *human5*, and *human7* sequences out of the 16 publicly available frame sequences, which contain the SVs. MEEM cannot achieve a favourable performance on these sequences and is therefore, outperformed by SGLST_HOG. Similar performance trends are also demonstrated in the SV success plots in Figs. 6 and 8 for the SV challenge subset of the OTB50 and OTB100 data sets. However, MEEM achieves better performance when OCC, OV, and MB happen. It is also designed to recover from drifting, which can happen in any sequences in the OTB50 and OTB100 data sets. Since the number of sequences containing the aforementioned challenges in OTB data sets is larger than the number of sequences containing the SV challenge, MEEM demonstrates slight better performance than the SGLST_HOG on OTB50 and OTB100 tracking benchmarks.

6 References

[1] Yilmaz, A., Javed, O., Shah, M.: 'Object tracking: a survey', *ACM Comput. Surv.*, 2006, **38**, (4), p. 13

[2] Salti, S., Cavallaro, A., Di Stefano, L.: 'Adaptive appearance modeling for video tracking: survey and evaluation', *IEEE Trans. Image Process.*, 2012, **21**, (10), pp. 4334–4348

[3] Pang, Y., Ling, H.: 'Finding the best from the second bests-inhibiting subjective bias in evaluation of visual tracking algorithms'. Proc. IEEE Int. Conf. on Computer Vision, Sydney, Australia, 2013, pp. 2784–2791

[4] Kristan, M., Matas, J., Leonardis, A., *et al.*: 'The visual object tracking vot2015 challenge results'. Proc. IEEE Int. Conf. on Computer Vision Workshops, Santiago, Chile, 2015, pp. 1–23

[5] Zhang, T., Liu, S., Xu, C., *et al.*: 'Structural sparse tracking'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Boston, United States, 2015, pp. 150–158

[6] Zhang, T., Bibi, A., Ghanem, B.: 'In defense of sparse tracking: circulant sparse tracker'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, United States, 2016, pp. 3880–3888

[7] Osmanlıoğlu, Y., Shakibajahromi, B., Shokoufandeh, A.: 'Autonomous multicamera tracking using distributed quadratic optimization'. Int. Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, Venice, Italy, 2017, pp. 175–188

[8] Javanmardi, M., Qi, X.: 'Visual tracking of resident space objects via an RFS-based multi-Bernoulli track-before-detect method', *Mach. Vis. Appl.*, 2018, **29**, (7), pp. 1191–1208. Available at https://doi.org/10.1007/s00138-018-0963-6

[9] Avidan, S.: 'Ensemble tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2007, **29**, (2), pp. 261–271

[10] Grabner, H., Grabner, M., Bischof, H.: 'Real-time tracking via on-line boosting'. British Machine Vision Conf. (BMVC), Edinburgh, Scotland, 2006, vol. 1, p. 6

[11] Grabner, H., Leistner, C., Bischof, H.: 'Semi-supervised on-line boosting for robust tracking'. European Conf. Computer Vision, Marseille, France, 2008, pp. 234–247

[12] Babenko, B., Yang, M.H., Belongie, S.: 'Visual tracking with online multiple instance learning'. IEEE Conf. Computer Vision and Pattern Recognition, 2009. CVPR 2009, Miami, United States, 2009, pp. 983–990

[13] Kalal, Z., Matas, J., Mikolajczyk, K.: 'P-N learning: bootstrapping binary classifiers by structural constraints'. 2010 IEEE Conf. Computer Vision and Pattern Recognition (CVPR), San Francisco, United States, 2010, pp. 49–56

[14] Danelljan, M., Häger, G., Khan, F., *et al.*: 'Accurate scale estimation for robust visual tracking'. British Machine Vision Conf., Nottingham, 1–5 September 2014

[15] Ma, C., Yang, X., Zhang, C., *et al.*: 'Long-term correlation tracking'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Boston, United States, 2015, pp. 5388–5396

[16] Hu, Q., Guo, Y., Chen, Y., *et al.*: 'Correlation filter tracking: beyond an open-loop system'. British Machine Vision Conf. (BMVC), London, United Kingdom, 2017

[17] Black, M.J., Jepson, A.D.: 'Eigentracking: robust matching and tracking of articulated objects using a view-based representation', *Int. J. Comput. Vis.*, 1998, **26**, (1), pp. 63–84

[18] Comaniciu, D., Ramesh, V., Meer, P.: 'Kernel-based object tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, **25**, (5), pp. 564–577

[19] Adam, A., Rivlin, E., Shimshoni, I.: 'Robust fragments-based tracking using the integral histogram'. 2006 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, New York, United States, 2006, vol. 1, pp. 798–805

[20] Ross, D.A., Lim, J., Lin, R.S., *et al.*: 'Incremental learning for robust visual tracking', *Int. J. Comput. Vis.*, 2008, **77**, (1–3), pp. 125–141

[21] Kwon, J., Lee, K.M.: 'Visual tracking decomposition'. 2010 IEEE Conf. Computer Vision and Pattern Recognition (CVPR), San Francisco, United States, 2010, pp. 1269–1276

[22] Roffo, G., Melzi, S.: 'Online feature selection for visual tracking', In British Machine Vision Conference (BMVC), York, United Kingdom, 2016

[23] Chao, Y.W., Yeh, Y.R., Chen, Y.W., *et al.*: 'Locality-constrained group sparse representation for robust face recognition'. 2011 18th IEEE Int. Conf. Image Processing (ICIP), Brussels, Belgium, 2011, pp. 761–764

[24] Li, S., Yin, H., Fang, L.: 'Group-sparse representation with dictionary learning for medical image denoising and fusion', *IEEE Trans. Biomed. Eng.*, 2012, **59**, (12), pp. 3450–3459

[25] Zohrizadeh, F., Kheirandishfard, M., Kamangar, F.: 'Image segmentation using sparse subset selection'. 2018 IEEE Winter Conf. Applications of Computer Vision (WACV), Lake Tahoe, United States, 2018, pp. 1470–1479

[26] Zohrizadeh, F., Kheirandishfard, M., Ghasedidizaji, K., *et al.*: 'Reliability-based local features aggregation for image segmentation'. Int. Symp. Visual Computing, Las Vegas, United States, 2016, pp. 193–202

[27] Tang, S., Zhou, N., Xiao, L.: 'Pansharpening via locality-constrained sparse representation'. British Machine Vision Conf. (BMVC), London, United Kingdom, 2017

[28] Mei, X., Ling, H.: 'Robust visual tracking and vehicle classification via sparse representation', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **33**, (11), pp. 2259–2272

[29] Bao, C., Wu, Y., Ling, H., *et al.*: 'Real time robust l1 tracker using accelerated proximal gradient approach'. 2012 IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Providence, United States, 2012, pp. 1830–1837

[30] Zhang, T., Ghanem, B., Liu, S., *et al.*: 'Low-rank sparse learning for robust visual tracking'. European Conf. Computer Vision, Firenze, Italy, 2012, pp. 470–484

[31] Hong, Z., Mei, X., Prokhorov, D., *et al.*: 'Tracking via robust multi-task multiview joint sparse representation'. Proc. IEEE Int. Conf. on Computer Vision, Sydney, Australia, 2013, pp. 649–656

[32] Mei, X., Hong, Z., Prokhorov, D., *et al.*: 'Robust multitask multiview tracking in videos', *IEEE Trans. Neural Netw. Learn. Syst.*, 2015, **26**, (11), pp. 2874–2890

[33] Liu, B., Huang, J., Kulikowski, C., *et al.*: 'Robust visual tracking using local sparse appearance model and k-selection', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2013, **35**, (12), pp. 2968–2981

[34] Jia, X., Lu, H., Yang, M.H.: 'Visual tracking via adaptive structural local sparse appearance model'. 2012 IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Providence, United States, 2012, pp. 1822–1829

[35] Jia, X., Lu, H., Yang, M.H.: 'Visual tracking via coarse and fine structural local sparse appearance models', *IEEE Trans. Image Process.*, 2016, **25**, (10), pp. 4555–4564

[36] Zhang, T., Xu, C., Yang, M.H.: 'Robust structural sparse tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018, **41**, (2), pp. 473–486

[37] Boyd, S., Parikh, N., Chu, E., *et al.*: 'Distributed optimization and statistical learning via the alternating direction method of multipliers', *Found Trends Mach Learn*, 2011, **3**, (1), pp. 1–122

[38] Wu, Y., Lim, J., Yang, M.H.: 'Online object tracking: a benchmark'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Portland, United States, 2013, pp. 2411–2418

[39] Wu, Y., Lim, J., Yang, M.H.: 'Object tracking benchmark', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015, **37**, (9), pp. 1834–1848

[40] Dalal, N., Triggs, B.: 'Histograms of oriented gradients for human detection'. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, 2005. CVPR 2005, San Diego, United States, 2005, vol. 1, pp. 886–893

[41] Henriques, J.F., Caseiro, R., Martins, P., *et al.*: 'Exploiting the circulant structure of tracking-by-detection with kernels'. European Conf. Computer Vision, Firenze, Italy, 2012, pp. 702–715

[42] Hare, S., Golodetz, S., Saffari, A., *et al.*: 'Struck: structured output tracking with kernels', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016, **38**, (10), pp. 2096–2109

[43] Zhang, T., Ghanem, B., Liu, S., *et al.*: 'Robust visual tracking via multi-task sparse learning'. 2012 IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Providence, United States, 2012, pp. 2042–2049

[44] Babenko, B., Yang, M.H., Belongie, S.: 'Robust object tracking with online multiple instance learning', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **33**, (8), pp. 1619–1632

[45] Henriques, J.F., Caseiro, R., Martins, P., *et al.*: 'High-speed tracking with kernelized correlation filters', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015, **37**, (3), pp. 583–596

[46] Zhang, J., Ma, S., Sclaroff, S.: 'MEEM: robust tracking via multiple experts using entropy minimization'. Proc. European Conf. on Computer Vision (ECCV), Zurich, Switzerland, 2014

[47] Javanmardi, M., Qi, X.: 'Robust structured multi-task multi-view sparse tracking'. 2018 IEEE Int. Conf. Multimedia and Expo (ICME), San Diego, United States, 2018, pp. 1–6

[48] Zhang, K., Liu, Q., Wu, Y., *et al.*: 'Robust visual tracking via convolutional networks without training', *IEEE Trans. Image Process.*, 2016, **25**, (4), pp. 1779–1792

[49] Gao, J., Ling, H., Hu, W., *et al.*: 'Transfer learning based visual tracking with Gaussian processes regression'. European Conf. Computer Vision, Zurich, Switzerland, 2014, pp. 188–203

[50] Wang, D., Lu, H.: 'Visual tracking via probability continuous outlier model'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Columbus, United States, 2014, pp. 3478–3485

[51] Held, D., Thrun, S., Savarese, S.: 'Learning to track at 100 fps with deep regression networks'. European Conf. Computer Vision, Amsterdam, The Netherlands, 2016, pp. 749–765

[52] Wang, L., Ouyang, W., Wang, X., *et al.*: 'Visual tracking with fully convolutional networks'. Proc. IEEE Int. Conf. on Computer Vision, Araucano Park, Chile, 2015, pp. 3119–3127

[53] Ning, J., Yang, J., Jiang, S., *et al.*: 'Object tracking via dual linear structured SVM and explicit feature map'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, United States, 2016, pp. 4266–4274

[54] Ma, C., Huang, J.B., Yang, X., *et al.*: 'Hierarchical convolutional features for visual tracking'. Proc. IEEE Int. Conf. on Computer Vision, Araucano Park, Chile, 2015, pp. 3074–3082