Semantics-Aware Visual Object Tracking

Rui Yao, Guosheng Lin, Chunhua Shen, Yanning Zhang, Qinfeng Shi

***Abstract*—In this paper, we propose a semantics-aware visual object tracking method, which introduces semantics into the tracking procedure and extends the model of object with explicit semantic prior to enhance the robustness of three key aspects of the tracking framework, *i.e.,* appearance model, search scheme, and scale adaptation. We first present a semantic object proposal generation method for video sequence to generate high-quality category-oriented object proposals. Then a hybrid semantics-aware tracking algorithm with semantic compatibility is proposed. This algorithm takes full advantages of globally sparse semantic object proposal prediction and locally dense prediction with a template model and semantic distractor-aware colour appearance model. Further, we propose to exploit semantics to localise object accurately via an energy minimisation framework based scale adaptation method, which jointly integrates dense location prior, instance-specific colour and category-specific semantic information. Extensive experiments are conducted on two widely used benchmarks, and the results demonstrate that our method achieves state-of-the-art performance.**

***Index Terms*—Visual object tracking, Semantic object proposal, Appearance model, Search scheme, Scale adaptation.**

I. Introduction

VISUAL object tracking is a pre-requisite for many important computer vision applications, such as automated surveillance and activity or behaviour recognition. Although significant progress has recently been achieved with respect to tracking performance [1], [2], [3], [4], [5] and evaluation methodology [6], [7], considerable challenges still exist in constructing a robust and efficient tracking system. In general, we can decompose a tracking system into two main constituent parts: appearance model and search scheme. Substantial effort has been devoted to maintaining an adaptive and robust appearance model. This problem is typically formulated as an online learning framework, where a generative model or discriminative model is incrementally updated during tracking. Most of these existing tracking algorithms assume that the target location changes smoothly over time, and localise the object within a search window centred at the previous object location [4], [3], which can be referred to as motion model [8]. However, this strategy may not be appropriate for handling some challenging situations, such as fast motion, abrupt deformation, and long-term occlusion, which may break the assumptions that the object is correctly tracked in the previous frames and that the temporal object motion is smooth.

Object-proposal-based methods have recently been proposed [9], [10], [11], [12] to address this problem. However, the quality of the object proposal is crucial for the tracking algorithm. The existing trackers generate proposals by using edge [9], [12], geometry [10], or saliency [11]. The candidate proposals generated by these features always include some non-object regions, which often mislead the tracking algorithm.

To address this limitation, in this work, we introduce semantic information into visual tracking. Compared to edge, geometry or saliency characteristics, semantics provides valuable guidance for tracking due to the following benefits.

- Semantics is category oriented, it can reduce the interference from other categories or background. Meanwhile, by taking semantic object proposals into account as negative samples during the update process, the discriminative capacity of the appearance model can be greatly enhanced.
- Semantics can be regarded as a special high-level feature, which is obtained through offline training on large-scale datasets. In this paper, we take advantage of two different levels of semantics. The category-level semantics indicates which type of object we are interested in, and it is not varying. The object-level semantics estimates the contours of the object of interest. It can perceive the shape and contours of an object accurately, which is useful for estimating object scale and location.

The first contribution of the paper is to propose a semantic object proposal generation method for video sequence. We extract a list of object mask proposals in each frame through the following four main steps: category-level semantics generation, object-level semantic estimation, semantic smoothing via temporal evidence passing, and object proposal generation. Compared with generic object proposal approach, the proposed method generates fewer high-quality semantic object proposals, which makes it more suitable for some vision problems that deal with specific categories of objects. In Fig. 16, the experimental results of our method and other state-of-the-art object proposal generation methods validate our motivation.

The second contribution is presenting a semantics-aware visual object tracking framework that extends the object model with explicit semantic information. In the proposed framework, semantics contributes to improving the tracking performance in three core components of a typical tracker: object appearance model, search scheme, and scale estimation.

- We propose a semantic distractor-aware colour appear-

This work was supported by the National Natural Science Foundation of China (No. 61772530), Natural Science Foundation of Jiangsu Province of China (No. BK20171192).

R. Yao is with School of Computer Science and Technology, China University of Mining and Technology, China. E-mail: ruiyao@cumt.edu.cn.

G. Lin is with School of Computer Science and Engineering, Nanyang Technological University, Singapore. E-mail: gslin@ntu.edu.sg.

C. Shen and Q. Shi are with School of Computer Science, The University of Adelaide, SA, Australia. E-mail: {chunhua.shen, qinfeng.shi}@adelaide.edu.au.

Y. Zhang is with School of Computer Science, Northwestern Polytechnical University, Xi'an, 710072, China. E-mail: ynzhang@nwpu.edu.cn.

(*Corresponding author: Guosheng Lin*)

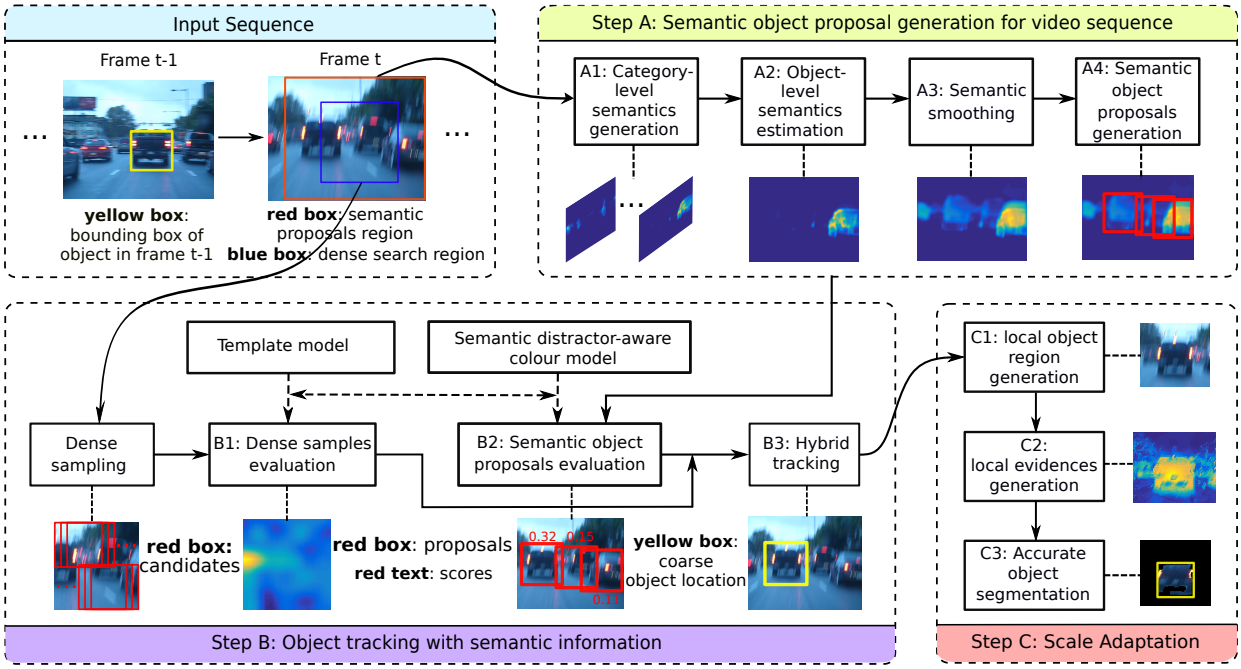


Fig. 1: Overall procedure of our tracking approach. Given the bounding box of an object at frame $t-1$, we will track the object at frame t within the local dense search region (*i.e.,* blue box) and sparse semantic proposal region (*i.e.,* red box) centred at the object location at frame $t-1$. In step A, we first perform semantics estimation to generate a smoothed object-level semantic score map for frame t. Then, several semantic object proposals are generated. In step B, we perform hybrid tracking by combining a wider semantic object proposal prediction and local dense sample prediction to obtain a coarse object location. Finally, a segmentation-based scale adaptation method is performed to obtain an accurate bounding box of object in step C. To this end, the template model and semantic-distractor-aware colour model are updated according to the new object appearance.

ance model to efficiently distinguish object pixels from background and distractors discovered by object-level semantic prior.

- In search scheme, globally sparse semantic object proposal prediction is employed to make up the limitation of locally dense sample prediction. We further present a semantic compatibility function to prevent the tracker drift to background explicitly.
- We present a scale adaptation method based on an energy minimisation framework by incorporating dense template, colour and sparse semantic evidence.

Experimentally, we show that our method achieves competitive tracking performance against the state-of-the-art trackers, and our framework can flexibly integrate semantic information into other tracking approaches.

Note that the semantics are determined adaptively, and our approach does not need any prior about the category of the target, and also does not assume what category of object to track. When there are no explicit matched semantics for the target, the algorithm can automatically find similar semantics. Even if there are no similar semantics, our approach can achieve object tracking in the absence of semantics, and the tracking procedure is similar to most existing dense object tracking algorithms. In the remainder of this paper, we will describe our tracking approach in detail according to main steps of the framework shown in Fig. 1.

II. Related Work

Visual Tracking. In this paper, we focus on several discriminative tracking algorithms based on tracking-by-detection schema. Examples of such approaches include the support vector tracker [13], online adaptive boosting (OAB) [14], multiple instance learning (MIL) [15], random forest [5], structural output tracking (Struck) [4] and correlation filter tracker [16], [3]. A number of trackers [17], [18], [19] developed from the correlation filter framework have been proposed to improve performance. In [20], [21], the correlation filter trackers are combined with colour features to achieve better performance. All of the above trackers rely on shallow features, the tracking performance have recently been improved using deep features [22], [23], [24], [25], [26].

Semantics in Visual Tracking. Semantics is always used as a priori information for a specific target tracking method, such as, human tracking [27], vehicles tracking [28], and hand tracking [29]. However, those methods are hard to be generalised to track other object categories. In contrast, semantics is rarely explored in generic object tracking task. Recently, a category-free tracking method is presented in [30], this method simultaneously tracks a single target and recognises its category. Our method is different from the previous works. First, the proposed method can handle generic category object. Second, [30] builds three deep networks to capture the category and generic features of objects. In contrast, our method can freely employ hand-craft and deep feature to

represent the appearance model of object.

Semantic Segmentation. In recent years, fully convolution network (FCN)-based methods [31] have become the most popular choice for semantic segmentation. However, FCN-based methods are generally limited by low-resolution prediction. Many techniques have been proposed to address this limitation [32]. Lin *et al.* [33] proposed RefineNet to address the problem for efficient high-resolution predictions. The network architecture of RefineNet explicitly exploits all the information available along the down-sampling process to enable high-resolution predictions using long-range and short-range residual connections with identity mapping. RefineNet can obtain a more accurate contour of objects in images, which is very suitable for object tracking.

Sequential Semantic Labelling. To exploit temporal inconsistencies in video sequence for semantic labelling, Miksik *et al.* [34] proposed an approach based on recursive weighted filtering in a small neighbourhood with optical flow and image-based similarities. Coprie *et al.* [35] presented a graph-matching-based temporal semantic segmentation approach for RGBD videos. Unlike our temporal evidence passing algorithm that only performs on semantic score maps, these methods require some auxiliary cues, such as pixels and optical flow, which is not efficient for online object tracking.

Scale Estimation. There are several types of scale estimation methods used in object tracking. Kalal *et al.* [5] utilised the relative position of local components to estimate the scale. Li *et al.* [36] and Danelljan *et al.* [37] proposed adaptive scale estimation methods. In [38], Possegger *et al.* proposed estimating scale by segmenting objects based on colour histograms. Son *et al.* [39] proposed generating a segmentation mask on the likelihood of boosting decision tree In contrast, we propose a scale adaptation algorithm by fusing a dense template location prior, instance-specific colour and category-specific semantic evidence.

III. Semantic Object Proposal Generation for Video Sequence

In this section, we will introduce our semantic object proposal generation method for specific object in video sequence. This section is referred to as Step A in Fig. 1. The novelty of the proposed method is three-fold.

First, we present an *object-level semantics generation method* that contains semantic ranking and selection components to incorporate all semantics within the target region in each frame. Second, simply applying per-frame object-level semantic score map to tracking algorithm is not sufficient because temporal consistency of semantic labels among video sequence is not considered over time. These temporal inconsistent semantic labels may be caused by motion blur, illumination variation, *etc.* To alleviate this problem, we propose an efficient temporal evidence passing algorithm by utilising the coherence of video in timing, and generate a smoothed object-level semantic score map that fuses per-frame semantic information and temporal evidences passed from previous frames in a greedy manner. We refer to this algorithm as *semantic smoothing via temporal evidence passing*. Third, we propose an iterative *semantic object proposals generation* method to obtain a small amount of high quality semantic object proposals based on the smoothed object-level semantic score map.

A. Category-level Semantics Generation

To generate category-level semantic score maps of independent images, we employ the deep CNN model, which has shown outstanding performance for semantic segmentation problems. In this work, we employ the RefineNet model [33] to generate semantic score maps. Note that our framework is general in that any other method can be incorporated to generate category-level semantic score maps.

We first train RefineNet on the PASCAL Visual Object Classes 2012 dataset [40] We note that our semantic labelling results can cover most of objects of interest in visual object tracking. Thus, the pre-trained semantic segmentation model is able to capture the general characteristics of objects. For training, RefineNet takes the ResNet-101 network [41] for initialisation. Given a 2D image I_t with width W and height H in a tracking video sequence, RefineNet takes this image as as input, and returns an output from the last layer $M'_t \in \mathbb{R}^{W\times H\times C}$ as semantic score maps of I_t, where C is the dimension of the score vector at one pixel location.

B. Object-level Semantics Estimation

The category-level semantic score maps only provide semantics for each general category object, whereas our objective is to estimate semantics for the unknown category tracking object, and only the bounding box (*i.e.*, a rectangular object region) is provided at the first frame; thus, we must decide which category-level semantic score maps are useful for localising the object. Therefore, we introduce a two-step method to generate an object-level semantic score map for the target, *i.e.,* semantic ranking and semantic selection.

Semantic ranking. The output of category-level semantic segmentation is C layers of score map M'_t, where the score at each pixel location (i, j) in the c-th layer represents the likelihood of this pixel belonging to category c. Let $\mathbf{b}_1$ denotes the bounding box of object in the first frame, where $\mathbf{b}_1$ is a 4-dimensional vector that indicates the coordinate of the left-top corner (2 dimensions), width and height of the object region. The average semantic score m_c of the object region in the c-th layer can be obtained by summing over all semantic scores at each location and then dividing by the size of region $\mathbf{b}_1$:

$$m_c = \frac{1}{|\mathbf{b}_1|}\sum\nolimits_{(i,j)\in\mathbf{b}_1} M'_t(i, j, c). \tag{1}$$

where $|\mathbf{b}_1|$ denotes the number of positions in $\mathbf{b}_1$. Given semantic score maps with C layers, we can rank the layers of these maps in descending order by the value of m_c, $c = \{1, \cdots, C\}$.

Semantic selection. The bounding box provided in the first frame may contain objects from multiple categories, for example, in *motorBike1* sequence of OTB dataset, the object of interest (*i.e.*, the content in bounding box of the first frame) is a person riding a motorbike, person and motorbike are belongs

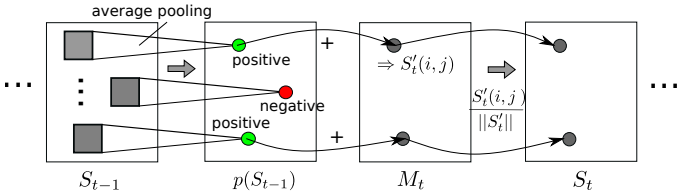


Fig. 2: Illustration of the proposed temporal evidence passing procedure. The smoothed object-level semantic score map S_t is obtained from the last semantic score map S_{t-1} and independent score map M_t in a greedy manner. The green circles denote that the value is positive at this location on score map $p(S_{t-1})$, and the red circle represents the negative value. We pass the positive temporal evidence to the next frame and ignore the negative one (as shown in Eq. (5)). Examples are presented in Fig. 3.

to different two categories in our pre-trained semantic model, they has different score map. Therefore, we need to determine which category score maps should be incorporated into the tracking process while avoiding selecting noisy ones.

We can exploit the cumulative average score $\sum_{c=1}^{C} m_c$ to estimate which semantic score maps contain the target. We first accumulate the values of m_c starting from $c = 1$, and we stop the accumulation when the cumulative average score $\sum_{c=1}^{c^*} m_c$ is greater than a pre-defined threshold τ (set as 0.5 in this paper). Note that m_c is normalised by the size of the object region in Eq. (1), and the accumulation of m_c in the object region is not greater than 1. Then we believe that the first to c^*-th layer score maps contain semantics of the tracked target, and these semantic score maps should be used for target tracking. Meanwhile, the remaining score maps are not associated with the target and should be ignored. Thus, the final semantic score map is obtained by summing all relevant score maps $M_t = \sum_{c=1}^{c^*} M'_t(\cdot,\cdot,c)$.

C. Semantic Smoothing via Temporal Evidence Passing

Because the object-level semantic score map M_t for the tracked object is generated independently for individual frames, it should be further smoothed by exploiting the temporal consistency in the video sequence. For example, in a scene that contains cars, motion blur will appear in the image if the camera or car moves rapidly, and the semantic segmentation algorithm may not be able to effectively label the target (*e.g.* car), causing the object to appear in the previous semantic score map and suddenly vanish in the current map (as shown in the second and fourth column of Fig. 3).

Inspired by the time series analysis literature, we can obtain the smoothed object-level semantic score map by calculating the mean semantic score of each location along the time series with a temporal smoothing bandwidth. However, this procedure has two limitations. First, simple averaging is not sensitive to the variation in pixels caused by target or camera motion, and it will introduce undesirable artifacts. Furthermore, we have to choose the temporal bandwidth (*i.e.*, length) for the moving average. To address these problems, we propose a temporal evidence passing algorithm.

We rescale the object-level semantic map to $[-1, 1]$, and then $M_t(i,j)$ represents the discriminative semantic score at

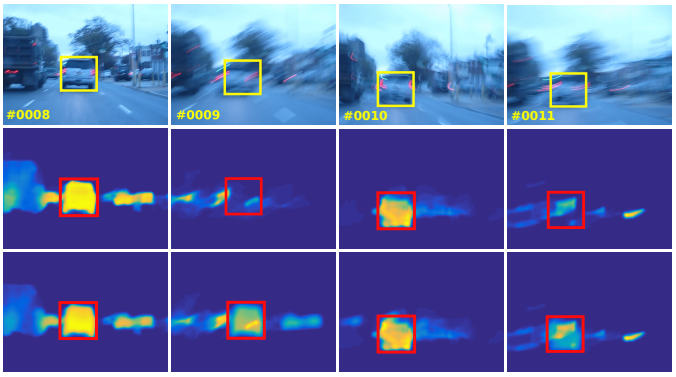


Fig. 3: Examples of semantic smoothing via temporal evidence passing for sequence *BlurCar1* from the OTB benchmark. **Top Row**: four consecutive frames in the sequence. **Middle Row**: per-frame object-level semantic score map obtained from Sec. III-B. **Bottom Row**: smoothed object-level semantic score map via temporal evidence passing. The rectangles show the location of tracked object. The images appear heavy blur caused by camera shake. Note that the second and fourth column, the target is clearer in smoothed object-level semantic score map (bottom row) than per-frame map (middle row).

pixel (i,j) in frame t. The smoothed object-level semantic score $S_t(i,j)$ should include the semantic scores in the current frame and previous δ_t frames, which can be defined as follows:

$$S_t(i,j) = \frac{1}{\delta_t} \sum_{k=t-\delta_t}^{t} p\big[M_{k-1}(i,j)\big] + M_k(i,j), \qquad (2)$$

where δ_t denotes the bandwidth of temporal evidence passing.

To take object movement into account, we define the temporal evidence from the last frame $t-1$ as the mean of the neighbouring region to obtain smoothed temporal evidence:

$$p\big[M_{t-1}(i,j)\big] = \frac{1}{|\Omega(i,j)|} \sum_{(i',j')\in\Omega(i,j)} M_{t-1}(i',j'), \qquad (3)$$

where $\Omega(i,j)$ denotes the neighbour region of (i,j). This neighbour region can be regarded as the smoothness constraint, which can retain the temporal consistency in the video sequence and make the algorithm invariant to local translation. By performing temporal evidence passing in a recursive manner, we can obtain the smoothed semantic score:

$$S_t(i,j) = \frac{1}{\delta_t}\big(p\big[S_{t-1}(i,j)\big] + M_t(i,j)\big), t > 1, \qquad (4)$$

where $S_t(i,j) = M_t(i,j)$ if $t = 1$. $p\big[S_{t-1}(i,j)\big]$ is the temporal evidence from all of the previous frames. Its definition is similar to $p\big[M_{t-1}(i,j)\big]$ in Eq. (3), we only need to replace the M_{t-1} on the right of the formula with P_{t-1}. Furthermore, to avoid passing negative temporal evidence in the future frames, we ignore such negative evidence:

$$S'_t(i,j) = \max\big(p\big[S_{t-1}(i,j)\big], 0\big) + M_t(i,j). \qquad (5)$$

Then the smoothed semantic score is defined by $S_t(i,j) = \frac{1}{\delta_t} S'_t(i,j)$. However, we note that the representation in Eq. (5) has to choose an appropriate bandwidth δ_t, which is not

always straightforward because the motion is often difficult to estimate. To address this limitation, we propose the normalised mean semantic score map as

$$S_t = \frac{S'_t}{||S'_t||}, \tag{6}$$

where S'_t represents a 2-D matrix at frame t, where each element $S'_t(i,j)$ denotes the semantic score at location (i,j).

Inspired by the efficiency of the convolutional layer in CNN, we implement Eq. (5) by using average pooling to obtain temporal evidence at each location. Specifically, the average pooling can be regarded as convolution of the input semantic score map and an $N \times N$ size kernel, where each value of the kernel is $1/(N \times N)$. Here $N \times N$ is the size of neighbour region $\Omega(i,j)$. In this way, the proposed temporal evidence passing algorithm is able to be performed in an online manner very efficiently. Fig. 2 shows the greedy procedure of the proposed evidence passing algorithm. As shown in Fig. 3, the target disappears at frame 9 (the second column) and is not clear at frame 11 (the fourth column) in the per-frame object-level semantic score map, and our method can recover and enhance the object-level semantic score map by propagating temporal semantic evidence of the relative region from previous frames effectively.

D. Semantic Object Proposal Generation

Based on the smoothed object-level semantic score map, we present a method to generate semantic object proposals to produce a limited number of high-quality candidate object locations in a wider search region for sparse prediction.

Given the smoothed object-level semantic score map S_t (as shown in Eq. (6)) at frame t, we follow an iterative procedure to generate high-quality semantic object proposals. We calculate the integral image of score map S_t and densely sample overlapping candidate bounding boxes within a large search region on this integral image efficiently. We find the bounding box $\mathbf{b}_P^*$ with maximum average semantic score $S_t(\mathbf{b}_P^*)$ as the first semantic object proposals,

$$S_t(\mathbf{b}_P^*) = \frac{1}{|\Omega_{\mathbf{b}_P^*}|} \sum\nolimits_{(i,j)\in\Omega_{\mathbf{b}_P^*}} S_t(i,j). \tag{7}$$

Then, the semantic score of the region covered by $\mathbf{b}_P^*$ on S_t is set to 0, and we make a candidate region $\mathbf{b}_P^2$ to be the second proposal if it sufficiently large, which can be written as

$$S_t(\mathbf{b}_P^2) \geq \sigma_S S_t(\mathbf{b}_P^*), \text{ where } \mathbf{b}_P^2 = \text{argmax}_{\mathbf{b}_P} S_t(\mathbf{b}_P). \tag{8}$$

Here, σ_S is a parameter that ranges from 0 to 1 (set as 0.5 in this paper). Once the second proposal is obtained, we again set the value of its region on S_t to zero, and we use Eq. (8) to find the next proposals until the condition in Eq. (8) is not satisfied. With the non-maximum suppression strategy [42], we discard the overlapping candidates to avoid ambiguous object proposals. After the above iterative process, a set of semantic object proposals $\{\mathbf{b}_P^1, \cdots, \mathbf{b}_P^K\}$ that sparsely scatter on image I_t can be obtained for both estimating the new object location and updating semantic distractor-aware colour model.

IV. Object Tracking with Semantic Information

In this section, we will introduce how to extend the object tracking framework with explicit semantic information. This section is referred to as Step B in Fig. 1. Our approach incorporates semantics into two components of object tracking: appearance model and search scheme. In the following, we first present our semantic distractor-aware colour appearance model to enhance the discriminative capabilities of tracking approach, and then propose hybrid search scheme with semantic compatibility that combines locally dense sampling prediction with globally sparse semantic object proposal prediction to handle complex tracking scene.

The proposed hybrid semantic-aware search scheme has two benefits. First, as mentioned above, semantic object proposals can expand the search range and handle rapid object or camera movements and occlusions. Second, the dense sampling-based prediction within a small window is able to handle small movements, motion blur, illumination changes and irregular translations of objects more effectively.

A. Template Model

Our basic template model is based on discriminative kernelised correlation filters (KCFs) [3]. Given the bounding box of the object $\mathbf{b}_{t-1}$ in the previous frame I_{t-1}, the task of the template model is to find a discriminative function f to separate the object from background in the current frame I_t. For this task, we train a classifier to find the function $f_T(\mathbf{x}_t;\Theta_t) = \mathbf{y}$ between input image patch $\mathbf{x}_t$ in frame I_t of size $W' \times H'$ and its label $\mathbf{y}$ from a training set, where Θ_t is the parameter of the discriminative function. For more details of KCFs, we refer to [3].

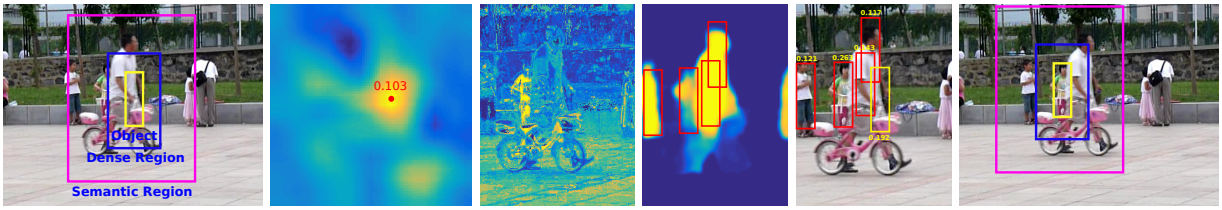
B. Semantic Distractor-Aware Colour Model

In addition to template model, we also learn a colour model with semantic distractor awareness to efficiently distinguish object pixels from background and distractors. Our colour model contains two components: object-background colour model and object-distractor colour model, where distractors are obtained from semantic object proposals. We define the score function f_C of the colour model as:

$$f_C(I_t;\mathbf{w}_t) = (1-\beta) f_B(I_t;\mathbf{u}_t) + \beta f_D(I_t;\mathbf{v}_t), \tag{9}$$

where β is a pre-defined linear combination parameter (set as 0.5 in this paper) and f_B and f_D are discriminative score functions for the object-background colour model and object-distractor colour model, respectively. In Eq. (9), $\mathbf{w}_t = [\mathbf{u}_t, \mathbf{v}_t]$ are the model parameters that we want to learn for the score function.

Because the score function f_C is a linear combination of f_B and f_D, we can separate the score function Eq. (9) into two independent problems to find the optimal parameter $\mathbf{w}_t$. The two score functions f_B and f_D are both calculated using histogram features [38]. For the object-background colour model, we use the object and its surrounding region to obtain the histogram. In contrast, we construct the histogram based on the object and potential semantic distractors for the object-distractor colour model. We use a variable learning rates η_D



(a) Frame t & windows (b) Template score (c) Colour score (d) Proposals (e) Hybrid predict (f) Tracking result

Fig. 4: Illustration of hybrid object tracking. (a) Image at time t and the corresponding three windows indicate the bounding box, the local search region of dense prediction and update, the wide search region to generate semantic proposals. (b) Densely response of template model (red point and text indicate the maximum position and value, the score is very low due to occlusion). Note that the figure just shows the template response map of dense prediction region. (c) The densely pre-pixel score of colour model with semantic distractor-aware. (d) Each red bounding box indicates one semantic object proposal. (e) Hybrid prediction score (text with yellow colour) of each object proposal (red rectangle) and best candidate of dense prediction (yellow rectangle). (f) Tracking result based on hybrid prediction.

(set as 0.04 in this paper) to online update the histogram. Please refer to [38], [21] for more details.

C. Hybrid Tracking with Semantic Compatibility

In the following, we will evaluate the discriminative scores of dense and sparse candidates to estimate the object location. We introduce a hybrid search scheme by combining local dense prediction with wider sparse prediction. When a new frame indexed by t enters, the hybrid tracking algorithm is performed by the following two steps.

In the first step, local dense prediction is performed on the dense search window. For each location in this window, we use the detection procedure of KCF (described in Sec.IV-A) to obtain the score of the template model. With the lookup-table-based histogram feature, the colour object likelihood of each location can also be efficiently obtained by using Eq. (9). Then, we select the region with the maximum merged score of two discriminative functions as the best candidate bounding box of dense prediction $\mathbf{b}_D^* = \text{argmax}_{\mathbf{b_D}} f(I_t, \mathbf{b_D})$, where

$$f(I_t, \mathbf{b_D}) = (1-\eta) f_T(I_t, \mathbf{b}_D; \boldsymbol{\alpha}_t) + \eta f_C(I_t, \mathbf{b}_D; \mathbf{w}_t), \quad (10)$$

where $\boldsymbol{\alpha}_t$ is the coefficient vector, and η is a merge factor to combine the template and colour models, and it is set to 0.3 in our implementation.

In the second step, we perform wider sparse prediction for the semantic object proposals obtained in Sec. III-D. Similar to the first step, the discriminative score of each semantic object proposal is composed of two parts (as shown in Eq. (10)). It is straightforward to calculate the colour likelihood of each object proposal by averaging the score of function f_C for all pixels in the proposal region. However, the response score of the template model is efficiently calculated in the Fourier domain for all shifted samples. To evaluate the single proposal $\mathbf{b}_P^n, n = 1, \cdots, K$, the template score can be obtained by

$$f_T(I_t, \mathbf{b}_P^n; \boldsymbol{\alpha}_t) = \sum_q \mathbf{k}^{\mathbf{b}_P^n}(q) \cdot \boldsymbol{\alpha}_t(q), \mathbf{k}^{\mathbf{b}_P^n} = k(\mathbf{b}_P^n, \hat{\mathbf{x}}), \quad (11)$$

where $\hat{\mathbf{x}}$ is the maintained appearance template and $F(q)$ represents the q-th element in the vector F. Because only one sample is evaluated, we calculate the score in the spatial domain.

To prevent the tracker from drifting to the background, we further introduce a *semantic compatibility function* f_S, which allows the bounding box to be valid only when it is detected in the object-level semantic score map:

$$f_S(\mathbf{b}_P^n) = \begin{cases} 0, & K \geq 1 \quad and \quad S_t(\mathbf{b}_P^n) \leq \lambda_S \\ 1, & otherwise \end{cases}, \quad (12)$$

where K is the total number of semantic object proposals and $S_t(\mathbf{b}_P^n)$ is the average semantic score (as defined in Eq. (7)). λ_S is the parameter to verify the existence of the semantic object, and λ_S is set to 0.1 in this paper. Then the final estimated new object location can be obtained by

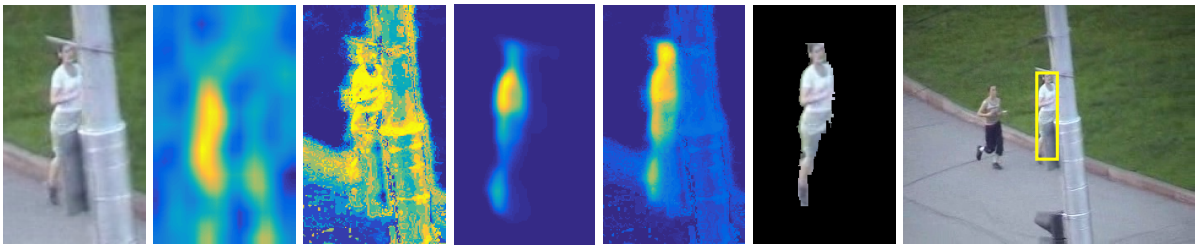
$$\mathbf{b}_t^* = \text{argmax}_{\mathbf{b} \in \{\mathbf{b}_D^*, \mathbf{b}_P^1, \cdots, \mathbf{b}_P^K\}} f(I_t, \mathbf{b}) f_S(\mathbf{b}). \quad (13)$$

Fig. 4 presents an illustration of the proposed hybrid object tracking procedure.

Difference with related work. It should be noted that the proposed semantic-aware tracking approach is different from recently proposed approaches Staple [21] and Distractor-Aware online Tracking (DAT) [38]. First, we use different search schemes from those approaches. We propose a globally sparse and locally dense hybrid search scheme to handle the complex scene of visual tracking. In contrast, both Staple and DAT employ locally dense window based search scheme. Second, we present a semantic distractor-aware colour appearance model. Compare to the use of colour similarity in DAT to get distractors, our model treats the remaining semantic object proposals as distractors to update the colour appearance model. Semantic object proposal is closer to the real object than colour distractor. The addition of semantic distractors enhances the discriminative capabilities of the colour appearance model. Third, we present a semantic compatibility function to prevent the tracker drift to background explicitly, which was not appeared in the previous tracking approaches.

V. Scale Adaptation via Segmentation

In Sec. IV, the object is localised coarsely in the new frame. We then introduce how to exploit semantic information by a scale adaptation method to obtain accurate localisation of the object. This section is referred to as Step C in Fig. 1.



(a) Local region (b) Location (c) Colour (d) Semantics (e) Posterior (f) Mask (g) Localisation result of frame t

Fig. 5: Illustration of object scale adaptation via segmentation. (a) The local segmentation region, which is a padding region of the estimated object bounding box obtained by hybrid object tracking. (b) The location prior (*i.e.*, response of template object model). (c) Instance-specific colour probability obtained from colour object model with semantic distractor-aware. (d) The smoothed semantic probability map. (e) The pre-pixel posterior probability map. (f) Segmentation mask generated by Graph Cut. (g) The localisation result with scale adaptation based on segmentation mask.

To adaptively adjust the scale of the object, we have to accurately localise the object in each frame, which will involve two major problems: the first is the local evidence that determines which pixels belong to the object, and the second problem is how to jointly infer the object/background regions based on this local evidence. Several sources of local evidence can be used. The template-based feature is robust for identifying the object location and distinguishing different objects, but it is sensitive to the spatial configuration of object (*i.e.*, deformable object). Additionally, it is not always able to cover the entire object. Colour information is the most common feature used to segment objects; however, for many objects, the colour of different parts are non-uniform across the entire object. In addition, the proposed category information in semantic probability maps can provide a prior of the shape of the object, but it is sensitive to motion and illumination changes. Therefore, we propose integrating these features together and then inferring the best object region with smoothness constraints.

A. Accurate Object Localisation via Segmentation

We employ an energy-minimisation-based segmentation method to obtain accurate object regions. We perform segmentation on a padding region of the estimated object location $\mathbf{b}_t^*$ based on the scale estimated in the previous frame (refer to Step C1 in Fig. 1), which leads to computational efficiency while avoiding sudden changes in the scale. The energy function for segmenting the local object region is given by

$$E(\mathbf{x}_t, \mathbf{z}) = -\sum_k p(z_k|\mathbf{x}_{t,k}) + \gamma \sum_{(k,h)\in\mathcal{E}} V(z_k, z_h|\mathbf{x}_t), \quad (14)$$

where the posterior $p(z_k|\mathbf{x}_{t,k})$ is regarded as the local evidence term, which will be introduced in Sec. V-B. $z_k \in \{0, 1\}$ and $\mathbf{x}_t$ is the local segmentation object patch. $\mathcal{E}$ denotes the standard eight-way connectivity edge set. γ is the smoothness parameter (set as 0.3). $V(z_k, z_h|\mathbf{x}_t)$ is the pairwise smoothness potential term, and is chosen from [43]. The segmentation is defined as the minimiser $\operatorname{argmin}_{\mathbf{z}} E(\mathbf{x}_t, \mathbf{z})$ of the energy using Graph Cut algorithm [44].

B. Local Evidence

1) Template-based Object Location Prior: The discriminative score of the template object model provides a strong prior of the object location. The location prior $p_L(z_k = 1|\mathbf{x}_{t,k})$ is the likelihood that the k-th pixel in image patch $\mathbf{x}_t$ belongs to the object ($z_k = 1$). The likelihood of each pixel can be obtained from the response of KCF on the segmentation region $f_T(\mathbf{x}_t; \boldsymbol{\alpha}_t)$. And then we resize the response map to fit the original size of image patch.

2) Instance-Specific Colour Probability: Instance-specific colour probabilities $p_C(z_k|\mathbf{x}_{t,k})$ are modelled by a linear combination of object-background colour model and object-distractor colour model, which is shown in Eq. (9). Therefore, the score of the colour appearance model is regarded as the probability that the k-th pixel belongs to a foreground object.

3) Category-Specific Semantic Probability: Each layer of the original semantic probability map M_t' represents the discriminative likelihood that the pixels belong to each category. The smoothed map S_t is the accumulated likelihood that obtained by accumulating maximum score of the trajectory. Therefore, we use the semantic score of each location in S_t as category-specific semantic probability $p_S(z_k|\mathbf{x}_t)$. To make the score a probability, we scale all scores to $[0, 1]$.

By combining these three types of probabilities (refer to Step C2 in Fig. 1), we can obtain a better unique foreground posterior $p(z_k|\mathbf{x}_{t,k}) = \gamma_L p_L(z_k|\mathbf{x}_{t,k}) + \gamma_C p_C(z_k|\mathbf{x}_{t,k}) + \gamma_S p_S(z_k|\mathbf{x}_{t,k})$, where $\gamma_L, \gamma_C, \gamma_S$ are weights. We set the weights as $\gamma_L = \gamma_S = 0.2$ and $\gamma_C = 0.6$. An example of foreground posterior is shown in Fig. 5e.

The final object bounding box is estimated as the maximum connected component that fully contains the segmented foreground region. The scale of the object can be calculated by comparing the bounding box size of the current frame and previous frame. If the estimated scale change is below a reasonable percentage, then we discard the segmentation and keep the scale of the previous frame. Fig. 5 presents an example of our object scale adaptation procedure.

Difference with related work. The proposed approach is different from the related work DAT and Staple in two aspects. First, our method explicitly exploits semantics to enhance the localisation accuracy of the object. Second, we employ an energy minimisation framework to incorporate template, colour and semantic evidence. In contrast, DAT and Staple use an adaptive segmentation threshold with colour information to get the object region.

TABLE I: Comparison with 12 state-of-the-art trackers on 100 sequences in terms of DP, OP, CLE, and VOR. The top performer in each measure is shown in red, and the second and third best are shown in blue and green, respectively.

	SAT (Ours)	SRDCF	DSST	CNN-SVM	DAT	Staple	SAMF	TLD	KCF	MEEM	STRUCK	MIL	OAB
DP	0.82	0.79	0.68	0.81	0.43	0.78	0.75	0.60	0.70	0.78	0.64	0.45	0.48
OP	0.75	0.73	0.60	0.65	0.36	0.71	0.67	0.50	0.55	0.62	0.52	0.33	0.40
CLE	25.63	38.53	50.34	21.76	77.19	31.42	36.39	60.70	44.75	27.71	47.03	71.78	69.21
VOR	0.63	0.61	0.52	0.56	0.33	0.59	0.56	0.43	0.48	0.53	0.47	0.33	0.37

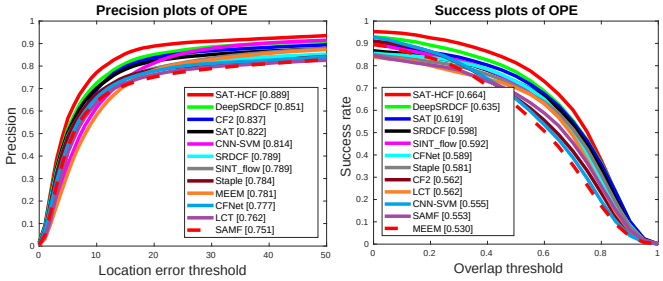


Fig. 6: Overall distance precision plot (left) and overlap success plot (right) with one-pass evaluation (OPE) over 100 sequences (OTB-100). The legends show the precision scores and AUC scores for each tracker.

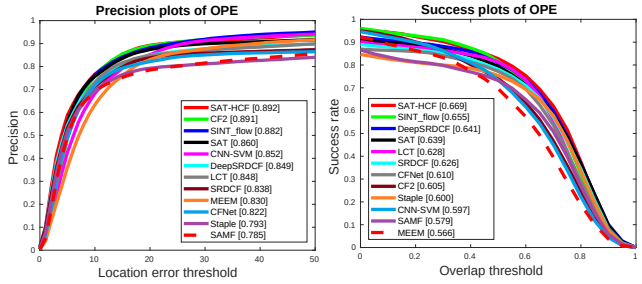


Fig. 7: Overall distance precision plot (left) and overlap success plot (right) with one-pass evaluation (OPE) over 51 sequences (OTB-2013).

VI. Experiments

In this section, we present our empirical evaluation on two state-of-the-art benchmark datasets, OTB [6] and VOT16 [7], and compare with several recent methods.

Experimental Setup. Our approach is implemented in MATLAB on a workstation with an Intel Core i7-4770 3.40GHz CPU and 12GB of RAM. We use the same HOG features for the template model as for the KCF tracker [3]. For the colour model, we use quantised RGB colours with a histogram using 32 bins per channel. The regularisation parameter λ and learning rate for the template model are chosen the same as for KCF.

A. Overall Performance

1) Object Tracking Benchmark: We first evaluate the proposed tracker on the benchmark [45], [6] which includes 100 fully annotated video sequences with various targets and backgrounds. We evaluated our method on two OTB datasets: OTB-2013 [45], which has 51 video sequences, and OTB [6], which has 100 video sequences including those of OTB-2013.

Evaluation Methodology. We follow the evaluation protocol provided by the benchmark [6]. Four metrics with one-pass evaluation (OPE) are used to evaluate all the compared trackers: 1) bounding box overlap, which is measured by VOC overlap ratio (VOR); 2) centre location error (CLE), 3) distance precision (DP), and 4) overlap precision (OP). Our tracking approach is denoted as SAT, which represents the characteristics of our approach, *i.e.*, semantics-aware tracker. Our method can also use CNN feature, to verify this, we build a new method called 'SAT-HCF (Hierarchical Convolutional Features)'. In SAT-HCF, we replace the KCFs-based template model in the proposed SAT method with the HCF-based template model. Here, we employ the HCF model proposed in [26]. The rest of the SAT-HCF algorithm remains the same as SAT.

Results. To verify the performance of our approach, we compare against existing algorithms on the benchmark and several state-of-the-art trackers including SINT_flow [25], CFNet [24], DeepSRDCF [23], CF2 [26], CNN-SVM [22], SRDCF [17], Staple [21], DAT [38], KCFDPT [12], MEEM [13], SAMF [36], and DSST [37]. SINT_flow, CFNet, Deep SRDCF, CF2 and CNN-SVM employ the features from CNNs, while the remaining methods are based on the traditional hand-crafted features. The precision and success plots are generated by computing the rates of successfully tracked frames at many different thresholds in terms of CLE and VOR. For the precision plots, we rank the trackers according to the result with an error threshold of 20 pixels. For the success plots, the trackers are ranked by the AUC scores.

Fig. 6 shows the overall precision and success plots on OTB-100. The proposed SAT performs significantly better than all the methods reported in [6]. In the precision plot, our tracker performs 16% better than the best tracker evaluated in the original benchmark (Struck [4]). As shown in Tab. I, our algorithm consistently performs better than 12 recently proposed methods. Since our approach includes a colour model, we also report the performance of DAT [38] in Tab. I. Our approach also performs better than Staple [21]. The results in Tab. I demonstrate that the proposed SAT improves the performance by utilising semantics. In addition, we also report our results on the OTB-2013 dataset, as shown in Fig. 7. From the results shown in Fig. 6 and Fig. 7, we find that the SAT-HCF obtains the best result both in terms of precision and success rate on OTB-100 and OTB-2013 datasets. In addition, the proposed SAT also achieves better performance than two deep learning based methods (*i.e.,* SINT_flow, CFNet) on OTB-100.

The performance of a tracking approach can be affected by several challenges. Fig. 8 presents the performance of tracking algorithm for various challenging attributes provided in the benchmark as mentioned above. SAT is effective in handling

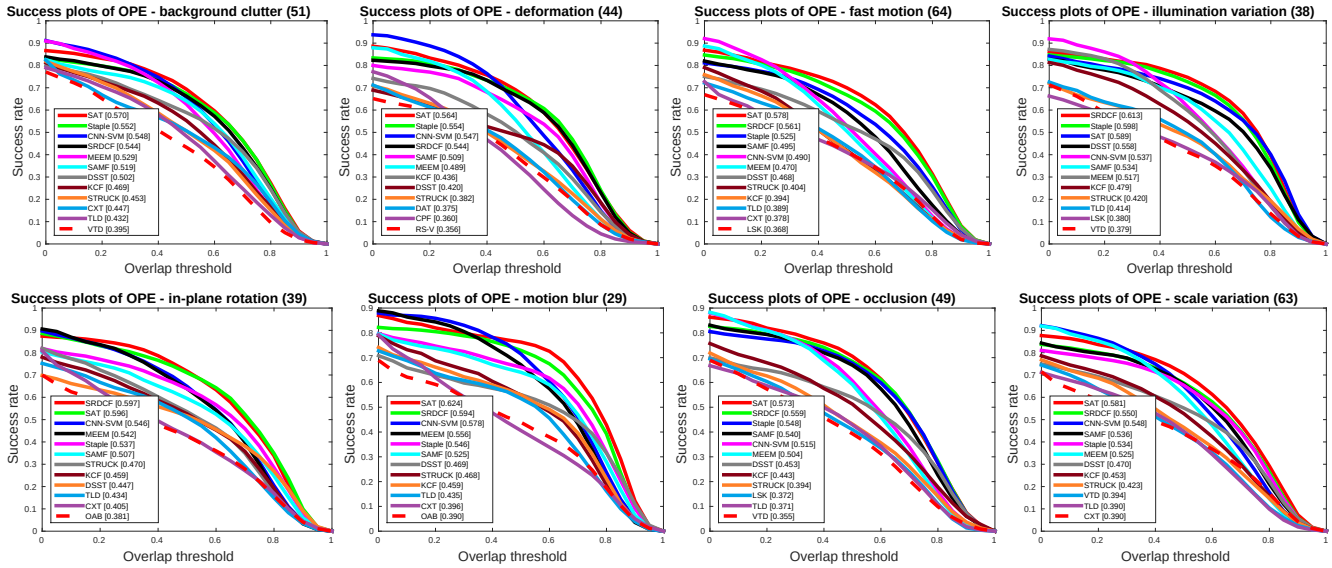


Fig. 8: Overlap success plots over eight tracking challenge attributes: background clutter, deformation, fast motion, illumination variation, in-plane rotation, motion blur, occlusion and scale variation. The value presented in the title represents the number of sequences corresponding to the attributes. The overlap score is shown in the legend for each tracker.

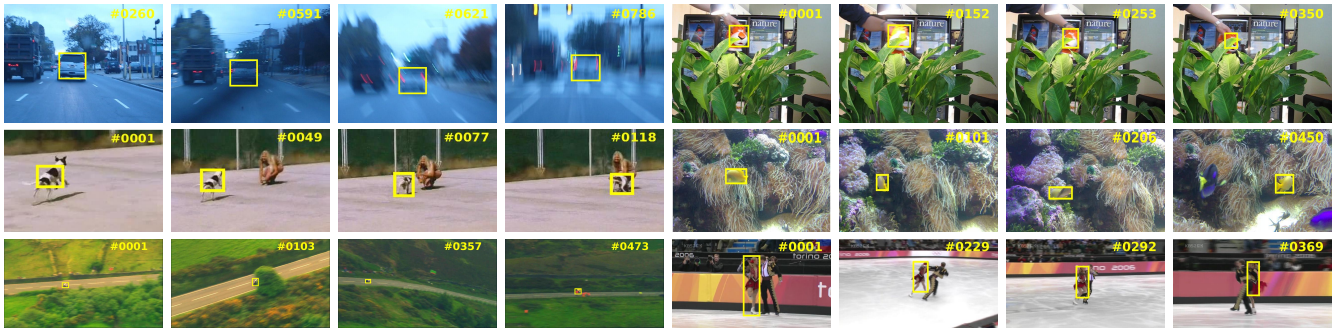


Fig. 9: Several representative frames of qualitative tracking results. Our tracker exhibits robustness in challenging scenarios like motion blur (left of row 1), illumination changes (right of row 1), deformation (left of row 2), drastic change of appearance (right of row 2), small target (left of row 3), and similar-object interference (right of row 3). These sequences come from the OTB benchmark (*BlurCar1, Tiger1, Dog* and *Skating2*), and VOT16 benchmark (*fish3* and *road*), and objects belong to different categories. The yellow rectangle indicates the bounding box obtained by the proposed tracker.

all kinds of attributes compared to the existing state-of-the-art approaches. In particular, our tracker is more robust than the compared trackers for deformable object. By introducing semantic object proposals and segmentation-based scale adaptation, our tracker can outperform Staple [21], SAMF [36], and DSST [37] by substantial margins, especially when fast motion, motion blur, and scale variation occur. The first plot of Fig. 8 shows our tracker achieves better performance than the competing trackers when the object suffers from background clutters, which refer to the background near the target has the similar colour or texture as the target.

2) Visual Object Tracking Challenge: For completeness of the analysis, we also present the evaluation results on the recent benchmark VOT16 [7]. The VOT16 challenge considers single-camera, single-target, model-free, causal trackers applied to short-term tracking, which evaluates trackers on 60 sequences chosen from a pool of 365, selected such that seven different challenging situations are well represented.

Evaluation Methodology. We follow the protocol described in [7] to evaluate our tracking algorithm, and we compare it with several state-of-the-art trackers. This evaluation protocol uses three primary measures to analyse tracking performance: accuracy, robustness, and expected average overlap (EAO). The accuracy is the average overlap between the predicted and ground truth bounding boxes during successful tracking periods. The robustness measures how many times the tracker loses the target (fails) during tracking. EAO estimates how accurate the estimated bounding box is after a certain number of frames are processed since initialisation.

Results. The results are summarised in sequence-pooled and attribute-normalised AR-rank and AR-score plots in Fig. 10. The sequence-pooled AR-rank plot is obtained by concatenating the results from all sequences and creating a single rank list, while the attribute-normalised AR-rank plot is created by ranking the trackers over each attribute and averaging the rank lists. The AR-score plot was constructed in a similar

TABLE II: The VOT16 performance comparison of top 20 trackers with the proposed tracking approach. We show the EAO, accuracy (Acc.) and robustness (Rob.) measures, as well as the rank of accuracy (Acc. Rank) and robustness (Rob. Rank). The trackers are ranked by EAO scores. The top performer in each measure is shown in red, and the second and third best are shown in blue and green, respectively.

Tracker	EAO	Acc.	Acc. Rank	Rob.	Rob. Rank
CCOT	0.331	0.54	10.5	0.24	3.80
TCNN	0.325	0.55	10	0.27	5.00
SSAT	0.321	0.58	8.5	0.29	5.67
MLDF	0.311	0.49	11.5	0.23	3.75
SAT (Ours)	0.296	0.53	11	0.33	12.86
Staple	0.295	0.54	10.5	0.38	15.12
DDC	0.293	0.54	10.5	0.34	11.50
EBT	0.291	0.46	15.64	0.25	4.00
SRBT	0.290	0.50	11.5	0.35	13.78
STAPLE+	0.286	0.56	10	0.37	13.40
DNT	0.278	0.51	11	0.33	6.40
SSKCF	0.277	0.55	11	0.37	15.12
SiamRN	0.277	0.55	10.5	0.38	15.00
DeepSRDCF	0.276	0.53	11	0.33	8.40
SHCT	0.266	0.55	10.5	0.40	13.40
MDNet_N	0.257	0.54	10.5	0.34	4.00
FCF	0.251	0.55	10.5	0.46	16.20
SRDCF	0.247	0.54	11	0.42	13.91
RFD_CF2	0.242	0.48	12.44	0.37	12.44
GGTv2	0.238	0.52	11	0.47	20.50
DPT	0.236	0.49	11.5	0.49	20.50

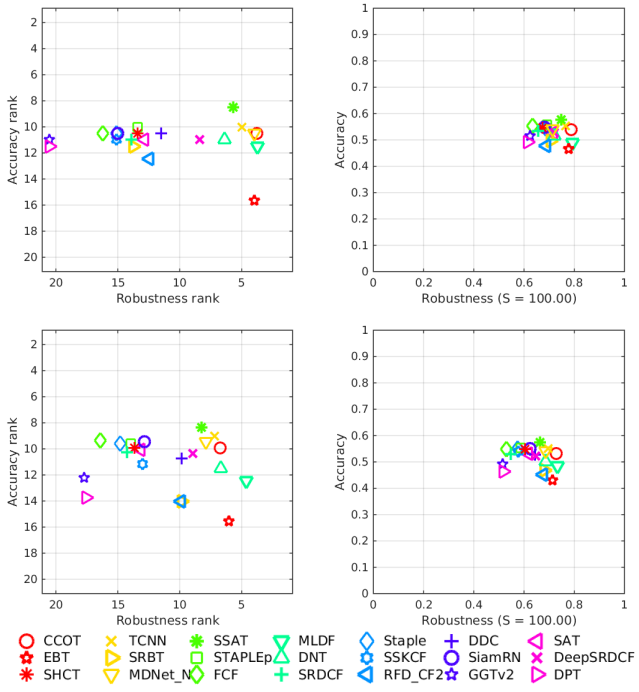


Fig. 10: The AR-rank plots and AR-score plots generated by sequence pooling (upper row) and attribute normalisation (lower row).

fashion. The raw values for the sequence-pooled results and the average overlap scores are shown in Tab. II. Overall, our tracker outperforms most of the best methods in the challenging VOT16 benchmark. Note that the proposed approach is superior to MDNet_N [1], which is the average performance of fifteen state-of-the-art trackers published in 2015 and 2016

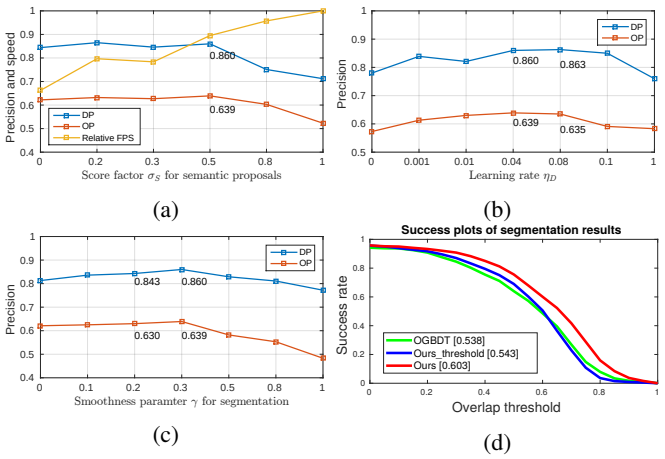


Fig. 11: Critical parameter and component analysis. (a), (b) and (c) show our tracking performance under different settings of score factor σ_S, learning rate η_D, and smoothness parameter γ, respectively. (a) also shows the tracking speed. (d) shows the success plots of segmentation results on NOT dataset.

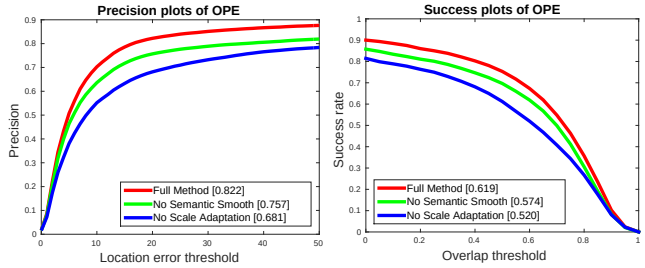


Fig. 12: Comparison of performance of our approach with various modified versions on OTB-100. The legends show the precision scores and AUC scores for each method.

at major computer vision venues [7]. In particular, our tracker also outperforms EBT [9], which is a recent tracker based on EdgeBox object proposal. Staple+ is an improved version of Staple tracker by integrating multiple features is presented [7]. The overall EAO our method is better than Staple+ (as shown in Tab. II). Our method is different from Staple+, we focus on extending the model of object with explicit semantic prior. For the segmentation part, our framework can fully use the semantics to obtain accurate object contour. This result shows the effectiveness of our semantic object proposals.

Qualitative Comparisons. In Fig. 9, we show some examples of our qualitative tracking results. For clarity, we only show our tracking results. As shown, our approach can handle different challenges well. Note that there are several categories of objects shown in Fig. 9, such as car, doll, dog, fish, motorbike, and person. This result demonstrates that our tracker is able to track various categories of objects without any prior semantic information.

B. Component Analysis

1) Parameter Analysis: To examine the effect of different values of parameters in each component, we conduct several experiments on the OTB-2013 dataset [45]. We investigate the settings of three critical parameters: score factor σ_S for

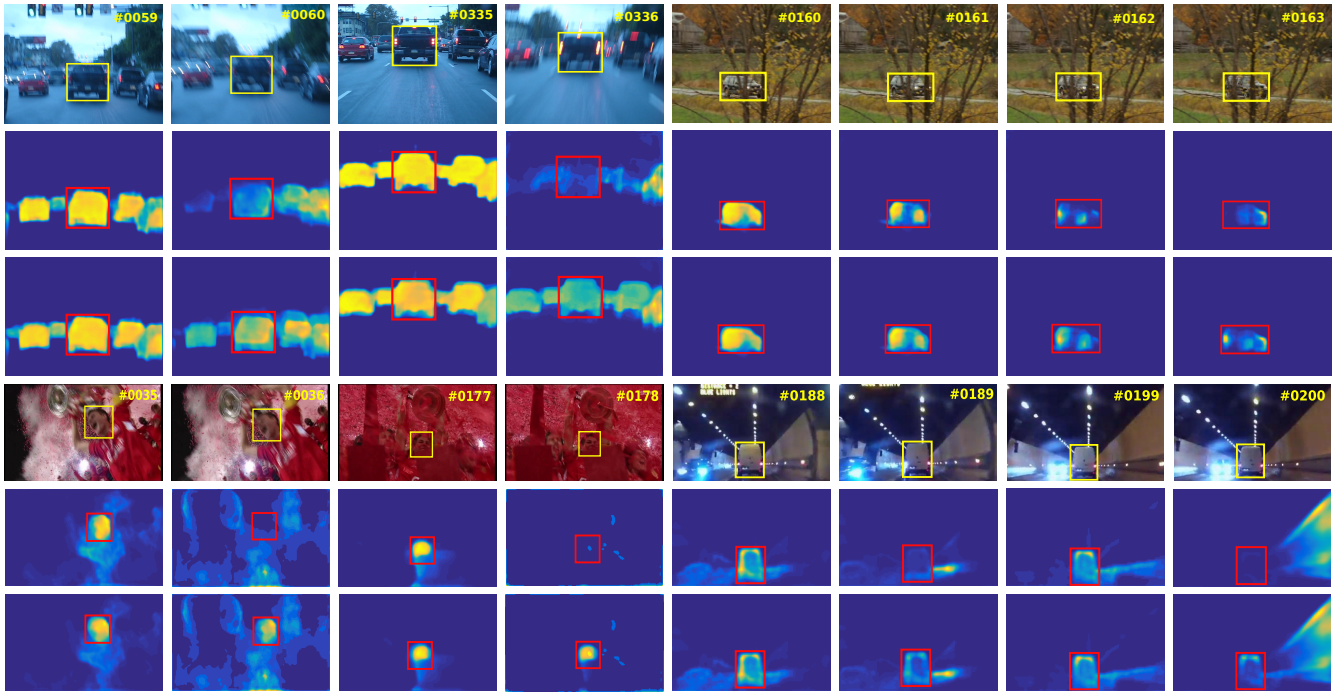


Fig. 13: Qualitative results of temporal evidence passing on 4 sequences from OTB and VOT16 benchmarks. The first and fourth row: the input video frame of sequence *i.e.: BlurCar4, CarScale, Soccer* and *Tunnel* (from left to right, and top to bottom). The second and fifth row: the corresponding per-frame semantic score map of each frame. The third and sixth row: the corresponding smoothed semantic score map via temporal evidence passing. The four sequences represent four challenges: motion blur (*BlurCar4*), occlusion (*CarScale*), background clutter (*Soccer*), and illumination changes (*Tunnel*). These examples show that our temporal evidence passing is robust to various challenges, and generates reliable semantics.

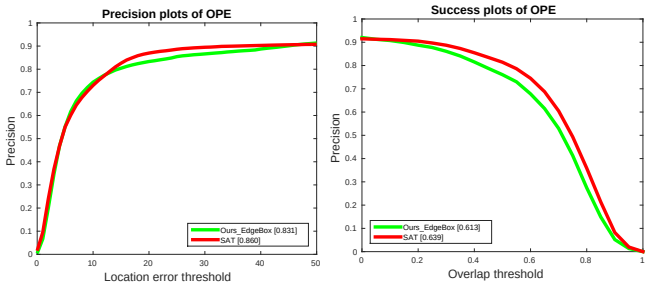


Fig. 14: Comparison of the overall tracking performance of our entire approach and our framework with EdgeBox object proposal on the OTB-2013 dataset.

semantic proposal in Eq. (8), learning rate η_D in Sec. IV-B, and the smoothness parameter γ in Eq. (14). In addition, we also investigate the effectiveness of the proposed semantic proposal generation method. We report the DP and OP of our tracking method under different settings. In Fig. 11 (a), tracking speed in the form of relative FPS (frames per second) normalised to the highest one is reported.

As shown in Fig. 11 (a), setting the score factor σ_S to less than 0.5 will obtain better performance. However, as the score factor increases, the tracking speed becomes faster. With a score factor of 0.5, the tracking speed is 27.3 FPS. As shown in Fig. 11 (b), setting the learning rate to 0.04 for online updating of the semantic distractor-aware colour model obtains slightly more promising tracking performance than other settings. For the smoothness parameter during segmentation, we find that 0.3 will obtain the best result both on DP and OP in Fig. 11 (c). To provide a more clear and standard evaluation of our semantic proposal method, we calculate the average recall on OTB-2013 sequences. Fig. 11 (d) demonstrates that to achieve a recall of 93.9% with an IoU of 0.5, our semantic proposal method only requires 10 proposals. This result may be due to the category-oriented characteristic of semantics.

2) Overall Quantitative Evaluation of Components: In Fig. 12, we show which components of our approach contribute the most to our performance. We present a semantic smoothing method based on the temporal evidence passing algorithm to provide more reliable semantic information to our tracker. Fig. 13 presents qualitative results of our semantic smoothing algorithm. Fig. 12 shows the benefit of this approach compared to the case without semantic smoothing. As shown, we improve the localisation precision by 6.5% and the success rate by 4.5%. We also show the benefit of using scale adaptation compared to a fixed scale. Using our segmentation-based scale adaptation approach significantly improves the overall tracking localisation precision from 68.1% to 82.2% and the success rate from 52.0% to 61.9%.

In addition, to further explore the effectiveness of our object proposal approach based on the category-level semantic map and temporal evidence passing, we employ EdgeBox [46] as our proposal generator, and we integrate it into our tracking

framework. Note that semantics is used several times in our framework; therefore, we remove the related components, such as object proposal generator (Step A in Fig. 1), semantic compatibility during hybrid tracking, and semantic probability in local evidence for segmentation. We feed the top EdgeBox proposal (top 100 in all experiments) into the hybrid tracking and segmentation algorithm, and we obtain the final tracking result (as shown in Fig. 14). The result demonstrates that semantics can effectively improve the performance of the tracking algorithm.

3) Qualitative Evaluation of Semantic Smoothing: Fig. 13 shows qualitative results of the proposed temporal evidence passing algorithm on four video sequences from the OTB [6] and VOT16 [7] benchmarks. These video sequences are very challenging since with the semantic score maps generated by RefineNet [33] without taking temporal consistency into account, the object is lost when motion blur, occlusions, background clutter, and illumination changes appear in the scenario. Our algorithm addresses this difficulty by selectively passing temporal evidence from previous frames and recovers the lost object without introducing any false positives. The smoothed semantic score maps are shown to have higher quality. Furthermore, with the efficient averaging pooling implemented by the convolutional operation, our online temporal evidence passing can run at 500 FPS when the input size is 640×480 pixels and the kernel size is 5×5 on our platform with a CPU processor.

Further, we evaluate the proposed semantic smoothing via temporal evidence passing (TEP) method on non-rigid object tracking (NOT) dataset, which is a benchmark dataset that contains pixel-level annotation of segmentation mask (elven sequences in total) [39]. We evaluate the semantic score of a region Ω by summing up the discriminative scores at every pixel: $\sum_{p\in\Omega} S_t(p)$, and then normalised by the number of pixel in the region. Tab. III shows the semantic score of our method (TEP), no-TEP (no processing on the result of object-level semantic estimation), and AVE (spatio-temporal average on object-level semantics, the temporal bandwidth is 5 frames). It is apparent from this table that the proposed semantic smoothing method improves the semantic score.

4) Qualitative Evaluation of Semantic Object Proposal Generation: Because we aim to apply semantic object proposals to visual tracking, we verify the performance of our approach on OTB-100 dataset. In our setting, the bounding box of the object is the only one ground truth of object proposal. We compare our method to three state-of-the-art approaches: EdgeBox [46], MCG [47] and Deep Mask [48]. We measures the results with two proposal metrics: Firstly, we set the Insertsection over Union (IoU) threshold to 0.5 and vary the number of object proposals from 10 to 10000. Secondly, given 100 proposals, the IoU threshold ranges from 0.5 to 1. The results are shown in Fig. 16. It is observed that when we introduce semantics into proposals, both the quality and localisation precision are enhanced. A probable explanation is that the bounding box of the object is given in the first frame in our method, we can use the category prior and the temporal evidence of sequence to generate high-quality semantic object proposals.

TABLE III: Semantic score of the proposed semantic smoothing method on non-rigid object tracking dataset.

Seqs	no-TEP	Ave	TEP	Seqs	no-TEP	Ave	TEP
cliff1	0.625	0.631	**0.657**	moto2	0.592	0.601	**0.620**
cliff2	0.473	0.472	**0.538**	bike	0.495	0.513	**0.539**
Diving	0.364	0.395	**0.439**	skiing	0.337	0.365	**0.402**
Gym	0.395	0.395	**0.466**	Trans	0.605	0.635	**0.693**
jump	0.416	0.407	**0.439**	volley	0.392	0.405	**0.433**
moto1	0.513	0.520	**0.557**	Mean	0.473	0.485	**0.526**

TABLE IV: Segmentation overlap ratio on non-rigid object tracking dataset.

Seqs	OGBDT	Ours-T	Ours	Seqs	OGBDT	Ours-T	Ours
cliff1	67.8	69.6	**73.2**	moto2	64.5	65.8	**71.6**
cliff2	36.5	38.7	**46.6**	bike	55.0	57.3	**58.2**
Diving	44.1	44.0	**50.7**	skiing	32.1	30.8	**37.1**
Gym	**70.1**	67.6	70.0	Trans	73.8	76.8	**78.1**
jump	43.0	41.9	**47.3**	volley	42.6	43.6	**49.2**
moto1	53.1	56.1	**58.8**	Mean	53.0	53.8	**58.3**

5) Qualitative Evaluation of Scale Adaption via Joint Segmentation: We evaluate the proposed joint segmentation method on NOT dataset [39]. we calculate the pixel-level segmentation overlap ratio of our method (Ours), and OGBDT. Further, we also generate segmentation masks of our hybrid tracking result without scale adaption though simple thresholding followed by finding maximum connected component like our algorithm (we call it 'OurT'). Tab. IV and Fig. 11 (d) show the segmentation overlap ratio and success plots of segmentation results, respectively. The results clearly show that the proposed algorithm is generally good for segmentation.

6) Qualitative Evaluation of Different Object Categories: To evaluate our method for tracking different object categories, Tab. V shows the average VOR and CLE of our method and four state-of-the-art deep learning based trackers (SINT_flow, CFNet, DeepSRDCF, and CF2). We also classify the OTB-100 benchmark dataset into two subsets: seen pre-trained and unseen pre-trained category of object, and evaluate our method separately on these two subsets, the precision plots and success plots are showed in Fig. 15. The results demonstrate that our method performed better than the other methods for well-trained category such as people, cars, cats and dogs. Meanwhile, our method achieves good performance for the remaining categories and categories unseen before.

VII. Conclusion

In this paper, we have demonstrated that we can utilise semantic information to enhance the robustness of key aspects of the tracking framework. We extract a list of high-quality proposal of object masks in each frame through our semantic object proposal generation method for video sequence. A semantics-aware object tracking algorithm is proposed to take full advantage of semantics by our semantic distractor-aware colour model and hybrid search scheme. Finally, we present a scale adaptation approach via segmentation to exploit semantics for handling scale variations during tracking. We achieve state-of-the-art performance on two popular public tracking benchmarks, OTB and VOT16.

TABLE V: Category-based performance: The average VOR and CLE of our methods (SAT and SAT-HCF) and four other methods. Numbers in brackets after the category indicate the number of sequences. Bold marks the best result.

	People(43)		Cars(12)		Cats(3)		Dogs(3)		Birds(2)		Bottles(5)		Motorbikes(2)		Unseen(30)	
	VOR	CLE	VOR	CLE	VOR	CLE	VOR	CLE	VOR	CLE	VOR	CLE	VOR	CLE	VOR	CLE
SINT_flow	0.57	34.14	0.68	16.57	**0.59**	11.86	0.58	12.42	0.47	69.46	**0.66**	14.79	0.57	17.30	0.61	19.60
CFNet	0.55	38.21	0.80	10.51	0.56	14.17	0.73	**4.15**	**0.57**	**27.42**	0.48	62.03	0.39	105.17	0.61	37.13
DeepSRDCF	0.62	20.42	0.82	4.52	0.53	26.03	0.74	4.41	0.49	40.90	0.63	**14.13**	0.52	44.30	**0.66**	**17.71**
CF2	0.55	25.02	0.60	12.14	0.53	14.45	0.56	6.74	0.53	35.13	0.47	60.96	**0.63**	**8.49**	0.59	20.01
SAT	0.60	25.64	0.81	4.46	0.56	25.96	0.74	6.51	0.52	39.20	0.58	38.21	0.50	43.30	0.63	19.27
SAT-HCF	**0.67**	**9.98**	**0.84**	**4.36**	**0.59**	**8.86**	**0.77**	4.32	**0.57**	37.80	0.65	23.50	0.56	18.27	**0.66**	19.10

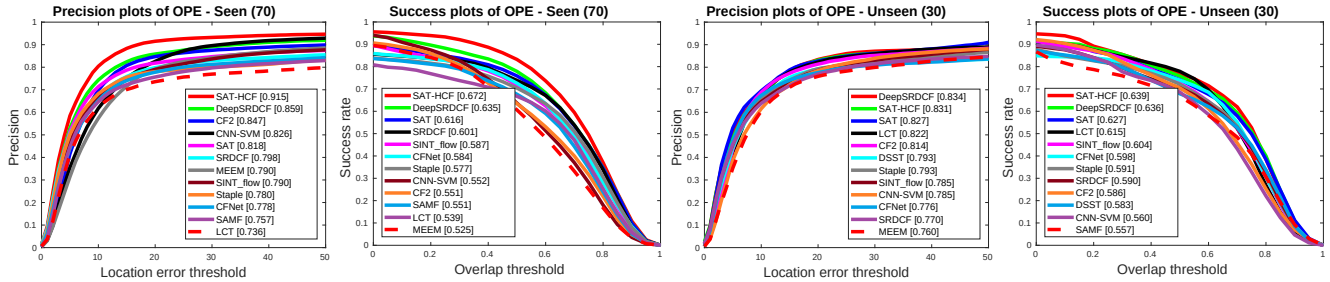


Fig. 15: Precision plot and overlap success plot over two subsets: seen pre-trained category and unseen pre-trained category.

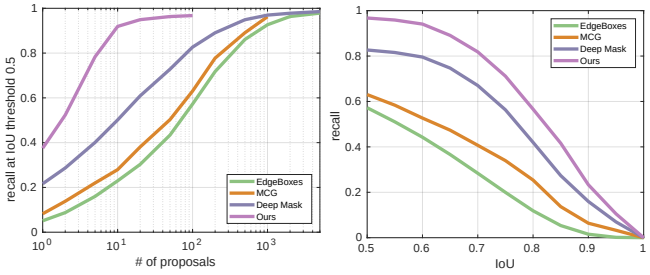


Fig. 16: Comparison of our method with state-of-the-art methods on OTB-100. **Left**: recall versus number of proposals given IoU = 0.5. **Right**: recall versus IoU overlap threshold given 100 proposals.

REFERENCES

[1] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[2] S. He, R. W. H. Lau, Q. Yang, J. Wang, and M. H. Yang, "Robust object tracking via locality sensitive histograms," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 5, pp. 1006–1017, May 2017.

[3] J. 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. 1–1, 2014.

[4] S. Hare, A. Saffari, and P. Torr, "Struck: Structured output tracking with kernels," in *Proc. ICCV*, 2011, pp. 263–270.

[5] 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.

[6] 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.

[7] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojíŕ̀f, and e. al., *The Visual Object Tracking VOT2016 Challenge Results*. Cham: Springer International Publishing, 2016, pp. 777–823.

[8] N. Wang, J. Shi, D.-Y. Yeung, and J. Jia, "Understanding and diagnosing visual tracking systems," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3101–3109.

[9] G. Zhu, F. Porikli, and H. Li, "Beyond local search: Tracking objects everywhere with instance-specific proposals," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 943–951.

[10] Y. Hua, K. Alahari, and C. Schmid, "Online object tracking with proposal selection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3092–3100.

[11] G. Zhu, J. Wang, Y. Wu, X. Zhang, and H. Lu, "Mc-hog correlation tracking with saliency proposal," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI Press, 2016, pp. 3690–3696.

[12] D. Huang, L. Luo, Z. Chen, M. Wen, and C. Zhang, "Applying detection proposals to visual tracking for scale and aspect ratio adaptability," *International Journal of Computer Vision*, vol. 122, no. 3, pp. 524–541, 2017.

[13] J. Zhang, S. Ma, and S. Sclaroff, "Meem: robust tracking via multiple experts using entropy minimization," in *European Conference on Computer Vision*. Springer, 2014, pp. 188–203.

[14] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proc. BMVC*, vol. 1, 2006, pp. 47–56.

[15] 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.

[16] D. S. Bolme, J. R. Beveridge, B. Draper, Y. M. Lui *et al.*, "Visual object tracking using adaptive correlation filters," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2544–2550.

[17] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4310–4318.

[18] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 749–758.

[19] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1430–1438.

[20] M. Danelljan, F. S. Khan, M. Felsberg, and J. Van, de Weijer, "Adaptive color attributes for real-time visual tracking," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2014, pp. 1090–1097.

[21] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1401–1409.

[22] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network." in

International Conference on Machine Learning (ICML), 2015, pp. 597–606.

[23] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 58–66.

[24] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr, "End-to-end representation learning for correlation filter based tracking," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 5000–5008.

[25] R. Tao, E. Gavves, and A. W. Smeulders, "Siamese instance search for tracking," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*. IEEE, 2016, pp. 1420–1429.

[26] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3074–3082.

[27] M. Vondrak, L. Sigal, and O. C. Jenkins, "Dynamical simulation priors for human motion tracking," *IEEE Trans Pattern Anal Mach Intell*, vol. 35, no. 1, pp. 52–65, 2012.

[28] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.

[29] J. Sanchez-Riera, K. Srinivasan, K.-L. Hua, W.-H. Cheng, M. A. Hossain, and M. F. Alhamid, "Robust rgb-d hand tracking using deep learning priors," *IEEE Transactions on Circuits and Systems for Video Technology*, 2017.

[30] J. Xiao, Q. Lan, L. Qiao, and A. Leonardis, "Semantic tracking: Single-target tracking with inter-supervised convolutional networks," *arXiv preprint arXiv:1611.06395*, 2016.

[31] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[32] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[33] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[34] O. Miksik, D. Munoz, J. A. Bagnell, and M. Hebert, "Efficient temporal consistency for streaming video scene analysis," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 133–139.

[35] C. Couprie, C. Farabet, L. Najman, and Y. Lecun, "Convolutional nets and watershed cuts for real-time semantic labeling of rgbd videos." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3489–3511, 2014.

[36] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *European Conference on Computer Vision*. Springer, 2014, pp. 254–265.

[37] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[38] H. Possegger, T. Mauthner, and H. Bischof, "In defense of color-based model-free tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2113–2120.

[39] J. Son, I. Jung, K. Park, and B. Han, "Tracking-by-segmentation with online gradient boosting decision tree," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3056–3064.

[40] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[42] M. B. Blaschko, J. Kannala, and E. Rahtu, "Non maximal suppression in cascaded ranking models," in *Scandinavian Conference on Image Analysis*. Springer, 2013, pp. 408–419.

[43] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *ACM transactions on graphics (TOG)*, vol. 23, no. 3. ACM, 2004, pp. 309–314.

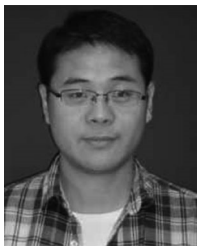
[44] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *IEEE International Conference on Computer Vision*, 2001, pp. 105–112.

[45] Y. Wu, J. Lim, and M. H. Yang, "Online object tracking: A benchmark," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2411–2418.

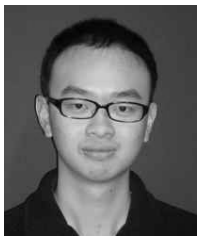
[46] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European Conference on Computer Vision*, 2014, pp. 391–405.

[47] J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 328–335.

[48] P. O. Pinheiro, R. Collobert, Doll, and R. Piotr, "Learning to segment object candidates," pp. 1990–1998, 2015.



Rui Yao received the Ph.D. degree in computer science from Northwestern Polytechnical University, Xi'an, China, in 2013. From September 2011 to September 2012, he was a Visiting Student with the University of Adelaide, Adelaide, SA, Australia. He is currently with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China. His research interests include com-puter vision and machine learning.



Guosheng Lin is an Assistant Professor at School of Computer Science and Engineering, Nanyang Technological University, Singapore. He received his PhD degree at The University of Adelaide in 2015. His research interests are in computer vision and machine learning. He received a Bachelor degree and a Master degree from the South China University of Technology in computer science in 2007 and 2010, respectively.

Chunhua Shen is a Professor at School of Computer Science, The University of Adelaide. His research interests are in the intersection of computer vision and statistical machine learning. He studied at Nanjing University, at Australian National University, and received his PhD degree from University of Adelaide. In 2012, he was awarded the Australian Research Council Future Fellowship.

Yanning Zhang received the B.Eng. degree from the Dalian University of Technology, Dalian, China, in 1988, and the Ph.D. degree from the School of Marine Engineering, Northwestern Polytechnical University, Xi'an, China, in 1996. She is currently a Professor and the Executive Dean with the School of Computer Science, North-western Polytechnical University, Xi'an, China. Her research interests include computer vision and pattern recognition, image and video processing.

Qinfeng Shi received the bachelor and master degrees in computer science and technology from Northwestern Polytechnical University, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree in computer science with a minor in machine learning from Australian National University, Canberra, ACT, Australia, in 2011. He is currently a Senior Lecturer with the School of Computer Science, The University of Adelaide, Adelaide, SA, Australia.