Shallow Cue Guided Deep Visual Tracking via Mixed Models

Fangwen Tu, Shuzhi Sam Ge, *Fellow, IEEE,* and Chang Chieh Hang, *Fellow, IEEE*

Abstract—In this paper, a robust visual tracking approach via mixed model based convolutional neural networks (SDT) is developed. In order to handle abrupt or fast motion, a prior map is generated to facilitate the localization of region of interest (ROI) before the deep tracker is performed. A top-down saliency model with nineteen shallow cues are employed to construct the prior map with online learnt combination weights. Moreover, apart from a holistic deep learner, four local networks are also trained to learn different components of the target. The generated four local heat maps will facilitate to rectify the holistic map by eliminating the distracters to avoid drifting. Furthermore, to guarantee the instance for online update of high quality, a prioritised update strategy is implemented by casting the problem into a label noise problem. The selection probability is designed by considering both confidence values and bio-inspired memory for temporal information integration. Experiments are conducted qualitatively and quantitatively on a set of challenging image sequences. Comparative study demonstrates that the proposed algorithm outperforms other state-of-the-art methods.

I. INTRODUCTION

Visual tracking which focuses on estimating the location of designated target in video clips is one of the most important research topic in computer vision. It plays a crucial role in numerous applications such as security surveillance, robotics, motion recognition and analysis, military patrol and etc.

Most of the existing trackers depending on either generative model [1] which performs template matching or discriminative model [2] [3] which separates foreground object from background by treating the tracking task as binary classification problem employ low-level hand-crafted features for the model construction. This kind of features are not sufficient to capture the semantic information and less robust to the appearance variation due to the limited discriminative power. Recently, there is an increasing trend to incorporate deep neural networks into visual tracking by exploiting the rich hierarchical features [4]. The superior performance is mainly attributed to the captured sophisticated hierarchies and the capability of semantic information expression. While, the direct utilization of deep neural networks such as CNN for online visual tracking confront several challenges. First of all, tracking task usually suffers from insufficient reliable positive instances, since the ground truth is only designated in the initial frame.

F. Tu is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: fangwen_tu@hotmail.com).

S. S. Ge is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 and also with the Social Robotics Lab, Interactive Digital Media Institute (IDMI), National University of Singapore, Singapore 117576, (e-mail: samge@nus.edu.sg).

Additionally, the online training is also computationally intensive which retards the speed. To customize CNN for visual tracking problem without compromising the performance, the reported methods tend to employ off-line trained networks for feature extraction. For example [5], the output of three layers *conv3-4*, *conv4-4* and *conv5-4* from a pretrained VGG-Net-19 network are conserved and imported into a set of correlation filters for the encoding of target appearance. The scheme is based on the fact that early layers retain more fine-grained spatial details that can contribute to a precise localization and last layer usually encodes semantic abstraction that is robust to appearance variations.

In practical implementation, it is almost impossible for the pretrained networks to process the whole image in each frame due to the huge computational load. Thanks to the essence of object tracking that the target's location in consecutive frames is highly correlated, most of the deep trackers only handle a cropped searching window centered at the location in previous frame. However, the window may fail to capture the object in the presence of abrupt or fast motion. To cope with this problem, we propose to introduce a shallow cue based prior map to guide the crop of the ROI window. Compared with deep features, although, shallow features have limited discriminative power, they cost less computational power to obtain and can competently act as an indicator to roughly describe the target. A top-down saliency structure using shallow cues has been verified to be feasible for visual tracking in [6]. In [7], the researchers propose to generate the saliency map by assigning weights to the three conspicuity maps in Itti and Koch's saliency model and conducting a weighted summation. Motivated by this work, we propose a novel prior map generation method by extending the three conspicuity maps to nineteen to enhance the descriptive capability. Moreover, the proposed method also considers the priori information in the previous frame as well as the constraints concerning target's size for the map building. Finally, the ROI is determined by performing template matching operation for all the candidate patches.

In this paper, we utilize the pretrained VGG-Net-16 network for deep feature extraction as [8]. To avoid the distraction in final heat map, [8] proposes two networks using different VGG layers of output features to assist the target identification. But these two networks are both designed to learn the holistic appearance of the target which would be very susceptible to partial occlusion and distracters in some cases. On the other hand, part-based visual tracking scheme [9] has shown its great superiority in dealing with these kind of problems mentioned above. Therefore, we incorporate the part-based idea into deep

tracking for the first time. Apart from the holistic learner network, four local networks are involved to learn one specific part of the target individually and generate four heat maps. The four local heat maps are then used to rectify the holistic map by removing the effect of distracters. In this manner, by adopting four additional hints to separate the real heat region from the fake ones, the risk of drifting is naturally reduced.

To accommodate appearance variation during tracking, an online update procedure is commonly included in deep learning based methods. Target appearance in first frame [8] and current frame [10] are usually collected as the positive samples. Although, the beginning frame contains reliable target depiction, it ignores the variation. And updating using current frame may cause overfitting occasionally to some detected false instances. To overcome this problem, we proposed a bio-inspired prioritised online update scheme. This scheme casts the selection of reliable positive instance as a label noise problem [11] by maximizing the joint probability which stands for an uncontaminated instance that is suitable for updating. The joint probability can be decomposed into two components, the selection possibility and a conditional probability which uses confidence value to evaluate the quality of the selected instance. The selection probability is designed by investigating both the temporal information and the sample quality in a created positive instance pool. A nonlinear weight allocation scheme is tailored to consider the temporal information inspired by bioinformatics that emphasizes the prediction results in beginning and recent frames and tend to forget the mid-term memories. In this manner, this method guarantees the selected positive instance is of high quality, at the same time, can reflect latest appearance of the target.

The main contribution of this work is three-fold and can be summarized as: (i) A novel shallow cue guided global search algorithm is proposed to facilitate the determination of ROI window. The global search can provide a rough target location prediction for the deep trackers to release the computational burden without risking tracking failure when abrupt or fast motion happens. (ii) A mixed deep architecture including both holistic and local models is proposed. Apart from the traditional holistic network, four additional local networks are learnt from distinct components of the object. The holistic heat map is then rectified through the four local hints to make the proposed tracker more robust against distraction as well as overfitting. (iii) To guarantee the positive instance fed into the network for online update is uncontaminated and representative, we cast the selection of instance into a label noise problem. This design intends to emphasize samples from the beginning and current frames which reflects either reliable or up-to-date appearance information. The proposed update scheme ensures the network adaptive to the appearance variation of the target and avoid the performance degradation caused by inappropriate training samples.

II. PRIOR MAP GENERATION

To achieve the tradeoff between computation load and the tracking accuracy, many CNN based approaches [8] [12] crop the ROI centered at the last target location before prorogating to the networks. This strategy inevitably causes the tracking failure when the target undergoes abrupt and fast motion and escapes from the ROI. To cope with this problem, this work adopts a prior map via shallow features to determine the ROI before deep feature extraction.

A. Prior Map Building and Candidate ROI Generation

Different from deep features, shallow features such as color, intensity, steerable pyramid subbands are more accessible and suitable for the rough localization of target. In [13], the researcher investigated thirty-three features distributed from low-, mid- and high-level. The low-level features are proven to be able to depict the fundamental and general characteristics of the object and shows great efficiency compared with mid-, high-level features which need off-line training in advance. By considering the balance between efficacy and speed, nineteen low-level features are extracted to construct the final prior map. The first thirteen features employ the steerable pyramid subbands in four orientation and three scales and denoted as F_{SPi}, $i = 1...13$. Moreover, four broadly tuned color channels (F_R, F_G, F_B, F_Y) as well as the intensity channel (F_I) are also taken into consideration. Finally, a channel of skin color F_{SK} is involved since the tracker is very likely applied to track human targets. A feature map set is constructed by concatenating these nineteen features as $F_{FM} = [F_{SP1}, ..., F_{SP13}, F_R, F_G, F_B, F_Y, F_I, F_{SK}]$. To further accelerate the algorithm, the arrived image are uniformly warped into 200×200 size first. Then we introduce a weight vector $w_S = [w_{SP1}, ..., w_{SP13}, w_R, w_G, w_B, w_Y, w_I, w_{Sk}]$ indicating the correlation degree between each feature map and the target area. The weights are determined in the first frame through an L_2 optimization problem as follow and kept fixed throughout the whole sequence.

$$\min_{w_S} \|X_b^* - F_{FM}'w_S\|_2^2 + \lambda_S\|w_S\|_2^2, X_b^* = \begin{cases} 1, (x,y) \in \phi_b \\ 0, \text{otherwise} \end{cases} \quad (1)$$

where F_{FM}' is obtained by vectorizing the candidate feature set F_{FM}. λ_S is a penalty coefficient and X_b^* represents a binary mask map where ϕ_b is a pixel set indicating the groundtruth bounding box in first frame. The optimal solution to (1) is computed as

$$w_S = (F_{FM}'^T F_{FM}' + \lambda_S I)^{-1} F_{FM}'^T X_b^* \quad (2)$$

With the initialized weights w_S, a top-down saliency map S_{map}' is created through the weighted combination of the nineteen low-level feature maps as $S_{map}' = \sum_{i=1}^{19} w_{Si} F_{FMi}$, where F_{FMi} indicates the ith element of F_{FM}. Different from the traditional center prior [13] which tend to believe human naturally arrange the object of interest near the center of the image, in this work, a revised center prior penalization is proposed to consider the spatial information in the previous frame into the prior map construction. To achieve this, we penalize the combined map S_{map}' with the distance to the center of target in the last frame as $S_{map}^c = C(p_c) \odot S_{map}'$, where $\odot$ is the Hadamard product (element-wise product). $C(p_c)$ denotes a distance penalty matrix defined as $C_{ij}(p_c) =$

$\delta_s \frac{Dis(p_{ij}, p_c)}{\max Dis(p_{ij}, p_c)}$, where p_c and p_{ij} represent the center of the estimated target in the last frame and a pixel position on S'_{map} respectively. $Dis(p_{ij}, p_c)$ returns the Euclidean distance between p_{ij} and p_c. δ_s is a tunable scaler and set as 2. In order to attenuate the noise and simplify the operation, the generated map S^c_{map} is subject to binarization with a threshold σ_b to produce the prior saliency map S_{map}. To proceed the analysis, we give the following observation.

Observation 1: The tracking target or part of it can map to a connected area on the saliency map S_{map}.

The intuition behind this observation is that the whole body or parts of an object with highly discriminative features are usually contiguous. For instance, a face with skin color against the entire head with black hair and red mouth is salient which allows us to locate the rough position of a person's head by simply identifying connected areas with skin color on the saliency map. Under this assumption, an "run-relabel" algorithm is employed to derive connected area set $\Omega_c = [a_{c1}, a_{c2}, ..., a_{cn}]$ with a predefined threshold σ_s for size constraint, which indicates the minimum area that can be selected into set Ω_c and its value is proportional to the size of bounding box. Next, we return the geometric center of each area a_{c1} to form $C_c = [c_{c1}, c_{c2}, ..., c_{cn}]$ and apply the bounding box with same scale and orientation in the last frame to every center element in C_c to crop out the candidate particles $X_c = [x_{c1}, x_{c2}, ..., x_{cn}]$.

To determine the final ROI region, we adopt a simple raw pixel template matching mechanism with the image patch x_1^* extracted using the ground truth in the initial frame. The candidate with largest observation likelihood c_i as calculated in (3) is regarded as the predicted ROI in the current frame.

$$c_i = \delta_c \exp(-\|x_{ci} - x_1^*\|_2^2) \tag{3}$$

where $\delta_c = 0.01$ is used to ensure c_i belonging to the range of [0,1]. In order to keep the algorithm as lite as possible, we fix the template x_1^* to avoid improper update. This strategy may lead to drift problem if the target undergoes severe appearance variation and (3) fails to give sufficient insight on the selection of correct ROI. To cope with this problem, we introduce an extra threshold σ_c to judge whether the determined ROI is applicable. If the maximum confidence $c_* > \sigma_c$, the deep tracker is performed centered at the determined ROI, otherwise, the cropped image patch centered at the estimated location in previous frame is propagated into the deep tracker.

III. DEEP TRACKING WITH MIXED MODEL

Deep neural network such as CNN has shown its powerful capability in encoding the target appearance without human's guide. But, different from traditional classification and detection problems, the target object in visual tracking usually undergoes dynamic appearance variations due to partial occlusion, illumination changes and etc, which make it inaccurate sometimes impossible to reply on sole holistic information when developing trackers. In this regard, collaborative model [2] with both local and holistic features have been proven to be competent in coping with the challenges mentioned above. Motivated by this, we propose a mixed model based deep tracking scheme to achieve a more robust performance.

A. Proposed Network Architecture

CNN architectures such as AlexNet, VGG-Net, GoogLeNet are efficient in extracting discriminative and semantic features if they are trained with sufficiently large scale datasets. It is inappropriate to employ these deep networks directly on visual tracking task to estimate the target location since (i) we are lack of sufficient samples for training, (ii) the online training of such deep network is very time-consuming. To handle these contradictions, [8] proposes to use a 16-layer VGG-net for capturing the visual representation related to the object. Two small networks are trained specifically with holistic features on the top of VGG to produce the foreground heat map regression. In this paper, we adopt a similar framework as shown in Figure 1.

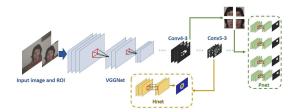The determined ROI of input frame is firstly propagated

Fig. 1. The architecture of proposed mixed deep tracker

into a VGG-Net that has been pretrained on large-scale ImageNet dataset with category label [14]. With the forward propagation of the network, though, the semantical discrimination power increases, the spatial resolution reduces due to the pooling operation. For this reason, features obtained from Conv4-3 layer showing more intra-class discrimination power and higher resolution (46×46) are more suitable for detail description, thus, extracted as the input of "part-based" network *Pnet*. *Pnet* consists of four two-layer convolutional networks *Pnet1*, *Pnet2*, *Pnet3* and *Pnet4* using ReLU nonlinearity without any pooling operation. They are supposed to learn one specific part of the target and produce four corresponding heat maps $M_{P1} \sim M_{P4}$. On the contrary, features (23×23) from Conv5-3 layer focusing on the semantic category information are fed into the holistic network *Hnet*. *Hnet* has the same architecture with *Pnet*. It mainly handles the detection of the whole body of the object in target's category and generates a holistic heat map M_H.

Before the initialization of *Hnet* and *Pnet*, we perform a feature map selection on the output of Conv4-3 and Conv5-3 in order to improve the detection accuracy by removing noisy features [8]. We similarly construct $Hnet_S$ and $Pnet_{S1} - Pnet_{S4}$ networks consisting of one dropout layer with dropout ratio of 0.3 and one convolutional layer with kernel size of 3×3 to predict the groundtruth heat map M_{HT} and $M_{PT1} \sim M_{PT4}$. M_{HT} is built by performing a 2-dimensional Gaussian distribution centered at the location of target object in the first frame. As for

$M_{PT1} \sim M_{PT4}$, by considering the performance and time consumption simultaneously, we divide the target into four parts: top-left(TL), top-right(TR), bottom-left(BL) and bottom-right(BR) as shown in Figure 1. The intuition behind this is that we hope to train each $Pnet$ and $Pnet_S$ to learn one component of the object in order to handle the cases that some parts are subject to occlusion or severe deformation while others not. Each heat map M_{PT} is constructed using one part through the 2-dimensional Gaussian distribution. The variance of the distribution is proportional to the corresponding object size. The loss functions are defined as $L_S = \|M - M_T\|^2$ where M stands for the predicted maps, i.e. M_H, M_P. M_T is the groundtruth maps. The impact of each input feature f_i is evaluated through its effect on the loss function L_S. Quantitatively, a two-order Taylor expansion is employed to measure the impact as $\delta L_S = \sum_i \frac{\partial L_S}{\partial f_i} \delta f_i + \frac{1}{2} \sum_i \frac{\partial^2 L_S}{\partial f_i^2} (\delta f_i)^2 + \frac{1}{2} \sum_{i \neq j} \frac{\partial^2 L_S}{\partial f_i, \partial f_j} \delta f_i \delta f_j$. After approximating the Hessian matrix with a diagonal matrix and set $\delta f_i = 0 - f_i$, the impact of f_i denoted as δL_{Si} can be simplified as $\delta L_{Si} = -\frac{\partial L_S}{\partial f_i} f_i + \frac{1}{2} \frac{\partial^2 L_S}{\partial f_i^2} f_i^2$. The final score SC_i is calculated by summing δL_{Si} value at each pixel location (x, y). The top $N_S = 384$ features are selected as the salient ones. Readers are recommended to refer to [8] for more details about the feature selection strategy. With the salient features, the five networks (*Hnet* and four *Pnet*) are initialized with the same groundtruth maps respectively.

B. Localizing the Target

For most input images, the output holistic heat map M_H is able to locate the target because of the strong discriminative power of deep features. However, M_H may fail to provide a "pure" indication with single peak on the heat map in the cases that distracters appear or insufficient online finetuning due to the lack of reliable samples as depicted in Figure 2 (b). For the *MotorRolling* case, M_H gives two peak areas peak1 and peak2. Peak2 corresponds to the real target, while peak1 incorrectly takes the grey wall as the target. This situation may be caused by the similar color and texture between the rider and the wall. In the case of *Deer*, the holistic map provides two peak areas when the target approaches to another deer with similar appearance. Apparently, M_H cannot be applied directly for target searching under this situation. Hence, when a new M_H is generated, it is passed through a watershed approach [15] to search for regional maximum areas with peaks located at (x_{hi}, y_{hi}) where $i = 1, 2...n_h$. Further, we remove the areas which do not satisfy $M_H(x_{hi}, y_{hi}) \geq 0.8 \max(M_H(x_{hi}, y_{hi}))$ and remains n_h' maximum areas. If $n_h' > 2$, it reveals that more than one peaks exist in the map M_H and it should be rectified before localizing the target. In this paper, we propose to take the advantage of the four M_Ps for the rectification since they can provide more intra-class specifics related with the target. Still, we obtain the peak locations of individual M_P in the same way with M_H and discard the maps with more than one peaks. The peaks' coordinates are recorded as $\{(x_{p1}, y_{p2})...(x_{pn_p}, y_{pn_p})\}$ where $n_p \leq 4$. Next, the peaks from part-based maps vote for each peak in the holistic map

with their Euclidean distance and the score SH_i of each peak is calculated through the average of the distance. The voting procedure can be expressed as

$$SH_i = \frac{\sum_{j=1}^{n_p} \sqrt{(x_{hi} - x_{pj})^2 + (y_{hi} - y_{pj})^2}}{n_p}, \ (i = 1..n_h') \quad (4)$$

The peak area with smallest score SH_i is regarded as valid indication area and others are filled with zeros. Figure 2 present a clear example for the afore-mentioned rectification operation. The upper row image of (c) shows the heat maps using part-based hints. It can be observed that in M_{p2}, there is more than one peaks exist and we cannot tell which one corresponds to the real target, thus, this map is discarded. Moreover, though there also exist distracters in M_{p4}, the peak values of the distracters do not exceed 80% of the maximum peak value, the peak in this map is valid for the voting stage. In this manner, the M_H map can be efficiently rectified into a single-peak indicator as shown in (d).

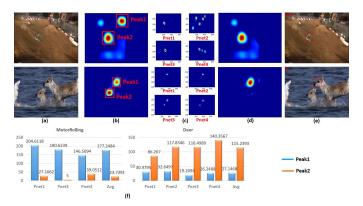We then perform particle filter on the rectified M_H to

Fig. 2. Case study of proposed mixed deep tracking on *MotorRolling* and *Deer*. First and second row present: (a) input image, (b) heat map M_H from Hnet, (c) heat maps $M_{P1} \sim M_{P4}$ from Pnet1~4, (d) rectified M_H, (e) tracking results. (f) voting distance between peaks in M_ps and M_H.

decide the target location $\hat{X}(x, y, \sigma)$ which is described by the center coordinates as well as the scale. Given the center of the ROI c_{ROI} and scale in the previous frame σ^{t-1}, the locations of target candidates are assumed to be subject to a Gaussian distribution centered at $X_{ROI}(c_{ROI}, \sigma^{t-1})$ with a fixed diagonal covariance Ψ. The corresponding patches of candidates on the heat map are warped into the same size before the average heat value v_i inside individual patch is calculated. The confidence of each candidate is finally derived by $C_i = v_i \sigma_i^\gamma$ where σ_i stands for the scale of *i*th candidate and $\gamma < 1$ is an important coefficient that controls the scale compensation on v_i in the confidence. It γ is too large, obviously, candidates with large scale will be selected as the target. On the contrary, a small γ will lead to a small scale of bounding box. Therefore, after trial and error, it is set as 0.7 in this paper. Candidate with highest confidence value C_* is determined as the location of target. In practical implementation, we observe that although this scheme can produce an accurate center location, the bounding box is not tight. To remedy this, we make a minor modification on the scale σ_* into $\sigma_* = \sigma_*^1 (C_*/C_*^1)^{\lambda_\sigma}$ where σ_*^1 and C_*^1 are the

scale and maximum confidence in the first frame respectively. $0 \leq \lambda_\sigma \leq 1$ adjusts the modification ratio. The efficacy of this modification is verified in the experiment section.

IV. PRIORITISED ONLINE UPDATE SCHEME

In order to adapt to the appearance variation of the target, the top added networks should be online updated. Due to the possible in-plane rotation of the target, it is difficult to determine the relative position and orientation of the four parts of *Pnet*s in current frame. To avoid an overcomplicated update, we adopt a conservative update strategy for the *Pnet*s by fixing them after the initialization. The update scheme in this section focuses on the holistic network *Hnet*.

In this paper, the positive training instances are selected among the tracking results in individual frame. The key condition to ensure an efficient update is to search "good" training samples which are not contaminated by occlusions or misalignment. We cast this problem into a label noise problem [11] by introducing a joint probability $P(y_t^*, \xi = 1)$. y_t^* stands for the estimated result in frame t. $\xi = 1$ denotes the sample is not contaminated and ready for update, otherwise $\xi = 0$. $P(y_t^*, \xi = 1)$ describes the likelihood that the selected positive sample within the tracked frames is suitable for the online updating. Considering the chain rule, we have

$$P(y_t^*, \xi = 1) = P(t, \xi = 1) = P(\xi = 1|t)P(y_t^*) \quad (5)$$

Our goal is to appropriately design the conditional probability $P(\xi = 1|t)$ and selection possibility $P(y_t^*)$ such that $P(y_t^*, \xi = 1)$ is as large as possible. Firstly, the challenge is how to estimate $P(\xi = 1|t)$, in other words, we hope to quantitatively measure the quality of a sample. In this regard, the optimal confidence value C_* is employed to achieve the measurement based on the observation that a lower C_* is usually caused by (i) full or partial occlusion (ii) severe scale variation. A case study on *FaceOcc1* is conducted as shown in Figure 3 to explain the observation (i). When no occlusion occurs, the optimal confidence usually corresponds to a high value, otherwise, a lower value is rewarded. For (ii), the reason is straightforward that the shrink of scale will definitely lead to a vanish of the value as well as the size of corresponding heat area on the heat map. Small scale of the target inevitably contains less information and is considered not "good" enough for updating. With the analysis above, a sample is determined as contaminated when either (i) or (ii) situation occurs. Therefore, we predict $P(\xi = 1|t)$ as $P(\xi = 1|t) \propto C_*^t$. To ensure a high confidence value of each selected y_t^*, we create a pool $\mathbb{Y}^* = \{y_{t(1)}^*, y_{t(2)}^*, ..., y_{t(N_y)}^*\}$ containing N_y samples with large C_*. The insertion of new element y_t^* should satisfy either (i) $C_*^t > \min\{C_*^{t(1,..,N_y)}\}$ or (ii) $C_*^t / \max\{C_*^{t(1,..,N_y)}\} > 0.85$. Once the condition is satisfied, we replace $y_{t(\arg\min_i\{C_*^{t(1,.i.,N_y)}\})}^*$ with y_t^*.

As for the selection probability $P(y_t^*)$, we design it by mainly considering two criteria: (i) uncontaminated samples possess larger selection probability, (ii) good image temporal location is rewarded with larger selection probability. (i) focuses on the evaluation of one specific sample and can be similarly quantified using C_*. (ii) bases on the intuition that
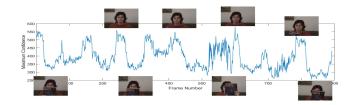

Fig. 3. Case study on video sequence of *FaceOcc1* containing 892 frames for quantitative measurement of $P(\xi = 1|t)$ using C_*. When the wave troughs appear, the face is usually partially or fully occluded as the lower image row shows. Rising of C_* is caused by recovery from the occlusion. Crest occurs when the face is completely unoccluded.

samples extracted from the beginning few frames contain the most accurate and reliable message on the object, moreover, the recent few images ahead of the current frame contain the most up-to-date appearance description about the object. Thus, we propose the following quadratic-like weight allocation function for the purpose of emphasizing the most original and recent samples and assign smaller weights for the mid-sequence ones.

$$W_i^t = \frac{4(1-\theta)}{t(t-2)}\mathscr{T}^2(i) + \frac{4(t+1)(\theta-1)}{t(t-2)}\mathscr{T}(i) + \frac{t-4\theta+2}{t-2} \quad (6)$$

where W_i^t denotes the temporal weight for ith element among $\mathbb{Y}^*$ in frame t. $1 \leq \mathscr{T}(i) \leq t$ records the frame number when the ith sample is added into $\mathbb{Y}^*$. $\theta < 1$ adjusts the smallest value of the weights which always occur at the middle of the tracked video sequence. Recalling the specific sample quality measurement index $C_*^{t(1,..,N_y)}$, a comprehensive evaluation index I_i^t is derived by $I_i^t = W_i^t * C_*^{t(i)} / \max\{C_*^{t(1,..,N_y)}\}$. In this manner, we rationally assign the weight for each sample in the pool considering specific (individual quality) and general (sequential characteristics) measurements. The selection probability $P(y_t^*)$ is finally determined based on I_i^t, i.e. $P(y_t^*) = \dfrac{I_i^t}{\sum_{i=1}^{N_y} I_i^t}$. The design of $P(\xi = 1|t)$ and $P(y_t^*)$ is able to ensure a high quality positive instance for the update. Different from positive sample generation, negative sample is captured directly by removing the estimated target in current frame. Since the distracters in background may belong to distinct categories and is subject to high uncertainty, there is no need to also create a negative sample pool like $\mathbb{Y}^*$. Using the negative samples in current frame is sufficient and easy-operating.

Update condition: we investigate two issues to determine if the update should be executed, which are the tracking quality of the target and the existence of distracters in background. Concretely, update is activated upon the satisfaction of the following two conditions simultaneously.

(i) Select one positive sample $y_{t(n)}^*$ in $\mathbb{Y}^*$ randomly according to the possibility $P(y_t^*)$ and $C_*^{t(n)} > 2C_*^t$.

(ii) There appears more than one peaks in heat map M_H.

In order to avoid updating the network over-frequently, we check these two conditions every ten frames. Once the update is required, the network is online tuned by minimizing the

following cost function.

$$\min \beta_W \|W_H\|_F^2 + \sum_{x,y} \Big[\mathscr{L}^n(x,y) Tru\big(M_H^n(x,y) - M_T^n$$
$$(x,y)\big)^2 + \big(1 - \mathscr{L}^t(x,y)\big) Tru\big(M_H^t(x,y) - M_T^t(x,y)\big)^2 \Big] \tag{7}$$

where W_H denotes the convolutional weights in Hnet. $\mathscr{L}^i(x,y)$ stands for the label of pixel (x,y) in ith frame (1 for foreground, 0 for background). M_T^i is the Gaussian distributed groundtruth map. $Tru(\bullet)$ represents the truncated loss to accelerate the online updating process [16]. It is defined as

$$Tru(\bullet) = |\bullet| \left(1 - \mathbf{l}\left[|\bullet| \le \frac{\epsilon}{(k + \mu\phi^i(x,y))} \right] \right) \tag{8}$$

This truncation is based on the observation that the tracking performance is more sensitive to the prediction error on positive samples than negative ones.

V. Experiments

The proposed SDT is implemented in MATLAB and run at 1.5 fps on an Intel Core i7-4710HQ 2.5GH PC with 16GB memory and NVIDIA Geforce GTX 860M GPU. The deep networks are built and trained on a wrapper of Caffe framework. The thresholds for prior map generation are set to $\sigma_b = 0.2$, $\sigma_s = 0.4 * w * h$ and $\sigma_c = 0.2$, where w and h are the width and height of current bounding box. The kernel sizes of the two convolutional layers in $Hnet$ are 9×9 and 5×5 with respective padding of 4 and 2. The four $Pnets$ share the same architecture with $Hnet$. The output features from Conv5-3 are resized into 46×46 by linear interpolation before importing to $Hnet$. $Hnet$ and $Pnets$ are initialized in the first frame using back-propagation for 100 iterations. The number of particles sampled for target localization is 700. The parameter θ in (6) is 0.7 and the truncation parameters in (8) are set to $\epsilon = e_{\max}$, $k = 20$, $\mu = 30$. The capacity of positive sample pool $\mathbb{Y}^*$ is $N_y = 10$.

Since the generation of prior map depends on all the three channels of a image, we conduct the test on twenty challenging color video sequences in [17]. And the results are compared with fifteen state-of-the-art trackers including L1APG [1], IVT [18], Frag [19], KCF [20], Struck [3], MTT [21], ASLA [9], KMS [22], CXT [23], CSK [24], DFT [25], LOT [26], SCM [2], TLD [27] and FCNT [8] for qualitative and quantitative study.

A. Ablation Study

This section conducts several tests on the main components of SDT for the performance verification. Figure 4 presents the experiment on the video sequence of *DragonBaby*. (a) and (b) are the input image of frame 44 and the produced saliency map S_{map}. Since we are supposed to track the head of the baby in this task, the weight of the channel of skin color w_{Sk} dominates over other channels according to the histogram in (j). Moreover, due to the similar color essence, weight of yellow channel w_Y also possesses relatively large value (half of w_{Sk}). This leads to the existence of distracter in (b) because

of the yellow leaves in background. Thanks to the template matching operation, which allows us to separate these two candidate ROIs with different confidence c_*. The upper salient area which represents the real target has higher confidence of 0.3361, while, the fake one only possesses 0.2816. In this manner, the ROI can be accurately captured in the presence of distraction and excellent tracking performance is guaranteed in (c). (d) and (e) show two consecutive frames 45 and 46 where the target is subject to abrupt motion from the upper left corner to the middle. (f) reveals that the location of target after the large motion can be uniquely indicated in the prior map S_{map} and no template matching separation is required. (g) shows the determined ROI with and without prior map's guide. The lower ROI is cropped centered at the location in frame 45 and it can be observed that target disappears in the patch, thus, the tracking fails as shown in (i). On the contrary, S_{map} can help to relocate the center of ROI as demonstrated in upper image of (g) and in turn contributes to an accurate tracking in (h).

To investigate the necessity and performance of the pri-

Fig. 4. Efficacy validation of prior map guidance on video sequence of *DragonBaby*. (a) Input image of frame 44, (b) S_{map} map for frame 44, (c) tracking result in frame 44, (d) input image of frame 45, (e) input image of frame 46, (f) S_{map} map for frame 46, (g) determined ROI of using prior map (upper figure) and not (lower figure), (h) tracking result in frame 46 using prior map, (i) tracking result in frame 46 without prior map, (j) weight allocation of nineteen features in prior map generation.

oritised update scheme for $Hnet$, a series of experiments on different video sequences using distinct update schemes are carried out and the results are shown in Table I. "NoUdt", "Udt1f" and "UdtCf" stand for "no update", "update using positive sample in first frame" and "update using positive sample in current frame". The negative samples in (7) among all the experiments are extracted from the current frame as in the proposed update scheme. The performance is evaluated via overlap rate and center error and the best performance is highlighted in red font. Generally, the proposed scheme can achieve better accuracy compared with other three schemes. Specifically, for the cases that the target undergoes severe appearance variation such as *Dog*, *MotorRolling* and *Skiing*, SDT shows its superiority since it can rationally select the positive samples without involving too much noise and ensure the network adaptive to the changes. For those no severe appearance variation occurs such as *Couple* and *Deer*, "NoUdt" can already handle the tracking well. The involvement of online update will not degrade the performance using the proposed scheme. When the target is subject to occlusions as in *Bird2*, the two insertion conditions for positive sample pool $\mathbb{Y}^*$ guarantee that only high-quality samples are available as the candidates for update, hence, the network's degradation

is alleviated.

TABLE I
STUDY OF DIFFERENT UPDATE SCHEMES

	NoUdt		Udt1f		UdtCf		SDT	
	OR	CE	OR	CE	OR	CE	OR	CE
Bird2	0.68	11.67	0.69	10.98	0.71	10.26	0.72	10.23
Couple	0.69	5.08	0.66	5.02	0.65	4.72	0.69	4.27
Deer	0.72	7.78	0.72	7.70	0.72	7.63	0.72	7.58
Dog	0.49	12.07	0.54	7.56	0.46	6.83	0.55	6.89
MotorR	0.58	14.89	0.61	14.44	0.59	15.16	0.65	14.13
Skiing	0.51	5.33	0.51	4.22	0.55	3.64	0.55	3.99
Average	0.61	9.47	0.62	8.32	0.61	8.04	0.65	7.85

Note: OR and CE are short for overlap rate and center error respectively.

B. Qualitative Evaluation

Qualitative investigation is carried out on twenty challenging video sequences with six state-of-the-art trackers. The result is reported in Figure 5 and the detailed analysis follows below.

Background clutter: cluttered background brings the challenge by degrading the contrast between foreground and background. In *Couple*, the running cars, jeep as well as the tree in the background may lead to drifting of the trackers. Only the proposed SDT is able to perform favorably by the end of the sequence since deep features equip the tracker with excellent discriminative power to separate the background and foreground. In *Deer*, the sequence undergoes both distracters and fast motions. Algorithms based on raw pixel or PCA template such as L1APG, SCM and ASLA are not efficient enough to handle the similar context objects nearby. All the tracking by detection methods (Struck, FCNT, SDT) can complete the task perfectly due to the discriminative capability of nonlinear classifier. Moreover, the four part-based heat maps facilitate to locate the real target from the distracters as mentioned in III-B. In *MountainBike*, the biker rides across the gap with varying postures. The rocks and bushes inside the gap as interference factors bring extra difficulty to the trackers. CXT and L1APG drifts after the biker's landing. FCNT does not produce a tight bounding box in the last few frames since it include noise in the background into the heat map due to imperfect online update.

Abrupt motion and motion blur: To tackle fast motion is usually very challenging for visual tracking task since trackers tend to search the target near the location in previous frame to achieve a good tradeoff between performance and efficiency. As can be seen in the first figure of *DragonBaby*, all the trackers fail to follow the target due to the abrupt motion except for our method with prior map's guide. An acceptable result can still be obtained when the target is subject to sudden scale variation as the second image shows. This benefits from the minor modification of scale parameter σ_* after the determination of target's location. Moreover, the tracking task becomes more complicated since abrupt or fast motion will always lead to motion blur which makes it ineffective to perform template matching for traditional methods. The robustness of the deep tracking against blur image can be demonstrated by *BlurFace*, *Boy* and *Human7*. The target can be captured throughout the whole sequence although slight out-of-plane rotation and scale variation occurs in *Boy* and *Human7* at the same time.

Full or partial occlusion: unlike some particle filter based methods such as L1APG using trivial templates to reconstruct the occluders, tracking-by-detection methods have no specific designed mechanism to cope with occlusions. The tracker may not be able to produce a reliable holistic heat map when occlusion occurs. Therefore, the proposed mixed model shows its superiority in dealing with partial occlusion since it can learn the unoccluded parts to rectify the holistic map for better performance. For example the last frame in *Girl*, when the girl's face is partially blocked by the man, the proposed tracker can still identify the real target through the assistance of part-based heat maps. While, the holistic feature based FCNT method drifts to the man's face incorrectly. Other part-based algorithm such as ASLA also performs well in this case. When full occlusion happens, we cannot indicate the target on both the image and the heat maps. Under this situation, to avoid drifting we involve a little trick by fixing the bounding box when extreme low confidence C_*^t is detected. Then, the tracker will recapture the target after the full occlusion disappears with increased C_*^t. This trick is sample but efficient as shown in the second image of *David3*. FCNT has drifted to the lower half of the man after fully blocked by the tree, while, the proposed SDT can still achieve the tracking. Finally, the prior map and deep features' discriminative power makes it possible to track the object again in case of drifting due to the occlusion as shown in *David3* and *Jogging*.

Non-rigid object deformation: non-rigid deformation indicates the severe appearance variation when human, animal moves. Trackers need to develop proper update scheme to capture the latest changes to improve the adaptiveness. In *Dog*, although most of the algorithms can track the target when the dog runs towards the woman, some of them may drift a lot when the dog shaking its tail under the coverage of her shadow. The slight illumination change brings fatal interference to these trackers. In *Gym*, the athlete performs varying gymnastic movement in the field throughout the video, the proposed tracker can always follow the torso of hers and some trackers may slightly miss the target at the end of the task. In *Skiing*, non-rigid deformation happens together with scale variation and background clutter when the skier flights over the hillock in front of the camera. Only the two deep learning based methods SDT and FCNT can finish the tracking. Since we perform scale modification after the target localization, a tighter bounding box can be achieve as shown in the last figure.

Illumination variation and rotation: The proposed tracker also shows pretty good capability in handling illumination changes as reported in *Trellis* when the guy walks under the light varying condition. CXT which depends on the context elements exploration is very susceptible to the changes and drifts almost at the beginning of the test. Other methods employing raw pixels or haar-like feature are all sensitive to the illumination variation as the result report. The ability that the proposed scheme can handle in-plane or out-of-plane rotation attributes to the architecture. The holistic map learns robust high-level semantic information and the part-based

(a)

(b)

— CXT — L1APG — SCM — Struck — ASLA — FCNT — SDT

Fig. 5. Tracking result screenshots of seven trackers. (a) Video sequences of *Bird2*, *BlurFace*, *Boy*, *Couple*, *Crossing*, *David*, *David3*, *Deer*. (b) Video sequences of *Dog*, *DragonBaby*, *FaceOcc1*, *Girl*, *Gym*, *Human7*, *Jogging*, *MotorRolling*, *MountainBike*, *Skiing*, *Trellis*, *Walking*.

learners are in charge with components of the target which is especially significant in the presence of in-plane rotation as demonstrated in *MotorRolling*. Moreover, SDT can also handle mirror rotation in *Brid2*, *David3* or even more comprehensive rotation scenario in *David*, *Girl*.

C. Quantitative Comparison

In order to achieve a comprehensive evaluation, quantitative comparison experiment is conducted with other fifteen state-of-the-art algorithms in terms of success score and precision score [17]. The experiment results are reported in Table II and III where the top three scores are highlighted in red, blue and green fonts.

It can be observed that the proposed tracker performs favorably on most of the video clips and outperforms other state-of-the-art methods. The superior result benefits a lot from the robust attributes of deep features. The same situation holds for FCNT. Furthermore, the tracking framework introduced in this paper guarantees a performance improvement compared with FCNT especially in terms of success scores. Apart from the deep learning based methods, ASLA possesses the third place in average success scores test, which demonstrates that part-based model is always efficient to facilitate the algorithms in dealing with challenging tracking tasks. It is worth noticing that we test all the video clips with three channels since the prior map relies on color images. However, if there is priori knowledge that the target will not undergoes severe abrupt motion or other challenges that need to determine the ROI in advance such as pedestrian tracking task or certain security surveillance systems, the proposed system can also be deployed on greyscale images without the prior map generation phase.

VI. CONCLUSION

In this paper, a shallow feature guided deep tracking algorithm has been developed with mixed models. To dynamically determine the ROI fed into the deep networks, nineteen hand-crafted shallow features are employed with learnt weights to generate a prior map. The pre-determined ROI helps to handle target's abrupt and fast motion. It is then passed into a novel mixed model based deep tracker with holistic learner to detect the semantic information in the image and part-based learners to handle the low-level discriminative features. The part-based maps facilitate to rectify the holistic heat map in the presence of occlusions, distracters and other interference factors. Finally, a prioritised update scheme is introduced for the online finetuning to alleviate degradation of the networks. A series of comprehensive experiments are conducted and demonstrate the superiority of the proposed tracker.

REFERENCES

[1] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1830–1837, IEEE, 2012.

[2] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparse collaborative appearance model," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 2356–2368, 2014.

[3] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. Torr, "Struck: Structured output tracking with kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2096–2109, 2016.

[4] K. Zhang, Q. Liu, Y. Wu, and M.-H. Yang, "Robust visual tracking via convolutional networks without training," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1779–1792, 2016.

[5] 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*, pp. 3074–3082, 2015.

[6] W. Li, P. Wang, and H. Qiao, "Top–down visual attention integrated particle filter for robust object tracking," *Signal Processing: Image Communication*, vol. 43, pp. 28–41, 2016.

TABLE II
AVERAGE SUCCESS SCORES

	SDT	L_1APG	IVT	Frag	KCF	Struck	MTT	ASLA	KMS	CXT	CSK	DFT	LOT	SCM	TLD	FCNT
Bird2	0.879	0.101	0.434	0.232	0.252	0.494	0.090	0.798	0.262	0.101	0.454	0.697	0.070	0.676	0.303	0.777
BlurF	0.990	0.385	0.119	0.661	0.140	0.612	0.357	0.091	0.780	1	0.998	0.281	0.359	0.135	1	0.998
Boy	0.962	0.566	0.294	0.518	0.318	0.938	0.435	0.420	0.794	0.962	0.814	0.481	0.629	0.438	0.601	0.968
Couple	0.779	0.550	0.085	0.492	0.235	0.514	0.492	0.064	0.500	0.085	0.085	0.450	0.450	0.078	0.178	0.671
Cross	0.817	1	0.191	0.366	0.300	0.800	0.216	0.991	0.233	0.316	0.166	0.508	0.450	0.991	0.425	0.816
David	0.540	0.428	0.581	0.057	0.199	0.189	0.233	0.646	0.023	0.482	0.515	0.189	0.084	0.318	0.518	0.634
David3	0.889	0.329	0.507	0.678	0.956	0.337	0.095	0.934	0.718	0.111	0.189	0.662	0.722	0.456	0.107	0.865
Deer	0.901	0.760	0.028	0.126	0.774	0.957	0.704	0.042	0.309	0.478	0.957	0.309	0.042	0.042	0.281	0.901
Dog	0.307	0.078	0.094	0.039	0.047	0.070	0.063	0.189	0.047	0.401	0.047	0.047	0.362	0.378	0.244	0.039
DragB	0.788	0.238	0.230	0.362	0.053	0.088	0.132	0.132	0.398	0.336	0.212	0.11	0.495	0.097	0.070	0.725
Occ1	0.924	0.992	0.871	1	0.956	0.970	0.698	0.915	0.873	0.651	1	0.698	0.245	0.998	0.191	0.937
Girl	0.816	0.440	0.168	0.488	0.482	0.302	0.858	0.606	0.246	0.598	0.294	0.184	0.470	0.324	0.258	0.328
Gym	0.144	0.003	0.003	0.136	0.160	0.015	0.010	0.006	0.109	0.088	0.010	0.014	0.011	0.087	0.138	0.109
Human7	0.692	0.516	0.264	0.128	0.420	0.152	0.156	0.240	0.092	0.276	0.160	0.152	0.404	0.240	0.820	0.144
Jogging	0.883	0.198	0.221	0.517	0.224	0.195	0.218	0.224	0.169	0.951	0.221	0.215	0.087	0.172	0.954	0.853
MotorR	0.640	0.030	0.042	0.073	0.054	0.134	0.048	0.067	0.054	0.018	0.048	0.048	0.030	0.042	0.115	0.481
MoutB	0.930	0.723	0.877	0.122	0.232	0.693	0.653	0.833	0.434	0.276	0.921	0.350	0.622	0.473	0.263	0.982
Skiing	0.432	0.086	0.074	0.037	0.049	0.037	0.098	0.111	0.012	0.111	0.049	0.049	0.012	0.049	0.061	0.432
Trellis	0.874	0.551	0.253	0.233	0.163	0.692	0.145	0.688	0.203	0.479	0.209	0.479	0.261	0.739	0.411	0.643
Walking	0.719	0.534	0.885	0.371	0.368	0.446	0.640	0.995	0.284	0.216	0.393	0.410	0.851	0.830	0.293	0.405
Average	0.745	0.425	0.311	0.332	0.319	0.431	0.317	0.451	0.305	0.418	0.387	0.299	0.333	0.378	0.362	0.635

TABLE III
AVERAGE PRECISION SCORES

	SDT	L_1APG	IVT	Frag	KCF	Struck	MTT	ASLA	KMS	CXT	CSK	DFT	LOT	SCM	TLD	FCNT
Bird2	0.990	0.131	0.484	0.313	0.343	0.545	0.090	0.878	0.393	0.282	0.515	0.717	0.080	0.808	0.313	0.949
BlurF	0.988	0.355	0.109	0.630	0.123	0.612	0.316	0.091	0.754	1	0.995	0.281	0.389	0.137	0.998	0.995
Boy	1	0.604	0.332	0.574	0.382	1	0.445	0.440	0.990	1	0.843	0.485	0.666	0.440	1	1
Couple	1	0.585	0.085	0.907	0.257	0.757	0.642	0.107	0.107	0.578	0.085	0.085	0.635	0.085	0.221	1
Cross	0.993	1	0.983	0.400	1	1	0.250	1	1	0.575	1	0.683	0.635	1	0.583	1
David	0.889	0.796	1	0.110	0.787	0.322	0.343	0.522	1	0.498	0.354	0.284	0.422		0.955	0.921
David3	0.976	0.345	0.754	0.789		0.337	0.107	0.674	0.976	0.158	0.658	0.746	0.988	0.674	0.115	0.936
Deer	0.972	0.816	0.028	0.154	0.816	1	0.760	0.042	0.535	0.831		0.309	0.183	0.084	0.309	
Dog	1	0.881	0.165	0.858	0.992	0.976	1	0.968	0.622	1		0.724	0.897	1	0.622	0.984
DragB	0.921	0.256	0.327	0.477	0.070	0.194	0.168	0.283	0.486	0.566	0.212	0.123	0.681	0.168	0.088	0.867
Occ1	0.618	0.746	0.649	0.980	0.681	0.566	0.318	0.657	0.543	0.389	0.947	0.622	0.253	0.947	0.109	0.626
Girl	1	0.596	0.452	0.652	0.864	0.974	1	1	0.536	0.936	0.552	0.296	0.640	0.356	0.90	0.914
Gym	0.930	0.020	0.509	0.942	0.794	0.651	0.255	0.573	0.967	0.749	0.520	0.233	0.958	0.645	0.799	0.959
Human7	0.896	0.948	0.288	0.484	0.472	1	1	0.240	0.572	1	0.656	0.172	0.460	0.284	1	0.808
Jogging	0.978	0.228	0.224	0.661	0.234	0.231	0.228	0.231	0.224	0.960	0.228	0.215	0.599	0.228	0.973	0.973
MotorR	0.829	0.030	0.036	0.073	0.048	0.365	0.048	0.042	0.054	0.024	0.042	0.042	0.048	0.042	0.103	0.865
MoutB	0.987	0.864	1	0.140	0.478	0.964	1	0.907	0.666	0.280	1	0.350	0.693	0.982	0.280	1
Skiing	1	0.135	0.111	0.030	0.074	0.037	0.123	0.135	0.111	0.209	0.098	0.074	0.024	0.074	0.061	1
Trellis	0.974	0.783	0.253	0.374	0.353	0.789	0.230	0.718	0.353	0.790	0.810	0.506	0.309	0.906	0.479	0.985
Walking	1	1	1	0.987	1	1	1	1	0.995	0.235	1	1	1		0.915	1
Average	0.947	0.556	0.439	0.527	0.538	0.666	0.466	0.549	0.570	0.628	0.633	0.401	0.521	0.514	0.541	0.939

[7] Y. Su, Q. Zhao, L. Zhao, and D. Gu, "Abrupt motion tracking using a visual saliency embedded particle filter," *Pattern Recognition*, vol. 47, no. 5, pp. 1826–1834, 2014.

[8] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3119–3127, 2015.

[9] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*, pp. 1822–1829, IEEE, 2012.

[10] N. Wang, S. Li, A. Gupta, and D.-Y. Yeung, "Transferring rich feature hierarchies for robust visual tracking," *arXiv preprint arXiv:1501.04587*, 2015.

[11] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," in *Advances in neural information processing systems*, pp. 1196–1204, 2013.

[12] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Stct: Sequentially training convolutional networks for visual tracking," CVPR, 2016.

[13] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 2106–2113, IEEE, 2009.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255, IEEE, 2009.

[15] S. Beucher and F. Meyer, "The morphological approach to segmentation: the watershed transformation," *OPTICAL ENGINEERING-NEW YORK-MARCEL DEKKER INCORPORATED-*, vol. 34, pp. 433–433, 1992.

[16] H. Li, Y. Li, and F. Porikli, "Deeptrack: Learning discriminative feature representations online for robust visual tracking," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1834–1848, 2016.

[17] 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.

[18] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125–141, 2008.

[19] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1, pp. 798–805, IEEE, 2006.

[20] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[21] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2042–2049, IEEE, 2012.

[22] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[23] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1177–1184, IEEE, 2011.

[24] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *European conference on computer vision*, pp. 702–715, Springer, 2012.

[25] L. Sevilla-Lara and E. Learned-Miller, "Distribution fields for tracking," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1910–1917, IEEE, 2012.

[26] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan, "Locally orderless tracking," *International Journal of Computer Vision*, vol. 111, no. 2, pp. 213–228, 2015.

[27] 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.