Output Constraint Transfer for Kernelized Correlation Filter in Tracking

Baochang Zhang, Zhigang Li, Xianbin Cao, *Senior Member, IEEE*, Qixiang Ye, *Senior Member, IEEE*, Chen Chen, Linlin Shen, Alessandro Perina, and Rongrong Ji, *Senior Member, IEEE*

***Abstract*—The kernelized correlation filter (KCF) is one of the state-of-the-art object trackers. However, it does not reasonably model the distribution of correlation response during tracking process, which might cause the drifting problem, especially when targets undergo significant appearance changes due to occlusion, camera shaking, and/or deformation. In this paper, we propose an output constraint transfer (OCT) method that by modeling the distribution of correlation response in a Bayesian optimization framework is able to mitigate the drifting problem. OCT builds upon the reasonable assumption that the correlation response to the target image follows a Gaussian distribution, which we exploit to select training samples and reduce model uncertainty. OCT is rooted in a new theory which transfers data distribution to a constraint of the optimized variable, leading to an efficient framework to calculate correlation filters. Extensive experiments on a commonly used tracking benchmark show that the proposed method significantly improves KCF, and achieves better performance than other state-of-the-art trackers. To encourage further developments, the source code is made available.**

***Index Terms*—Correlation filter, online learning, tracking.**

I. Introduction

VISUAL object tracking is a fundamental problem in computer vision, which contributes to various applications, including robotics, video surveillance, and intelligent vehicles [1]–[3]. While many works consider object tracking in simple scenes as a solved problem, online object tracking in uncontrolled real-world scenarios remains open, with key challenges like illumination change, occlusion, motion blur, and texture variation [1], [2], [4]–[6]. To this end, the conventional data association and temporal filters [7] that rely on motion modeling typically fail due to the dynamic and changing object/background appearances.

Most recently, kernalized correlation filters (KCFs), which aims to construct discriminative appearance model for tracking from a learning-based perspective, has shown to be promising to handle the appearance variations [8]–[10]. KCF incorporates translated and scaled patches to make a kernelized model distinguishing between the target and surrounding environment [8]. It also adopts fast Fourier transform (FFT) and inverse FFT (IFFT) to improve the computational efficiency. Experimental comparisons show that KCF-based tracking is competitive among the state-of-the-art trackers in terms of speed and accuracy [8]. Although much success has been demonstrated, irregular correlation responses and target drifting have been observed. These are particularly common when updating target appearance in a long tracking stream with occlusion, camera shake, and great appearance changes [11]. From the perspective of learning, sample noises are introduced to the filter, which degrades the model learning and drift the tracker away [11], [12]. To alleviate such risk of drifting, we advocate that the tracker should model the correlation response (output) to reduce noisy samples to achieve stable tracking. We propose preventing the drifting through controlling maximum response to follow the Gaussian distribution, which not only reduce noise samples, but also gain the robustness to variations.

As another intuition, it is well known that data lies on specific distributions, i.e., faces are considered to be from subspace [13], [14]. As long as the optimal solution resides on the data domain, the constraints derived from the data structure can bring robustness to the variations [15], [16]. To this end, any tracking framework taking advantage of the implicit data structure can improve tracking. Imposing a data structure (distribution) as a constraint is actually a new and flexible way to solve the optimization problems [15], [16], which has been promising in various learning algorithms. The main crux is how to efficiently embed structure constraint in the optimization method. In this paper, we demonstrate the existence of a highly practical solution to include Gaussian constraints in KCF.

Fig. 1 shows the proposed output constraint transfer (OCT) method,[1] which mainly innovates at learning robust

Manuscript received June 19, 2016; accepted November 4, 2016. This work was supported in part by the Natural Science Foundation of China (NSFC) under Contract 61672079 and Contract 61672357, and in part by the Science and Technology Innovation Commission of Shenzhen under Grant JCYJ20160422144110140. The work of B. Zhang was supported in part by the Program for New Century Excellent Talents University within the Ministry of Education, China, in part by the Beijing Municipal Science and Technology Commission under Grant Z161100001616005, and in part by the NSFC under Grant 61272052. This paper was recommended by Associate Editor K. Huang. *(Corresponding author: Linlin Shen.)*

B. Zhang, Z. Li, and X. Cao are with Beihang University, Beijing 100080, China.

Q. Ye is with the University of Chinese Academy of Sciences, Beijing 100049, China.

C. Chen is with the Center for Research in Computer Vision, University of Central Florida, Orlando, FL 32816 USA.

L. Shen is with Computer Vision Institute, School of Computer Science & Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: llshen@szu.edu.cn).

A. Perina is with Microsoft Corporation, Redmond, WA, USA.

R. Ji is with Xiamen University, Xiamen 361005, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2016.2629509

[1]The source code will be publicly available on mpl.buaa.edu.cn.

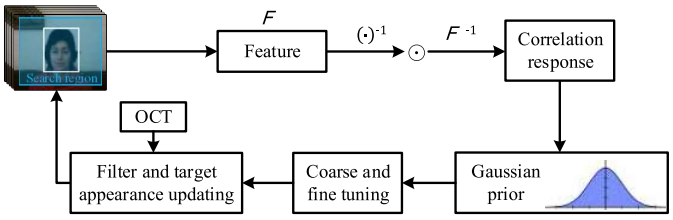


Fig. 1. Scheme of OCT-KCF for object tracking.

kernerlized correlation filters for object tracking. Two key innovations are introduced: 1) the Gaussian prior constraint is exploited to model the filter response and reduce noisy samples and 2) a new theory termed OCT is proposed to transfer data distribution to be a constraint of the optimized variable. By the constraint, our correlation filters are particularly prone to find the *data* (response output) following a certain distribution[2] and gain the robustness to variations. By the proposed OCT theory, instead of directly controlling the response output in a brute-force way, we alternatively transfer the distribution information from the data to be a constraint of the optimized variable.

The Gaussian assumption on correlation output is supported from three aspects.

1) It is first supported by [8], and shows that a single threshold on the correlation response (output) is used, which inspire us that the correlation response (output) actually follows a simple distribution, i.e., Gaussian.
2) As evident on tracking, a simple distribution is necessary and significant to achieve high efficiency. The complex distribution, i.e., Gaussian mixture model, can not result in an efficient model as ours. As for the complex distribution, it would be further considered in our future work and could be possible to have a more general theory.
3) As a final evidence, our extensive experiments on the commonly used benchmark [17] confirm that the Gaussian distribution is highly effective.

The rest of this paper is organized as follows. Section II introduces the related work. We present the constraint problem on correlation filters in Section III, and detail how Gaussian constraints can be efficiently embedded in an online optimization framework in Section IV. Finally, extensive experiments are discussed in Section V, while we draw our conclusions in Section VI.

II. Related Work

Visual tracking has been extensively studied in [12] and [17]. In this section, we discuss the methods closely related to this paper, i.e., the appearance models, and more particularly, the correlation filter-based models.

An appearance model consists of learning a classifier online, to predict the presence or absence of the target in an image patch. This classifier is then tested on many candidate patches to find the most likely location [8], [18]–[20]. Popular learning schemes include kernel learning [21], [22], latent structure [2], multiple instance learning [23], boosting [24], [25], metric learning [26], and structured learning [27]. However, the online tracking algorithms often encounter the drifting problems. As for the self-taught learning, these misaligned samples are likely to be added and degrade the appearance models. To avoid drifting, the most famous tracking-by-detection (TLD) method employs positive–negative learning to choose "safe" samples [16]. The compressed tracking method employs nonadaptive random projections that preserve the structure of the image feature space of objects [28]. It compresses samples of foreground targets and the background using the same sparse measurement matrix to guarantee the stability of tracking [28].

The initial motivation for our research was the recent success of correlation filters in tracking [29]. Correlation filters have been proved to be competitive with far more complicated approaches, but using only a fraction of the computational power, at hundreds of frames per second. They take advantage of the fact that the convolution of two patches is equivalent to an element-wise product in the FFT domain. Thus, by formulating their objective in the FFT domain, they can specify the desired output of a linear classifier for several translations, or image shifts [8]. Taking the advantages of correlation filters, Bolme *et al.* [29] proposed to learn a minimum output sum of squared error filter for visual tracking on gray-scale images. Heriques *et al.* [30] proposed using correlation filters in a kernel space based on exploiting the circulant structure of tracking-by-detection with kernels (CSK), which achieves the highest speed in the commonly used benchmark [17]. CSK is introduced based on kernel ridge regression, which has been one of the hottest topics in correlation filter learning. Using a dense sampling strategy, the circulant structure exploits data redundancy to simplify the training and testing process. By using histogram of gradient (HOG) features, KCF is further proposed to improve the performance of CSK. Danelljan *et al.* [10] exploited the color attributes of a target object and learn an adaptive correlation filter by mapping multichannel features into a Gaussian kernel space. Recently, Ma *et al.* [11] introduced a redetecting process to further improve the performance of KCF [8]. Zhang *et al.* [9] incorporated context information into filter learning and model the scale change based on consecutive correlation responses. The DSST tracker [10] learns adaptive multiscale correlation filters using HOG features to handle the scale variations. Recent works involve using learned convolutional filters for visual object tracking [32], [33]. Although much success has been demonstrated, the existing works do not principally incorporate the distribution information into the procedure of solving the optimized variable.

III. Output Constraint Transfer in KCF

In this section, we first introduce KCF, and then describe how the response output is constrained by a Gaussian distribution.

A. Kernelized Correlation Filter

KCF starts from the kernel ridge regression method [8], which is formulated as

$$\min_{w,\xi} \sum_i \xi_i^2$$
$$\text{subject to } y_i - \boldsymbol{w}^T\boldsymbol{\phi}(x_i) = \xi_i \quad \forall i; \|w\| \le B \qquad \text{(P1)}$$

where x_i is the $M \times N$-sized image. $\phi(.)$ is a nonlinear transformation. $\phi(x_i)$ (later ϕ_i) and y_i are the input and output,

[2]Assumed Gaussian for its simplicity, although other complex distributions may be more reasonable.

respectively. ξ_i is a slack variable. B is a small constant. Based on the Lagrangian method, the objective corresponding to (P1) is rewritten as

$$\mathcal{L}_p = \sum_{i=1}^{M\times N} \xi_i^2 + \sum_{i=1}^{M\times N} \beta_i\big[y_i - \boldsymbol{w}^T\phi_i - \xi_i\big] + \lambda\Big(\|\boldsymbol{w}\|^2 - B^2\Big) \quad (1)$$

where λ is a regularization parameter ($\lambda \geq 0$). From (1), we have

$$\begin{aligned} \boldsymbol{\alpha} &= (K + \lambda \boldsymbol{I})^{-1}\boldsymbol{y} \\ \boldsymbol{w} &= \sum_i \alpha_i \phi_i. \end{aligned} \quad (2)$$

The matrix K with elements $K_{ij} = k(P^i\boldsymbol{x}, P^j\boldsymbol{x})$ is circulant given a kernel such as the gaussian kernel k [30]. Taking advantage of the circulant matrice, the FFT of $\boldsymbol{\alpha}$ denoted by $\mathcal{F}(\boldsymbol{\alpha})$ is calculated by

$$\mathcal{F}(\boldsymbol{\alpha}) = \frac{\mathcal{F}(y)}{\mathcal{F}(k^{xx}) + \lambda} \quad (3)$$

where $\mathcal{F}$ denotes the discrete Fourier operator, and k^{xx} is the first row of the circulant matrix K. In tracking, all candidate patches that are cyclic shifts of test patch z are evaluated by

$$\mathcal{F}\big(\hat{\boldsymbol{y}}\big) = \mathcal{F}\Big(k^{z\hat{x}}\Big) \odot \mathcal{F}(\boldsymbol{\alpha}) \quad (4)$$

where $\odot$ is the element-wise product and $\hat{x}$ is a learned target appearance image calculated by (6a) [11], $\mathcal{F}(\hat{\boldsymbol{y}})$ is the output response for all the testing patches in frequency domain. We then have

$$\hat{\boldsymbol{y}} = \max\Big(\mathcal{F}^{-1}\big(\mathcal{F}\big(\hat{\boldsymbol{y}}\big)\big)\Big) \quad (5)$$

where $\mathcal{F}^{-1}$ is the IFFT. The target position is the one with the maximal value among $\hat{\boldsymbol{y}}$ calculated by (5). The target appearance and correlation filter are then updated with a learning rate η as

$$\begin{cases} \hat{x}^t = (1-\eta)\hat{x}^{t-1} + \eta x^t & (6a) \\ \mathcal{F}\big(\boldsymbol{\alpha^t}\big) = (1-\eta)\mathcal{F}\Big(\boldsymbol{\alpha^{t-1}}\Big) + \eta\mathcal{F}(\boldsymbol{\alpha}). & (6b) \end{cases}$$

Kernel ridge regression relies on computing kernel correlation (k^{xx} and $k^{z\hat{x}}$). Considering that kernel correlation consists of computing the kernel for all relative shifts of two input vectors. This represents the last computational bottleneck, as an evaluation of n kernels for signals of size n will have quadratic complexity. However, using the cyclic shift model will allow us to efficiently exploit the redundancies in this expensive computation. The computational complexity for the full kernel correlation is only $O(n \cdot \log(n))$.

B. Problem Formulation

In tracking applications, the correlation response of the target object is assumed to follow a Gaussian distribution, which is not discussed in the existing works. In this section, we solve the (P1) by exploiting the Gaussian assumption in an optimization process. Now, the original (P1) can be rewritten in the tth frame as

$$\begin{aligned} &\min_{w,\xi} \ \sum_i \xi_i^2 \\ &\text{s.t.:} \ \ y_i - \boldsymbol{w}^{T,t}\boldsymbol{\phi}_i = \xi_i \\ &\qquad \hat{\boldsymbol{y}}^t \sim \mathcal{N}\Big(\mu^t, \sigma^{2,t}\Big) \\ &\qquad \|w\| \leq B \end{aligned} \quad (P2)$$

where y_i is the Gaussian function label for the ith sample $\boldsymbol{\phi}_i$ in the tth frame [8]. μ and σ^2 are the mean and variance of the Gaussian model $\mathcal{N}$, respectively. $\hat{y}^t$ is a new variable to represent the response of the target image based on $\boldsymbol{w}^{T,t}$ and (5). As mentioned above, Gaussian prior is defined as

$$\hat{\boldsymbol{y}}^t \sim \mathcal{N}\Big(\mu^t, \sigma^{2,t}\Big). \quad (7)$$

In (P2), only $\hat{\boldsymbol{y}}^t \sim \mathcal{N}(\mu^t, \sigma^{2,t})$ is unsolved. As shown in the maximum likelihood method in the probability theory [34], Gaussian prior can be alternatively solved through minimizing $[(\hat{\boldsymbol{y}}^t - \mu^t)^2/2\sigma^t] + [ln(2\pi \cdot \sigma^t)/2]$. As only $[(\hat{\boldsymbol{y}}^t - \mu^t)^2/2\sigma^t]$ is related to the optimized variable, μ^t and $\sigma^{2,t}$ are solved iteratively. And for simplicity, $\sigma^{2,t}$ can be considered as a constant in the tth time. Thus, the denominator is ignored, we alternatively minimize $(\hat{\boldsymbol{y}}^t - u^t)^2$. The smaller the value is, the more possible the correlation response in current frame satisfies the Gaussian prior.[3]

IV. OCT-Based KCF

In the previous section, we describe a new framework to calculate KCF. Nevertheless, it remains complex to solve the tracking problem, due to the new variable $\hat{\boldsymbol{y}}^t$. Here, we introduce the proposed OCT theory to further reformulate (P2) into an extremely simple problem.

A. Theory of Transferring Constraints: OCT

The OCT theory aims to simplify the optimization process in particular for $(\hat{\boldsymbol{y}}^t - u^t)^2$. As a result of the theory $\hat{\boldsymbol{y}}^t$ is replaced by a new constraint only added on the variable $\boldsymbol{w}$. This is remarkable, the Gaussian constraint is deployed without extra complexity, i.e., $\hat{\boldsymbol{y}}^t$ is not involved. Here, $\hat{\boldsymbol{y}}^t = \boldsymbol{w}^{T,t}x^t$ with x^t as the target object.

Theorem: Minimizing of $(\boldsymbol{w}^{T,t}x^t - \mu^t)^2$ is transferred to minimizing $\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|^2$, when the learned target appearances have no great changes in two consecutive frames.

Based on the theorem, the data distribution is transferred to a constraint only for the unsolved variable, by which (P2) is further relaxed, leading to an extremely efficient method to calculate correlation filters.

Proof: The mean of Gaussian is updated as

$$\mu^t = (1-\rho)\mu^{t-1} + \rho\boldsymbol{w}^{T,t}\hat{x}^t \quad (8)$$

where $\hat{x}^t$ is the learned target appearance in the tth frame, iteratively acquired by

$$\hat{x}^t = (1-\rho)\hat{x}^{t-1} + \rho x^t \quad (9)$$

where x^t is the target in the tth frame. Similar to (8), μ^{t-1} can be calculated as

$$\mu^{t-1} = (1-\rho)\mu^{t-2} + \rho\boldsymbol{w}^{T,t-1}\hat{x}^{t-1}. \quad (10)$$

By plugging (10) back into (8), we get

$$\mu^t = (1-\rho)\Big((1-\rho)\mu^{t-2} + \rho\boldsymbol{w}^{T,t-1}\hat{x}^{t-1}\Big) + \rho\boldsymbol{w}^{T,t}\hat{x}^t \quad (11)$$

[3] μ and σ^2 are calculated based on all previous frames in the tracking procedure.

which is rewritten as

$$\boldsymbol{w}^{T,t}x^t - \mu^t = -\rho(1-\rho)\boldsymbol{w}^{T,t-1}\hat{x}^{t-1} - (1-\rho)^2\mu^{t-2} - \rho\boldsymbol{w}^{T,t}\hat{x}^t + \boldsymbol{w}^{T,t}x^t. \quad (12)$$

Here, $\boldsymbol{w}^{T,t}x^t$ is approximated by $\boldsymbol{w}^{T,t}\hat{x}^t$, which does not change the tracking result. Thus, (12) is rewritten as

$$\boldsymbol{w}^{T,t}x^t - \mu^t = -\rho(1-\rho)\boldsymbol{w}^{T,t-1}\hat{x}^{t-1} - (1-\rho)^2\mu^{t-2} + (1-\rho)\boldsymbol{w}^{T,t}\hat{x}^t + \epsilon_1. \quad (13)$$

where ϵ_1 is a small constant. Plugging (9) back into the above equation, we have

$$\boldsymbol{w}^{T,t}x^t - \mu^t = -\rho(1-\rho)\boldsymbol{w}^{T,t-1}\hat{x}^{t-1} - (1-\rho)^2\mu^{t-2} + (1-\rho)^2\boldsymbol{w}^{T,t}\hat{x}^{t-1} + (1-\rho)\rho\boldsymbol{w}^{T,t}\hat{x}^t + \epsilon_1. \quad (14)$$

Based on the hypothesis that the learned target appearances $(\hat{x}^t,\ \hat{x}^{t-1})$ have no great changes in two consecutive frames, we have

$$\boldsymbol{w}^{T,t}x^t - \mu^t = \rho(1-\rho)\left(\boldsymbol{w}^{T,t} - \boldsymbol{w}^{T,t-1}\right)\hat{x}^{t-1} + (1-\rho)^2\left(\boldsymbol{w}^{T,t}\hat{x}^{t-1} - \mu^{t-2}\right) + \epsilon_2 \quad (15)$$

where ϵ_2 is a small constant

$$\begin{aligned}\left\|\boldsymbol{w}^{T,t}x^t - \mu^t\right\| &\le \left\|(1-\rho)^2\left(\boldsymbol{w}^{T,t}\hat{x}^{t-1} - \mu^{t-2}\right)\right\| \\ &\quad + \left\|\rho(1-\rho)\left(\boldsymbol{w}^{T,t} - \boldsymbol{w}^{T,t-1}\right)\hat{x}^{t-1}\right\| + \|\epsilon_2\| \\ &\le \rho(1-\rho)\left\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\right\| \cdot \left\|\hat{x}^{t-1}\right\| + C \end{aligned} \quad (16)$$

where C is a constant. From the above inequality, the minimization of $(\boldsymbol{w}^{T,t}x^t - \mu^t)^2$ is converted to minimizing $\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|^2$, theorem is proved. ■

B. OCT Solution to (P2)

Bayesian optimization is a powerful framework which has been successfully applied to solve various problems, i.e., parameter tuning. The Bayesian optimization can also be used to solve (P1). Two of the KKT conditions from (1) are

$$2\xi_i = \beta_i^t \quad (17)$$

$$2\lambda\boldsymbol{w} = \sum_i \beta_i^t\boldsymbol{\phi}_i. \quad (18)$$

According to our theory, we add the minimizing of $\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|^2$ to replace the Gaussian constraint in (P2). However, it is still a little complicated for our problem. Based on (18), we simply use $\|\boldsymbol{\beta}^t - \boldsymbol{\beta}^{t-1}\|^2$, and obtain a dual form for (P2) via the Lagrangian method, which is formulated in a Bayesian framework as

$$\mathcal{L}_P\left(\alpha|\left(\mu^t, \sigma^{2,t}\right)\right) = -\frac{1}{4}\sum_i \beta_i^{2,t} - \frac{1}{4\lambda}\sum_{i,j}\beta_i^t\beta_j^t K_{ij} + \sum_i \beta_i^t y_i - s\sum_i\left(\beta_i^t - \beta_i^{t-1}\right)^2 - \lambda B^2. \quad (19)$$

Redefining $\alpha_i^t = \beta_i^t/2\lambda$, we come up with the following optimization problem:

$$\max_{\boldsymbol{\alpha},\lambda} -\lambda^2\sum_i \alpha_i^{2,t} + 2\lambda\sum_i \alpha_i^t y_i - \lambda\sum_{i,j}\alpha_i^t\alpha_j^t K_{ij} - 4\lambda^2 s\sum_i\left(\alpha_i^t - \alpha_i^{t-1}\right)^2. \quad (20)$$

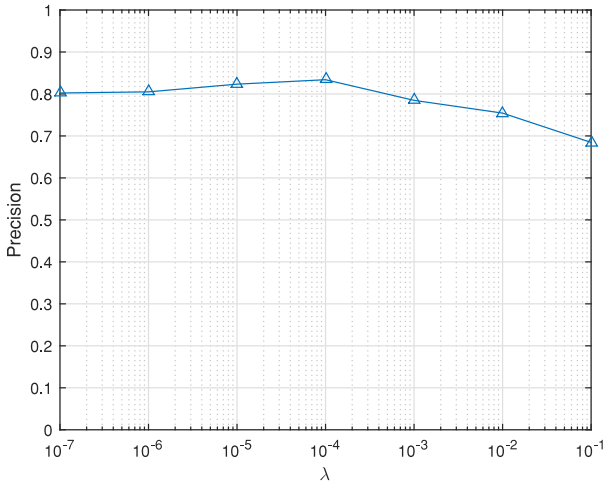


Fig. 2. Evaluation of λ based on precision.

and we have

$$(\lambda I + 4\lambda s I + K)\boldsymbol{\alpha}^t = \boldsymbol{y} + 4\lambda s\boldsymbol{\alpha}^{t-1}. \quad (21)$$

Then the FFT of $\boldsymbol{\alpha}$ is calculated as

$$F\left(\boldsymbol{\alpha}^t|\left(\mu^t, \sigma^{2,t}\right)\right) = \frac{F(\boldsymbol{y}) + 4\lambda s F\left(\boldsymbol{\alpha}^{t-1}\right)}{F(\boldsymbol{k}) + \lambda + 4\lambda s} \quad (22)$$

which is rewritten as

$$F\left(\boldsymbol{\alpha}^t|\left(\mu^t, \sigma^{2,t}\right)\right) = \frac{F(\boldsymbol{k}) + \lambda}{F(\boldsymbol{k}) + \lambda + 4\lambda s} \odot \frac{F(\boldsymbol{y})}{F(\boldsymbol{k}) + \lambda} + \frac{4\lambda s}{F(\boldsymbol{k}) + \lambda + 4\lambda s} \odot F\left(\boldsymbol{\alpha}^{t-1}\right). \quad (23)$$

Defining $\boldsymbol{\eta}$ as

$$\boldsymbol{\eta} = \frac{F(\boldsymbol{k}) + \lambda}{F(\boldsymbol{k}) + \lambda + 4\lambda s} \quad (24)$$

we have

$$F\left(\boldsymbol{\alpha}^t|\left(\mu^t, \sigma^{2,t}\right)\right) = \boldsymbol{\eta} \odot F(\boldsymbol{\alpha}) + (1-\boldsymbol{\eta}) \odot F\left(\boldsymbol{\alpha}^{t-1}\right) \quad (25)$$

where μ^t and $\sigma^{2,t}$ are used to select the samples as shown in (26). $\boldsymbol{\eta}$ is a matrix with the same size as $F(\boldsymbol{\alpha})$. According to (25), the update of the filter relies on the evolving $\boldsymbol{\eta}$, which is different with KCF (6b) that relies on a constant. More details about $\boldsymbol{\eta}$ can also refer to our source code. To be concluded from the results mentioned above, the iterative formula of correlation filter (25) is obtained from the theoretical derivation.

C. Coarse and Fine Tuning Based on Gaussian Prior

Due to the appearance variations of the target, the tracker might gradually drift and finally fail. Different from existing works using threshold to detect the failure case, we argue that the property of Gaussian prior can well prevent drifting. In particular, we adopt the Gaussian prior to select samples when their response output belong to a Gaussian distribution, that is: the sample is chosen, only when its response output belongs to a Gaussian distribution

$$\left|\frac{\hat{\boldsymbol{y}}^t - \mu^t}{\sigma^t}\right| < \mathcal{T}_g \quad (26)$$

Algorithm 1 OCT Algorithm for Object Tracking

Initial target bounding box $\boldsymbol{b}_0 = [x_0, y_0, w, h]$,
if the frame $n \leq 20$
repeat
  Crop out the search windows according to $\boldsymbol{b}_{n-1}$, and extract the HOG features.
  Compute the maximum correlation response $\hat{\boldsymbol{y}}$ using Eqn.4 and Eqn.5 and record the maximal correlation response as y_n
  The position is obtained according to the maximal correlation response
  Updating target appearance and correlation filter using Eqn.6a and Eqn.25.
until $n == 20$
end
Compute the mean μ and variance σ^2 using all previous frames.
if $n > 20$
repeat
  Crop out the search window and extract the HOG features.
  Compute the maximal correlation response $\hat{\boldsymbol{y}}$ using Eqn.4 and Eqn.5.
  if $\left|\frac{\hat{\boldsymbol{y}}-\mu}{\sigma}\right| > \mathcal{T}_g$
  Crop out the coarse regions $Z = \{z_1, z_2, ..., z_{n_r * n_t}\}$ according to the coordinates calculated by Eqn.27 and Eqn.28 around the center of $\boldsymbol{b}_{n-1}$
  Coarse searching step:
  Detect the patch $\hat{z}$ in which the target appears with maximal probability using Eqn.30 and Eqn.31
  Fine searching step:
  Locate the object precisely using Eqn.(4) and Update target appearance and correlation filter using Eqn.6a and Eqn.25
  end
  Updating μ and σ^2
until End of the video sequence.
end

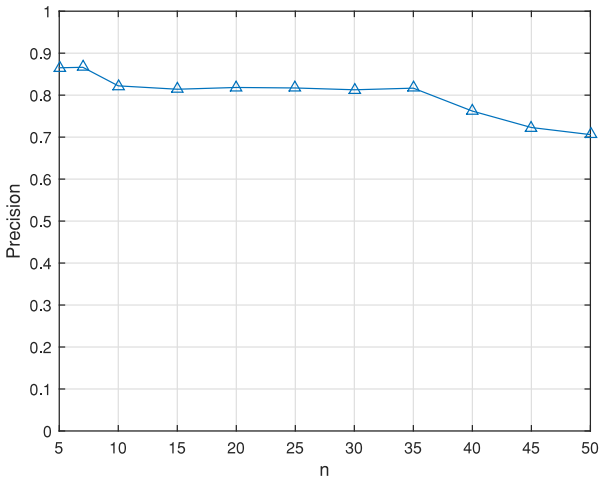


Fig. 3. Evaluation of n based on precision.

where $\mathcal{T}_g = 1.6$ is empirically set to a constant. Here, we introduce a fine-tuning process to precisely localize the target for sample selection in a local region, instead of searching over the whole image extensively. The tracker activates the fine-tune process when the maximal correlation response is out of the Gaussian distribution (drifting). We first detect the coarse region, where the target is most likely to appear near the location in previous frame. We then search a coarse region from n_t directions around the center of the latest location (x_0, y_0). The coordinates of a center location for coarse regions

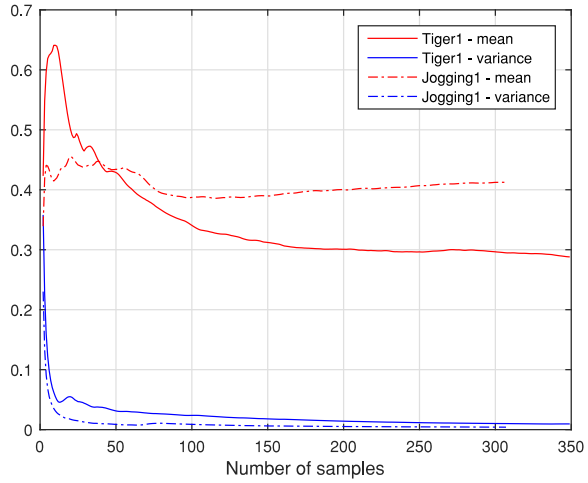


Fig. 4. Illustration of Gaussian mean and variance on the *Tiger1* and *Jogging1* sequences.

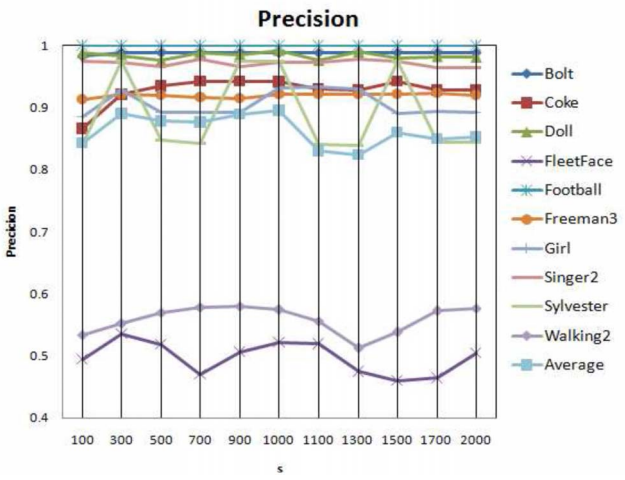


Fig. 5. Evaluation of s based on precision.

are calculated by

$$p_x = \begin{cases} x_0 + i_r * r_s * \cos(i_t * t_s) & \text{for } i_t \bmod 2 = 0 \\ x_0 + i_r * r_s * \cos(i_t * t_s + \phi) & \text{for } i_t \bmod 2 = 1 \end{cases} \quad (27)$$

$$p_y = \begin{cases} y_0 + i_r * r_s * \sin(i_t * t_s) & \text{for } i_t \bmod 2 = 0 \\ y_0 + i_r * r_s * \sin(i_t * t_s + \phi) & \text{for } i_t \bmod 2 = 1. \end{cases} \quad (28)$$

where $r_s = (\text{radius}/n_r)$, $i_r \in \{1, \ldots, n_r\}$, $t_s = (2\pi/n_t)$, $i_t \in \{1, \ldots, n_t\}$, $\phi = (t_s/2)$. Finally, $n_r * n_t$ patches centered around the target are cropped as

$$Z = \{z_1, z_2, , , z_{n_r * n_t}\}. \quad (29)$$

In the coarse process, the maximal correlation response of each patch is obtained by

$$r_i = \max\left(F^{-1}\left(\mathcal{F}(z_i) = \mathcal{F}\left(k^{z_i \hat{x}}\right) \odot \mathcal{F}(\boldsymbol{\alpha})\right)\right). \quad (30)$$

Then the patch in which the target appears with maximum probability is calculated as

$$\hat{z} = \arg\max_i \left(z_1, \ldots, z_i, \ldots z_{n_r * n_t}\right). \quad (31)$$

The fine-tuning step is executed to find the location $(\hat{z})$ of the object precisely as shown in (4). The initialized process

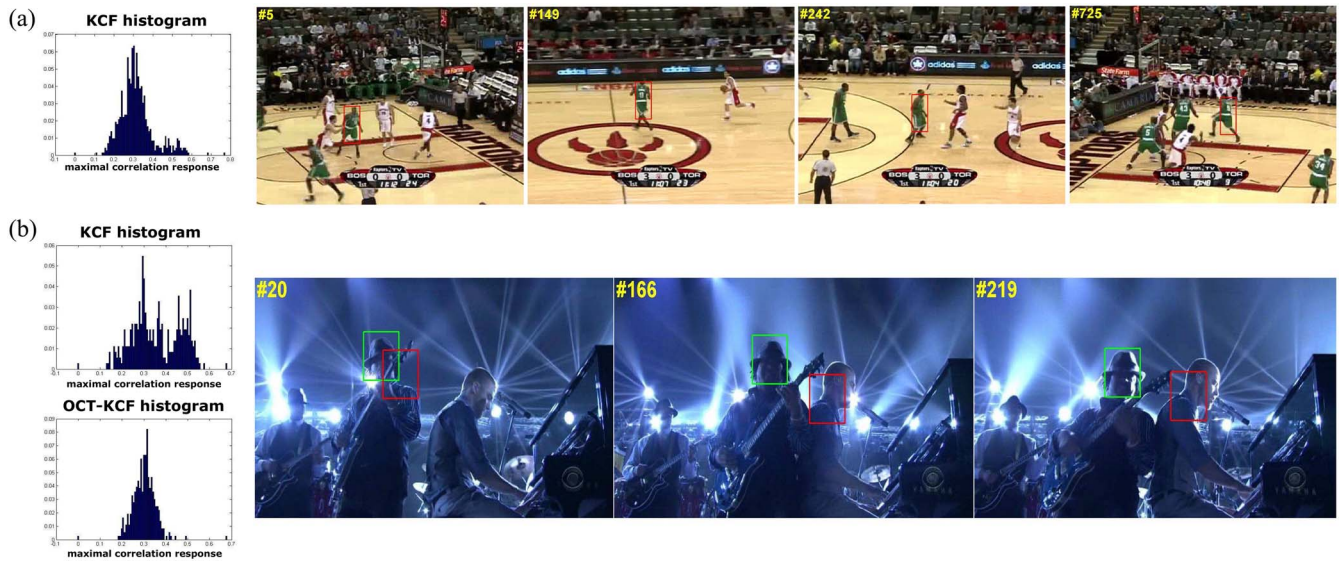


Fig. 6. Illustration of KCF and OCT-KCF on *Basketball* and *Shaking* sequences. (a) Good performance is achieved when the response (output) of KCF is observed to follow a Gaussian distribution on the *Basketball* sequence. (b) OCT (green rectangular) is used to improve the performance of KCF (red rectangular) on the *Shaking* sequence, and correlation response in OCT-KCF follows a Gaussian distribution.

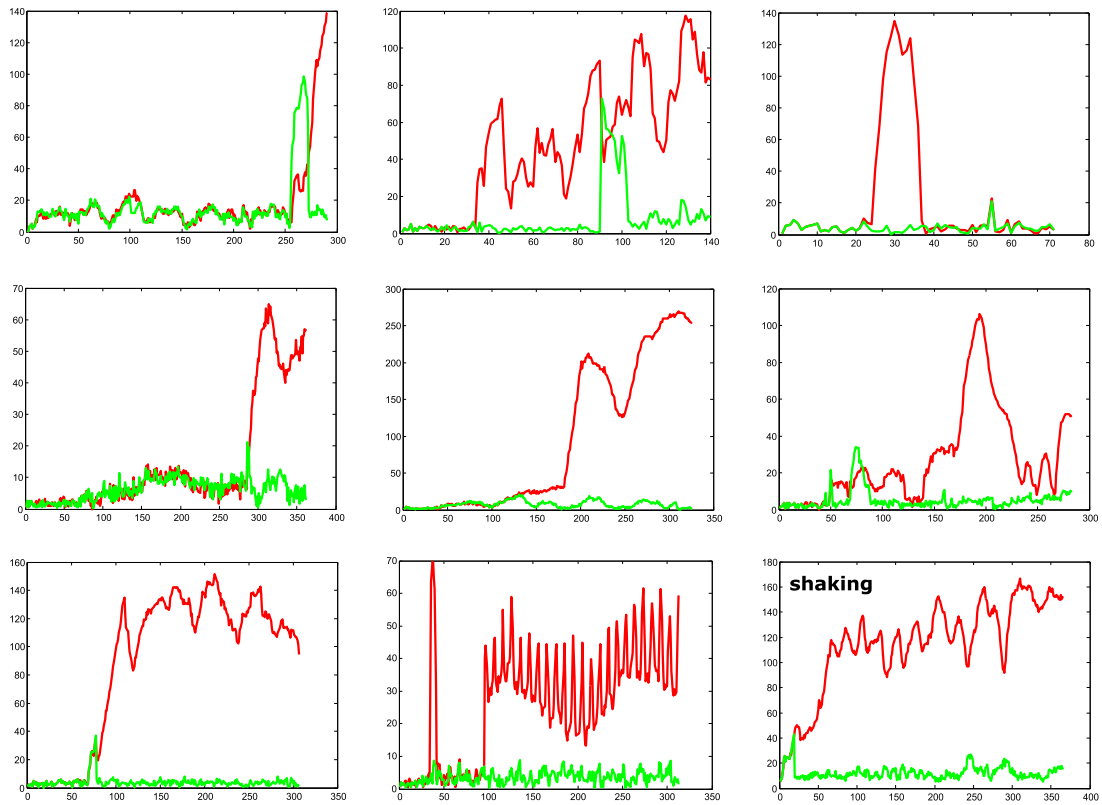


Fig. 7. Comparison between OCT-KCF (green line) and KCF (red line) based on CLE.

is empirically set during the first 20 frames. The fine-tuning strategy is easily implemented to update the localization of the tracked target. To sum up, Algorithm 1 recaps the complete method.

V. Experiments

In this section, we evaluate the performance of our tracker on 51 sequences of the commonly used tracking benchmark [17]. In this tracking benchmark [17], each sequence is manually tagged with 11 attributes which represent challenging aspects in visual tracking, including *illumination variations*, *scale variations*, *occlusions*, *deformations*, *motion blur*, *abrupt motion*, *in-plane rotation*, *out-of-plane rotation*, *out-of-view*, *background clutters*, and *low resolution*.

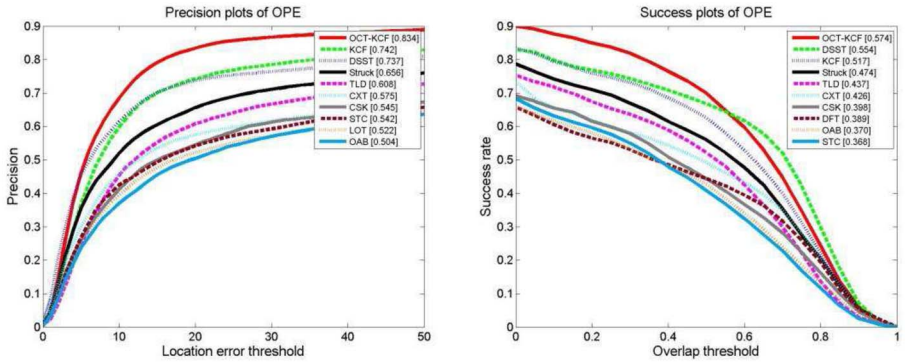


Fig. 8. Success and precision plots according to the online tracking benchmark [17].

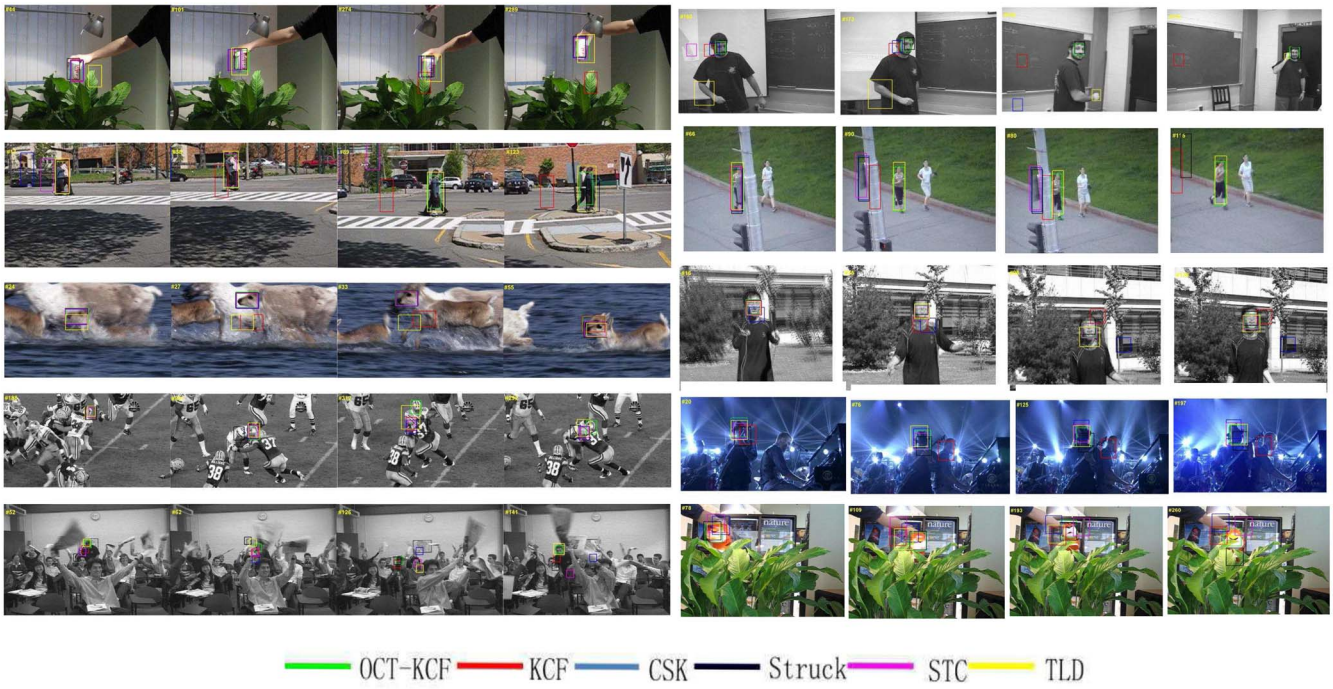


Fig. 9. Illustration of some key frames.

TABLE I
COMPARISONS WITH STATE-OF-THE-ART TRACKERS ON THE 51 BENCHMARK SEQUENCES

	OCT-KCF	KCF	DSST	TLD	STC	CSK	Struck
Ref.	Ours	[8]	[10]	[16]	[9]	[31]	[28]
Speed (FPS)	51	185	59	87	410	430	13
Precision	83.4	74.2	73.7	60.8	54.2	54.5	65.6
Success rate	57.4	51.7	55.4	43.7	36.8	39.8	47.4

A. Parameters Evaluation

We have tested the robustness of the proposed method in various parameter settings. For example, an experiment is done based on a subset of [17][4] as shown in Fig. 2, the precision is not changed much when λ is set from 10^{-7} to 10^{-3}. About the initialized number of samples for Gaussian model, we tested different values in Fig. 3, and the performance is very stable around 20 that is finally chosen in the following experiments. Moreover, we illustrate Gaussian mean and variance in the tracking process in Fig. 4, which appear to be stable if the target is well tracked for the *Tiger1* sequence, otherwise it seems randomly for the *Jogging1* sequence due to the wrong candidate tracked. s is a parameter used in (25), the experiment based on a subset of [17] is done as shown in Fig. 5. The performance of OCT is affected a litter by choosing different values of s. On most sequences (also average) the results on $s = 1000$ is better than others. So we choose $s = 1000$ in our experiment. To be consistent with [8], we set $\lambda = 10^{-4}$, $\rho = (1/t)$ with t as the frame number, and the searching size is 1.5. The Gaussian kernel function (standard variance $= 0.5$) and most parameters used in OCT-KCF are empirically chosen according to [8]. For other parameters, we empirically set $n_r = 5$, $n_t = 16$ on all sequences.

Fig. 6 shows that KCF achieves a good performance when the correlation response of the target image follows a Gaussian distribution, i.e., in the *Basketball* sequence. A failure, i.e., in the *Shaking* sequence, is observed when the output is sharply changed. Fig. 6 also shows that the proposed OCT method can force correlation response of KCF to follow a near-Gaussian distribution, and improves the tracking results of KCF on

[4]The datasets are shown in Fig. 5.

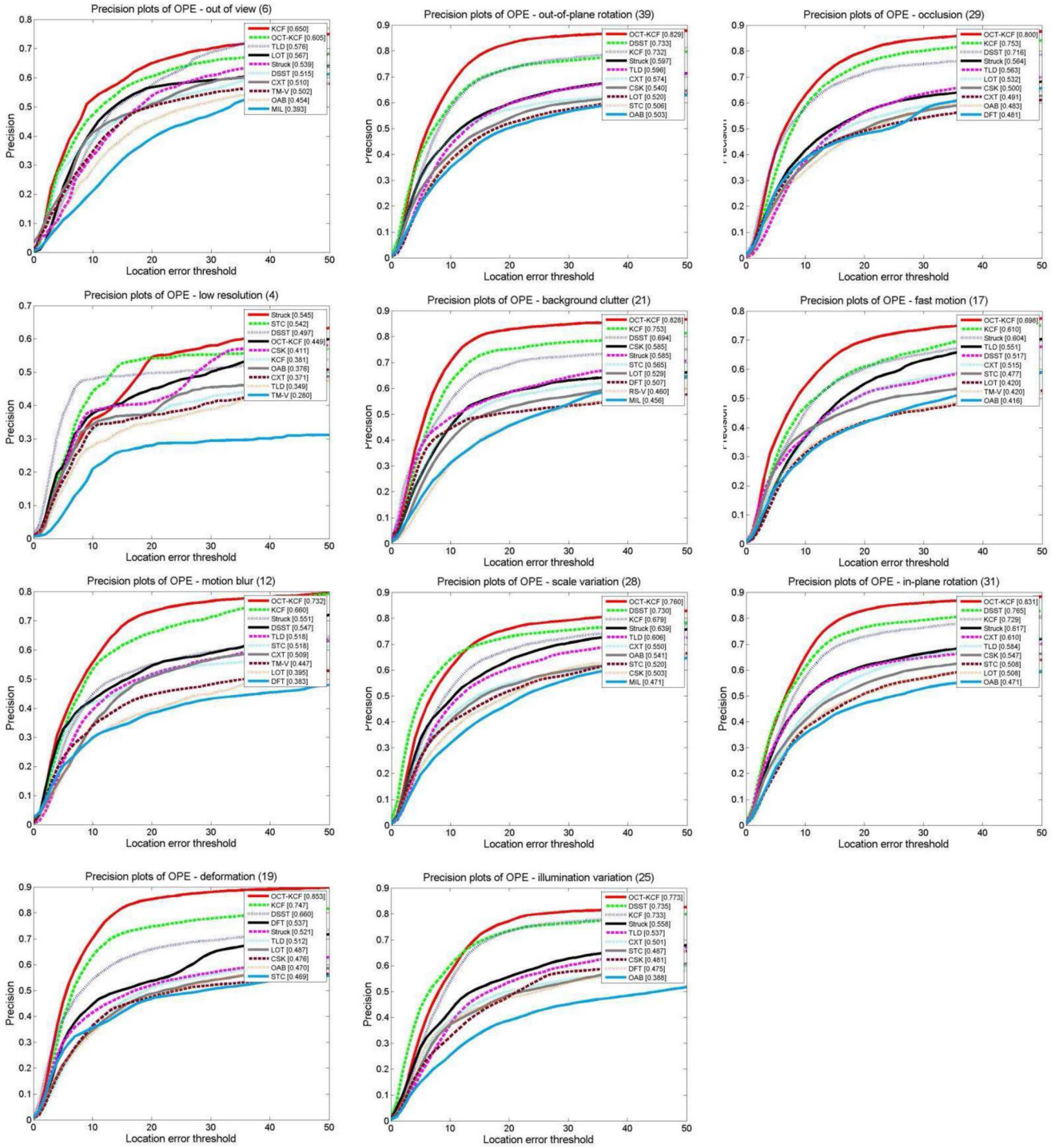


Fig. 10. Precision plots for the 11 attributes of the online tracking benchmark.

the *Shaking* sequence. We compare OCT-KCF with KCF in terms of the central location error (CLE) in Fig. 7. It can been seen that the proposed OCT-KCF gets stable performance in terms of CLE, which is clearly indicated by the smooth curves. In contrast, the curves of KCF have hitting turbulence. As illustrated in the coke sequence, both OCT-KCF and KCF lose the target at about the 275th frame, nevertheless, the OCT-KCF can relocate it at about the 275th frame while KCF fails to do that. The reason is that OCT can help KCF finding the candidate patch whose correlation response satisfies a Gaussian distribution and constraining the tracker from drifting. Similarly, the OCT-KCF tracker achieves much better performance in the sequences of *Couple*, *Deer*, *Football*, etc., than KCF. The CLE results support our previous analysis that OCT-KCF significantly outperforms the conventional KCF.

In Fig. 8, we report the precision plots which measures the ratio of successful tracking frames whose tracker output is within the given threshold (the x-axis of the plot, in pixels) from the ground-truth, measured by the center distance between bounding boxes. The overall success and

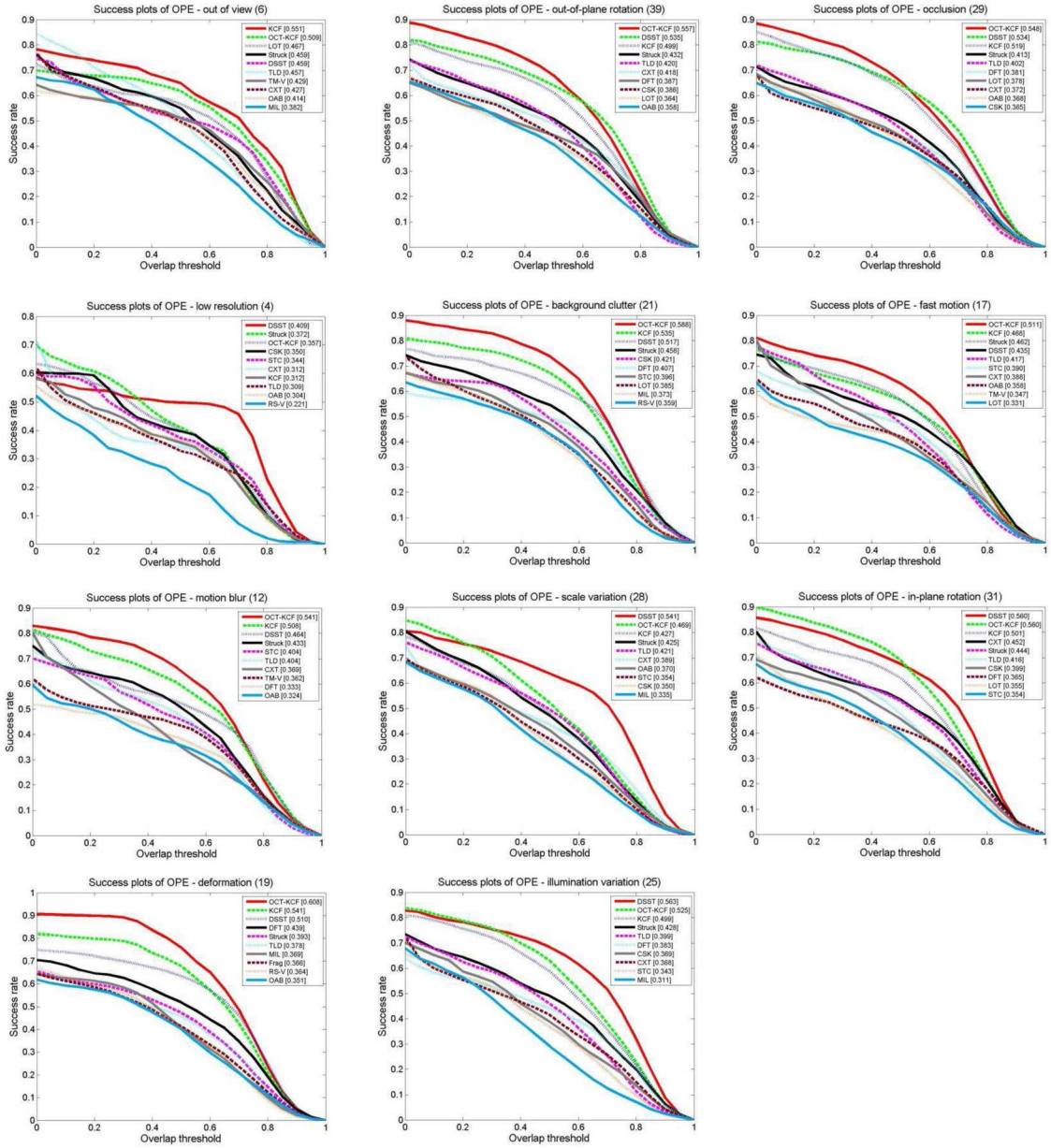


Fig. 11. Success plots for the 11 attributes of the online tracking benchmark.

precision plots generated by the benchmark toolbox are also reported. These plots report top-10 performing trackers in the benchmarks. As shown in Table I, the proposed method reports the best results. The OCT-KCF and KCF achieve 57.4% and 51.7% based on the average success rate, while the famous Struck and TLD trackers, respectively, achieve 47.4% and 43.7%. In terms of Precision, OCT-KCF and KCF, respectively, achieve 83.4% and 74.2% when the threshold is set to 20. We also compare with DSST, one of latest variants of KCF, which shows that OCT-KCF achieves a significant performance improvement in terms of precision (10.7% improved) and success rate (2% improved). These results confirm that the Gaussian prior constraint model contributes to our tracker and enable it performs better than state-of-the-art trackers. The full set of plots generated by the benchmark toolbox are also reported in Figs. 10 and 11. From the experimental results, it can be seen that the proposed OCT-KCF achieves significantly higher performance in cases of in-plane rotation (5.9% improvement over KCF), scale variations (4.2% improvement over KCF), deformations (6.7% improvement over KCF), motion blue (3.3% improvement over KCF) than other trackers (i.e., KCF). This shows that the distribution constrained tracker is more robust to variations mentioned above.

In Fig. 9, we illustrate tracking results from some key frames. In the first row, OCT-KCF can precisely track the coke, while the conventional KCF tracker fails to do that. The famous TLD tracker could relocate the coke target after missing it in 44th frame. Nevertheless the tracking bounding boxes of the TLD tracker is not as precise as those of OCT-KCF. It is also observed that our proposed OCT-KCF tracker works very well in other sequences, e.g., *Couple*, *Deer*, and *Football*. In contrast, all other compared trackers get false or imprecise results in one sequence at least.

On an Intel I5 3.2 GHz (4 cores) CPU and 8GB RAM, the KCF can run up to 185 frames/s, while the OCT-KCF

achieves 51 frames/s. Without losing the real-time performance, the tracking performance is significantly improved by OCT-KCF about 6% on the average success rate and 10% on the precision.

VI. Conclusion

We proposed an OCT method to enhance commonly used correlation filter for object tracking. OCT is a new framework introduced to improve the tracking performance based on the Bayesian optimization method. To improve the robustness of the correlation filter to the variations of the target, the correlation response (output) of the test image is reasonably considered to follow a Gaussian distribution, which is theoretically transferred to be a constraint condition in the Bayesian optimization problem, and successfully used to solve the drifting problem. We obtained a new theory which can transfer the data distribution to be a constraint of an optimization problem, which leads to an efficient framework to calculate correlation filter. Extensive experiments and comparisons on the tracking benchmark show that the proposed method significantly improved the performance of KCF, and achieved a better performance than state-of-the-art trackers. In addition, the performance is obtained without losing the real-time tracking performance. Although high performance is obtained, the drifting detection function (26) is too simple for practical tracking problems, which might fail to start the fine-tuning process when the targets suffer from occlusion or abrupt motion. Therefore, the future work will focus on new drifting detection methods to achieve higher tracking performance. Moreover, we will also try to improve OCT based on other machine learning methods, such as [35]–[39], to solve the long-term tracking problem.

References

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surveys*, vol. 38, no. 4, pp. 81–93, 2006.

[2] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. V. Hengel, "Part-based visual tracking with online latent structural learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, 2013, pp. 2363–2370.

[3] B. Zhang *et al.*, "Bounding multiple Gaussians uncertainty with application to object tracking," *Int. J. Comput. Vis.*, vol. 118, no. 3, pp. 364–379, 2016.

[4] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, 2006, pp. 798–805.

[5] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 125–141, 2008.

[6] B. Zhuang, H. Lu, Z. Xiao, and D. Wang, "Visual tracking via discriminative sparse similarity map," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1872–1881, Apr. 2014.

[7] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image Vis. Comput.*, vol. 21, no. 1, pp. 99–110, 2003.

[8] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[9] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via spatio-temporal context learning," in *Proc. Eur. Conf. Comput. Vis.*, Zürich, Switzerland, 2014, pp. 127–141.

[10] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, Nottingham, U.K., 2014.

[11] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, 2015, pp. 5388–5396.

[12] B. Zhang *et al.*, "Adaptive local movement modelling for robust object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, Mar. 2016, to be published, doi: 10.1109/TCSVT.2016.2540978.

[13] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[14] B. Zhang, A. Perina, V. Murino, and A. Del Bue, "Sparse representation classification with manifold constraints transfer," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, 2015, pp. 4557–4565.

[15] G. Cabanes and Y. Bennani, "Learning topological constraints in self-organizing map," in *Proc. ICONIP*, Sydney, NSW, Australia, 2010, pp. 367–374.

[16] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[17] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, 2013, pp. 2411–2418.

[18] Z. Han, J. Jiao, B. Zhang, Q. Ye, and J. Liu, "Visual object tracking via sample-based adaptive sparse representation," *Pattern Recognit.*, vol. 44, no. 9, pp. 2170–2183, 2011.

[19] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, 2010, pp. 1269–1276.

[20] W. Hu *et al.*, "Incremental tensor subspace learning and its applications to foreground segmentation and tracking," *Int. J. Comput. Vis.*, vol. 91, no. 3, pp. 303–327, 2011.

[21] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1064–1072, Aug. 2004.

[22] C.-T. Chu, J.-N. Hwang, H.-I. Pai, and K.-M. Lan, "Tracking human under occlusion based on adaptive multiple kernels with projected gradients," *IEEE Trans. Multimedia*, vol. 15, no. 7, pp. 1602–1615, Nov. 2013.

[23] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, Aug. 2011.

[24] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proc. Brit. Mach. Vis. Conf.*, vol. 1. Edinburgh, U.K., 2006, pp. 47–56.

[25] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proc. Eur. Conf. Comput. Vis.*, Marseilles, France, 2008, pp. 234–247.

[26] X. Wang, G. Hua, and T. X. Han, "Discriminative tracking by metric learning," in *Proc. Eur. Conf. Comput. Vis.*, Heraklion, Greece, 2010, pp. 200–214.

[27] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Barcelona, Spain, 2011, pp. 263–270.

[28] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2002–2015, Oct. 2014.

[29] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, 2010, pp. 2544–2550.

[30] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. Eur. Conf. Comput. Vis.*, Florence, Italy, 2012, pp. 702–715.

[31] M. Danelljan, F. S. Khan, M. Felsberg, and J. V. D. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 1090–1097.

[32] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," *CoRR*, vol. abs/1510.07945, 2015.

[33] J. Fan, W. Xu, Y. Wu, and Y. Gong, "Human tracking using convolutional neural networks," *IEEE Trans. Neural Netw.*, vol. 21, no. 10, pp. 1610–1623, Oct. 2010.

[34] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer-Verlag, 2006.

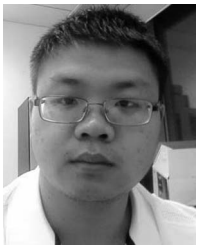
[35] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Inf. Sci.*, vol. 295, no. 1, pp. 395–406, 2015.

[36] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano, and S. Li, "Incremental support vector learning for ordinal regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1403–1416, Jul. 2015.

[37] Z. Lai, Y. Xu, Z. Jin, and D. Zhang, "Human gait recognition via sparse discriminant projection learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 10, pp. 1651–1662, Oct. 2014.
[38] J. Liu *et al.*, "The BeiHang keystroke dynamics systems, databases and baselines," *Neurocomputing*, vol. 144, no. 1, pp. 271–281, 2014.
[39] X. Shi, Y. Yang, Z. Guo, and Z. Lai, "Face recognition by sparse discriminant analysis via joint L2, 1-norm minimization," *Pattern Recognit.*, vol. 47, no. 7, pp. 2447–2453, 2014.

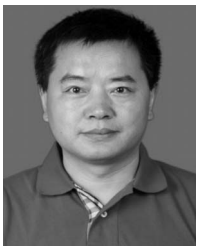
Baochang Zhang received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, 2001, and 2006, respectively.

From 2006 to 2008, he was a Research Fellow with the Chinese University of Hong Kong, Hong Kong, and with Griffith University, Brisbane, QLD, Australia. He is currently an Associate Professor with the Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. He also held a Senior Post-Doctoral Position with PAVIS Department, Italian Institute of Technology, Genoa, Italy. He was supported by the Program for New Century Excellent Talents in University of Ministry of Education of China. His current research interests include pattern recognition, machine learning, face recognition, and wavelets.



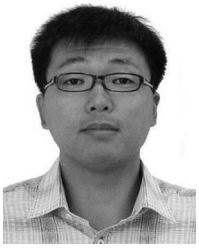
Zhigang Li is currently pursuing the master's degree with Beihang University, Beijing, China.

His research interests include pedestrian detection and object tracking.



Xianbin Cao (M'08–SM'10) received the B.Eng. and M.Eng. degrees in computer applications and information science from Anhui University, Hefei, China, in 1990 and 1993, respectively, and the Ph.D. degree in information science from the University of Science and Technology of China, Beijing, China, in 1996.

He is currently a Professor with the School of Electronic and Information Engineering, Beihang University, Beijing, where he is also the Director of the Laboratory of Intelligent Transportation System. His research interests include intelligent transportation systems, airspace transportation management, and intelligent computation.



Qixiang Ye (SM'14) received the B.S. and M.S. degrees in mechanical and electrical engineering from the Harbin Institute of Technology, Harbin, China, in 1999 and 2001, respectively, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2006.

Since 2015, he has been a Full Professor with the University of Chinese Academy of Sciences, and was a Visiting Assistant Professor with the Institute of Advanced Computer Studies, University of Maryland at College Park, College Park, MD, USA, until 2013. He has published over 50 papers in refereed conferences and journals. His current research interests include image processing, visual object detection, and machine learning.

Dr. Ye was a recipient of the Sony Outstanding Paper Award.

Chen Chen received the B.E. degree in automation from Beijing Forestry University, Beijing, China, in 2009, the M.S. degree in electrical engineering from Mississippi State University, Starkville, MS, USA, in 2012, and the Ph.D. degree from the University of Texas at Dallas, Richardson, TX, USA, in 2016.

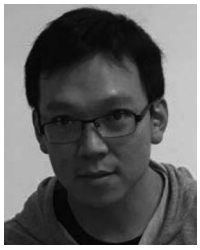
He is currently a Postdoctoral Fellow with the Center for Research in Computer Vision, University of Central Florida, Orlando, FL, USA. His current research interests include compressed sensing, signal and image processing, pattern recognition, and computer vision. He has published over 40 papers in refereed journals and conferences in the above areas.

Linlin Shen received the B.Sc. degree from Shanghai Jiaotong University, Shanghai, China, and the Ph.D. degree from the University of Nottingham, Nottingham, U.K., in 2005.

He was a Research Fellow with the Medical School, University of Nottingham, researching in brain image processing of magnetic resonance imaging. He is currently a Professor and the Director of the Computer Vision Institute, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include Gabor wavelets, face/palmprint recognition, medical image processing, and image classification.

Alessandro Perina received the Ph.D. degree in computer science from the University of Verona, Verona, Italy, with a thesis on the use of generative models in computer vision.

He is a Scientist with Microsoft Corporation, Redmond, WA, USA. From 2006 to 2010, he has been a member of the Vision, Image Processing and Sound Group, University of Verona. From 2010 to 2014, he was an Associate Researcher with Microsoft Research, Redmond, researching with the e-Science Group. From 2014 to 2015, he was a Research Scientist with the Computer Vision and Pattern Analysis Department, Italian Institute of Technology, Genoa, Italy. His current research interests include the use of machine learning techniques and in particular probabilistic graphical models, to engineer solutions to problems in computer vision, multimedia, and text analysis.



Rongrong Ji (SM'14) received the B.S. degree from Harbin Engineering University, Harbin, China, in 2005, and the master's and Ph.D. degree from the Harbin Institute of Technology, Harbin, in 2007, and 2011, respectively.

He is a Professor, the Director of the Intelligent Multimedia Technology Laboratory, and the Dean Assistant of the School of Information Science and Engineering, Xiamen University, Xiamen, China. He has authored over 100 paper published in international journals and conferences. His current research interests include innovative technologies for multimedia signal processing, computer vision, and pattern recognition.

Prof. Ji was a recipient of the Association for Computing Machinery (ACM) Multimedia Best Paper Award and the Best Thesis Award of the Harbin Institute of Technology. He is an Associate/Guest Editor of the international journals and magazines, such as *Neurocomputing*, *Signal Processing*, *Multimedia Tools and Applications*, and the *IEEE Multimedia Magazine and Multimedia Systems*. He also serves as a Program Committee Member for several tier-1 international conference. He is a member of the ACM.