

# Correlation Particle Filter for Visual Tracking

Tianzhu Zhang, *Member, IEEE*, Si Liu, *Member, IEEE*, Changsheng Xu<sup>ID</sup>, *Fellow, IEEE*,  
Bin Liu, and Ming-Hsuan Yang<sup>ID</sup>, *Senior Member, IEEE*

**Abstract**—In this paper, we propose a novel correlation particle filter (CPF) for robust visual tracking. Instead of a simple combination of a correlation filter and a particle filter, we exploit and complement the strength of each one. Compared with existing tracking methods based on correlation filters and particle filters, the proposed tracker has four major advantages: 1) it is robust to partial and total occlusions, and can recover from lost tracks by maintaining multiple hypotheses; 2) it can effectively handle large-scale variation via a particle sampling strategy; 3) it can efficiently maintain multiple modes in the posterior density using fewer particles than conventional particle filters, resulting in low computational cost; and 4) it can shepherd the sampled particles toward the modes of the target state distribution using a mixture of correlation filters, resulting in robust tracking performance. Extensive experimental results on challenging benchmark data sets demonstrate that the proposed CPF tracking algorithm performs favorably against the state-of-the-art methods.

**Index Terms**—Visual tracking, correlation filter, particle filter.

## I. INTRODUCTION

VISUAL tracking is one of the most important tasks in computer vision because of its wide range of applications such as video surveillance, motion analysis, human-computer interaction, and vehicle navigation, to name a few. The main challenge for robust visual tracking is to handle large appearance changes of target objects over time. Although significant progress has been made in recent years, it remains a difficult task to develop robust algorithms for object state estimation for tracking scenarios with challenging factors such as illumination changes, fast motions, pose variations, partial occlusions and background clutter.

Manuscript received June 8, 2016; revised December 9, 2016; accepted January 23, 2017. Date of publication December 8, 2017; date of current version March 13, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61432019, Grant 61572498, Grant 61532009, Grant 61572493, and Grant U1536203, in part by the Beijing Natural Science Foundation under Grant 4172062, and in part by the Key Research Program of Frontier Sciences, CAS, under Grant QYZDJ-SSW-JSC039. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Christopher Wyatt. (*Corresponding author: Si Liu*.)

T. Zhang and C. Xu are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: tzhang10@gmail.com; csxu@nlpr.ia.ac.cn).

S. Liu is with the Beihang University, Beijing Key Laboratory Digital Media, School of Computer Science and Engineering, Beijing 100191, China (e-mail: 272235682@qq.com).

B. Liu is with Moshanghua Tech Co., Ltd., Beijing 100081, China (e-mail: liubin@dress-plus.com).

M.-H. Yang is with the School of Engineering, University of California at Merced, Merced, CA 95344 USA (e-mail: mhyang@ucmerced.edu).

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/TIP.2017.2781304

Visual tracking algorithms can be generally categorized as either generative or discriminative approaches. Generative trackers typically formulate tracking problem as searching for the best image regions which are similar to the tracked targets [4]–[12]. Black *et al.* [4] utilize an off-line subspace model to represent target object for tracking. In [6], the mean shift tracking algorithm models a target with nonparametric distributions of color features and locates the object with mode shifts. In [7], an adaptive appearance model is developed based on the mixture of Gaussians to model objects with three components. The Frag tracker [13] addresses the partial occlusion problem by modeling object appearance with histograms of local patches. However, this method is less effective for handling large appearance changes as the model is not updated. Kwon *et al.* [9] decompose the observation model into multiple basic observation models to cover a wide range of pose and illumination variations. The sparse tracker [11], [14], [15] utilizes a sparse representation to model target appearance. Different from generative trackers, discriminative approaches cast tracking as a classification problem that distinguishes tracked targets from backgrounds [16]–[22]. In [16], a target confidence map is constructed by determining the most discriminative features based on color pixels. Avidan [17] combines a set of weak classifiers into a strong one to do ensemble tracking. Grabner *et al.* [18] propose an online boosting tracking method to update discriminative features. Babenko *et al.* [19] introduce multiple instance learning into online tracking where samples are considered within positive and negative bags or sets. The TLD tracker [20] explicitly decomposes the long-term tracking task into tracking, learning and detection. Hare *et al.* [21] use an online structured output support vector machine for adaptive visual tracking. Zhang *et al.* [22] utilize multiple experts using entropy minimization to address the model drift problem in online tracking.

Recently, tracking methods based on correlation filters have been proved to be able to achieve fairly high speed and robust performance [1], [2], [23]–[33]. For tracking, a correlation filter evaluates the similarity degree by computing the dot product for each possible alignment of a learned template (or filter) relative to a test image sample. The computation of correlation filters can be sped up based on the convolution theorem, which states that the convolution in the spatial domain can be calculated as the element-wise multiplication of the Fourier transforms in the frequency domain. Due to its computational efficiency, correlation filters have attracted considerable attention in visual tracking [2], [25]. Although the CSK [25] and KCF [2] methods achieve the state-of-the-art

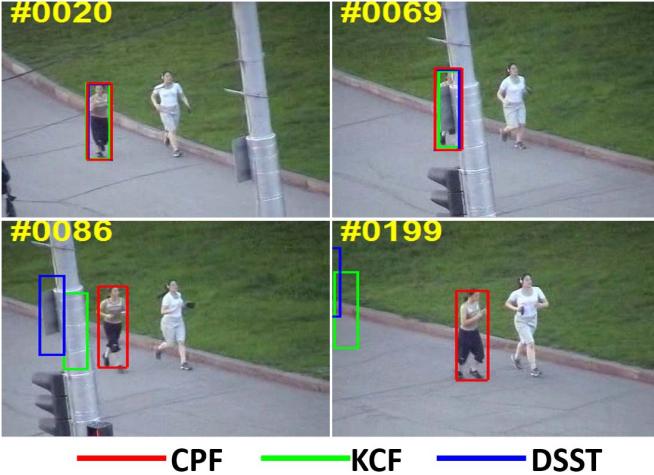


Fig. 1. Comparisons of the proposed CPF tracker with the state-of-the-art correlation filter trackers (DSST [1] and KCF [2]) in challenging situations of heavy occlusions on the *jogging-I* sequence [3]. The proposed tracker uses a particle sampling strategy to maintain multiple hypotheses. It is robust to heavy occlusions and can recover from lost tracks.

performance both in accuracy and robustness, these correlation filter based trackers do not deal with scale variation and occlusion well. To handle the scale variation of target objects, Danelljan *et al.* [1] propose a novel DSST tracker with adaptive multi-scale correlation filters using HOG features. Although the DSST tracker performs well for robust scale estimation by learning discriminative correlation filters based on a scale pyramid representation, it does not deal with partial occlusion or total occlusion well. Figure 1 shows some tracking results on the *jogging-I* sequence by two correlation filter based trackers, DSST [1] and KCF [2], which have achieved state-of-the-art results in terms of accuracy in the VOT challenge [34]. However, these two trackers drift off when the target objects undergo heavy occlusions. These correlation filter based trackers do not handle occlusions well because they use only one single hypothesis, which means these trackers search for the current state of target object only around the previous state. As a result, these trackers are likely to fail when partial occlusion or fast motion occurs.

On the other hand, particle filters [35], [36] can be used to handle large-scale variation and partial occlusion. A particle filter is based on the Bayesian formulation where the samples are propagated over time to maintain multiple hypotheses, and uses a stochastic motion model to predict the states in the next time instance. In a particle-based tracking method, maintaining multiple hypotheses allows the tracker to handle clutters in the background, partial and total occlusions, and recovery from failure or temporary distraction. Thus, particle filters have been widely used in tracking methods for the strength to handle nonlinear target motion and flexibility to incorporate with different object representations. In general, when more particles are sampled and a robust target representation is constructed, particle filter based tracking algorithms are more likely to perform reliably in cluttered and noisy environments. However, the computational cost of particle filter based



Fig. 2. A correlation filter can be used to shepherd particles toward the modes of the target state distribution. The numbers in (b) are the output scores of correlation filter for the particles. Different colored boxes are used to indicate the respective location and scores.

trackers tend to increase linearly with the number of particles, which is the bottleneck for its use in visual tracking. Moreover, particle filter based trackers determine each target object state based on the sampled particles. If the sampled particles do not cover target object states well as shown in Figure 2(a), the predicted target state may be not correct. To overcome this problem, it is better to shepherd the sampled particles toward the modes of the target state distribution.

In this work, we aim to exploit the strength of correlation filters and particle filters and complement each other: (1) Particle filters provide a probabilistic framework for tracking objects by propagating the posterior density over time based on a factored sampling technique. With dense sampling, the states for target objects undergoing large-scale variation can be covered. Moreover, particle filters can also handle multi-modal problems by maintaining multiple hypotheses. Therefore, particle filters can effectively help correlation filter handle scale variation and partial occlusion problems. (2) For each sampled particle, a correlation filter can be applied such that particles are shepherded to the local modes of the target object as shown in Figure 2. Here, the sampled states by a standard particle filter are shown in Figure 2(a). With the correlation filter, these particles are shepherded the modes of the target state distribution as shown in Figure 2(b). Therefore, we do not need to draw particles *densely* to maintain multiple modes, because particles are moved to the local maxima actively after the correlation filter analysis. As a result, we can maintain multiple modes using fewer particles in comparison to the conventional particle filter. Since the computational load of a particle-based tracking method depends heavily on the number of drawn particles, correlation filters can be used in these methods for efficient and effective visual tracking.

Motivated by the above observations, we propose a novel Correlation Particle Filter (CPF) for robust visual tracking, which has the advantages of both particle filters and correlation filters, e.g., robustness to partial occlusion as well as scale variation, and computational efficiency. The merits of the proposed CPF tracking method are as follows. (1) The proposed CPF tracker is robust to partial and total occlusions, and can recover from lost tracks by maintaining multiple hypotheses. (2) The proposed CPF tracker can effectively overcome the scale variation problem via a particle sampling strategy as in traditional particle filter. (3) The proposed CPF

tracker can efficiently maintain multiple modes in the posterior density using fewer particles than conventional particle filters do, resulting in low computational cost. (4) The proposed CPF tracker can shepherd the sampled particles toward the modes of the target state distribution using a mixture of correlation filters, resulting in robust tracking performance. During tracking, target object state is estimated as a weighted average of all particles. We evaluate the proposed tracking algorithm on a large-scale benchmark with 50 challenging image sequences [3]. Extensive experimental results show that the proposed CPF tracking algorithm performs favorably against the state-of-the-art methods in terms of accuracy, efficiency, and robustness.

The paper is organized as follows. In Section II, we summarize the methods that are most related to our work. Section III gives a detailed description of the proposed correlation particle filter tracking approach. Experimental results are reported and analyzed in Section IV. We conclude the paper in Section V.

## II. RELATED WORK

Visual tracking has been studied extensively with numerous applications. A comprehensive review of the tracking methods is beyond the scope of the paper, and surveys of this field can be found in [3] and [37]–[41]. In this section, we discuss the methods closely related to this work including trackers based on correlation filter and particle filters in details.

### A. Correlation Filter Trackers

Correlation filters have attracted considerable attention recently in visual tracking due to its computational efficiency and robustness. Bolme *et al.* [23] encode target appearance by learning an adaptive correlation filter which is optimized by minimizing the output sum of squared error (MOSSE). With the use of correlation filters, the MOSSE tracker is computationally efficient with a speed capability several hundreds frames per second. Heriques *et al.* [25] exploit the circulant structure of adjacent image patches in a kernel space and propose the CSK method, which achieves the highest speed in a recent benchmark [3]. The CSK method builds on illumination intensity features and is further improved by using HOG features in the KCF tracking algorithm [2]. Danelljan *et al.* [26] exploit adaptive color attributes by mapping multi-channel features into a Gaussian kernel space, and propose the DSST tracker [1] with adaptive multi-scale correlation filters using HOG features to handle the scale change of target objects. Zhang *et al.* [24] incorporate context information into filter learning and model the scale change based on consecutive correlation responses. Hong *et al.* [27] propose a biology-inspired framework (MUSTer) where short-term processing and long-term processing are cooperated with each other. Ma *et al.* [28] introduce an online random fern classifier as a re-detection component for long-term tracking. Li *et al.* [31] introduce reliable local patches to exploit the use of local contexts and treat the KCF as the base tracker. In [29], a real-time part-based visual object tracker is proposed to handle partial occlusion and other challenging factors. The key idea is to use the correlation filters [2] as part

classifiers, and a structural constraint mask to handle various appearance changes. In contrast, the proposed method can shepherd the sampled particles toward the modes of the target state distribution as shown in Figure 2, and does not need to draw particles densely as in [29] to maintain multiple modes.

### B. Particle Filters

Particle filters or Sequential Monte Carlo (SMC) methods, developed based on the Monte Carlo methodologies, have been widely used in visual tracking in [42]. To guarantee the robustness of particle filters, particle sampling must be sufficient to capture the variations in the state space. A dense sampling of particles brings high computation load, and it conflicts with the low cost demand of real-time visual tracking. Consequently, numerous techniques have been presented to improve the sampling efficiency of particle filtering [36], [43]–[46]. Importance sampling [36] is introduced to obtain better proposal by combining prediction based on the previous configuration with additional knowledge from auxiliary measurements. In [43], the observation likelihood is computed in a coarse-to-fine manner, which allows the computation to quickly focus on the more promising regions. In [44], subspace representations are adopted in the particle filter, which is made efficiently by applying Rao-Blackwellization to the subspace coefficients in the state vector. In [45], the number of particle samples is adjusted according to an adaptive noise component. Different from the existing methods, we adopt correlation filter to shepherd particles toward the modes of the target state distribution and reduce the number of particles.

## III. CORRELATION PARTICLE FILTER TRACKING

To effectively incorporate object appearance variations over time, rather than explicitly modeling object appearance by one correlation filter as existing methods [2], [25], we model object appearance variation as a mixture of correlation filters. We first introduce the use of a mixture of correlation filters and discuss the correlation particle filter model in details.

### A. Mixture of Correlation Filters

The basic idea of the KCF tracker [2] is that many negative samples are employed to enhance the discriminative ability of the track-by-detector scheme while exploring the structure of the circulant matrix for the high efficiency. The KCF tracker models the appearance of a target object using a correlation filter  $\mathbf{w}$  trained on an image patch  $\mathbf{x}$  of  $P \times Q$  pixels, where all the circular shifts of  $\mathbf{x}_{p,q}$ ,  $(p, q) \in \{0, 1, \dots, P - 1\} \times \{0, 1, \dots, Q - 1\}$ , are generated as training samples with a Gaussian function label  $\mathbf{r}_{p,q}$ . The goal is to find the optimal weights  $\mathbf{w}$ ,

$$\mathbf{w} = \arg \min_{\mathbf{w}} \sum_{p,q} |\langle \phi(\mathbf{x}_{p,q}), \mathbf{w} \rangle - \mathbf{r}_{p,q}|^2 + \lambda \|\mathbf{w}\|^2, \quad (1)$$

where  $\phi$  denotes the mapping to a kernel space and  $\lambda$  is a regularization parameter. Using the fast Fourier transformation (FFT) to compute the correlation, this objective function is

minimized as  $\mathbf{w} = \sum_{p,q} \alpha(p,q) \phi(\mathbf{x}_{p,q})$ , and the coefficient  $\alpha$  is calculated as

$$\alpha = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathbf{r})}{\mathcal{F}(\langle\phi(\mathbf{x}), \phi(\mathbf{x})\rangle) + \lambda}\right), \quad (2)$$

where  $\mathbf{r} = \{\mathbf{r}(p, q)\}$ ,  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  denote the Fourier transform and its inverse, respectively. Given the learned  $\alpha$  and target appearance model  $\bar{\mathbf{x}}$ , the tracking task is carried out on an image patch  $\mathbf{u}$  in the new frame with the search window size  $P \times Q$  by computing the response map as

$$\tilde{\mathbf{r}} = \mathcal{F}^{-1}(\mathcal{F}(\alpha) \odot \mathcal{F}(\langle\phi(\mathbf{u}), \phi(\bar{\mathbf{x}})\rangle)), \quad (3)$$

where  $\odot$  is the Hadamard product. Then, the new position of the target is detected by searching for the location of the maximal value of  $\tilde{\mathbf{r}}$ .

The KCF tracker only uses one correlation filter  $\mathbf{w}$  for object appearance modeling. In tracking, object appearance may have significant changes over time because of a number of factors such as illumination and pose changes. To effectively model object appearance distribution, we propose to use a mixture of correlation filters. Here, we adopt  $K$  correlation filters to handle appearance variation, and the proposed model consists of the learned target appearance  $\bar{\mathbf{x}}_k$  and the classifier coefficients  $\alpha_k, k = 1, \dots, K$ . Note that, if we simply combine the  $K$  correlation filters with the same weights, their importance may be unfairly emphasized. Therefore, we have the weight  $\pi_k$  to emphasize the importance of each correlation filter. The proposed mixture correlation filter at time  $t$  is updated as in (4).

$$\begin{aligned} \mathcal{F}(\alpha_k)^t &= (1 - \eta)\mathcal{F}(\alpha_k)^{t-1} + \eta\mathcal{F}(\alpha_k) \\ \mathcal{F}(\bar{\mathbf{x}}_k)^t &= (1 - \eta)\mathcal{F}(\bar{\mathbf{x}}_k)^{t-1} + \eta\mathcal{F}(\mathbf{x}_k) \\ \pi_k^t &= (1 - \eta)\pi_k^t + \eta\pi_k. \end{aligned} \quad (4)$$

Here, the index  $k$  means that the  $k$ -th correlation filter has the maximum response among the  $K$  filters at time  $t$ ;  $\eta$  is a learning rate parameter; and  $\alpha_k$  as well as  $\bar{\mathbf{x}}_k$  are updated by taking the current values into account, and the  $\pi_k$  is the maximal response value of the  $k$ -th correlation filter. After each update, the weights  $\pi_k$  are normalized.

### B. Correlation Particle Filter Model

The proposed correlation particle filter is based on a Bayesian sequential importance sampling technique, which recursively approximates the posterior distribution using a finite set of weighted samples for estimating the posterior distribution of state variables characterizing a dynamic system. It provides a convenient framework for estimating and propagating the posterior probability density function of state variables regardless of the underlying distribution through a sequence of prediction and update steps. Let  $\mathbf{s}_t$  and  $\mathbf{y}_t$  denote the state variable describing the parameters of an object at time  $t$  (e.g. location and scale) and its observation respectively. The problem consists in calculating the posterior probability density function  $p(\mathbf{s}_t|\mathbf{y}_{1:t-1})$  at each time instant  $t$ . This posterior density function can be obtained recursively in two steps, namely prediction and update. The prediction stage uses the probabilistic system transition model  $p(\mathbf{s}_t|\mathbf{s}_{t-1})$  to predict

the posterior distribution of  $\mathbf{s}_t$  given all available observations  $\mathbf{y}_{1:t-1} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{t-1}\}$  up to time  $t-1$ , and is recursively computed as

$$p(\mathbf{s}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{s}_t|\mathbf{s}_{t-1})p(\mathbf{s}_{t-1}|\mathbf{y}_{1:t-1})d\mathbf{s}_{t-1}, \quad (5)$$

where  $p(\mathbf{s}_{t-1}|\mathbf{y}_{1:t-1})$  is known from the previous iteration. When the observation  $\mathbf{y}_t$  is available, it is possible to perform the update step using the Bayes' rule as in (6),

$$p(\mathbf{s}_t|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{s}_t)p(\mathbf{s}_t|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}, \quad (6)$$

where  $p(\mathbf{y}_t|\mathbf{s}_t)$  denotes the observation likelihood. The posterior  $p(\mathbf{s}_t|\mathbf{y}_{1:t})$  can be approximated by a finite set of  $n$  samples  $\{\mathbf{s}_t^i\}_{i=1}^n$  (called particles) as

$$p(\mathbf{s}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^n w_t^i \delta(\mathbf{s}_t - \mathbf{s}_t^i), \quad (7)$$

where  $w_t^i$  are the importance weights associated to the particles and are calculated as

$$w_t^i \propto w_{t-1}^i \frac{p(\mathbf{y}_t|\mathbf{s}_t^i)p(\mathbf{s}_t^i|\mathbf{s}_{t-1}^i)}{q(\mathbf{s}_t^i|\mathbf{s}_{t-1}^i, \mathbf{y}_t)}, \quad (8)$$

where  $q(\cdot)$  is the importance density function which is chosen to be  $p(\mathbf{s}_t|\mathbf{s}_{t-1}^i)$  and this leads to  $w_t^i \propto w_{t-1}^i p(\mathbf{y}_t|\mathbf{s}_t^i)$ . Then, a re-sampling algorithm is applied to avoid the degeneracy problem [35]. In this case, the weights are set to  $w_{t-1}^i = 1/n \forall i$ . Therefore, we can rewrite the importance weights in (9), which are proportional to the likelihood function  $p(\mathbf{y}_t|\mathbf{s}_t^i)$ ,

$$w_t^i \propto p(\mathbf{y}_t|\mathbf{s}_t^i). \quad (9)$$

The above re-sampling step derives the particles depending on the weights of the previous step, then all the new particles receive a starting weight equal to  $1/n$  which will be updated by the next frame likelihood function. With mixture correlation filter, each particle can be shepherded toward the local modes of the target object by use of its circular shift information. For simplicity, we define the mixture correlation filter operator for state calculation  $\mathcal{S}_{mcf} : \mathcal{R}^d \rightarrow \mathcal{R}^d$ , where  $d$  is the state space dimensionality. Note that  $\mathcal{S}_{mcf}$  is a  $d$ -dimension operator, but in the proposed algorithm it operates on the bi-dimensional position sub-space only. Each  $\mathcal{S}_{mcf}$  procedure guides the generated particle over the subspace independently from all the others. The  $\mathcal{S}_{mcf}(\mathbf{s}_t^i)$  is the new state of particle  $\mathbf{s}_t^i$ , which is obtained via the mixture correlation filters. We define the response of the mixture correlation filter for particle  $\mathbf{s}_t^i$  as  $\mathcal{R}_{mcf}(\mathbf{s}_t^i)$  by

$$\mathcal{R}_{mcf}(\mathbf{s}_t^i) = \sum_k \pi_k \mathcal{F}^{-1}(\mathcal{F}(\alpha_k) \odot \mathcal{F}(\langle\phi(\mathbf{y}_t^i), \phi(\bar{\mathbf{x}})\rangle)), \quad (10)$$

where  $\mathbf{y}_t^i$  is the observation of particle  $i$ . Based on the above response, we define  $p(\mathbf{y}_t|\mathbf{s}_t^i) = \mathcal{R}_{mcf}(\mathbf{s}_t^i)$ , and the particle weights are proportional to the response of mixture correlation filter as

$$w_t^i \propto \mathcal{R}_{mcf}(\mathbf{s}_t^i). \quad (11)$$

**Algorithm 1** The Proposed Correlation Particle Filter Tracking Algorithm

---

**Input** : Image Sequences and Initialization.  
**Output**: Tracking Results  $\mathbf{s}_t \forall t$ .

```

1 for each frame do
2   Generate particles using the transition model
       $p(\mathbf{s}_t | \mathbf{s}_{t-1})$  and re-sample them.
3   Shift particles with a mixture correlation filter
       $\mathbf{s}_t^i \rightarrow \mathcal{S}_{mcf}(\mathbf{s}_t^i)$ .
4   Update particle importance weights using (11).
5   Predict target object state using (12).
6 end
```

---

After the proposed mixture correlation filter analysis, the state of each particle is shifted  $\mathbf{s}_t^i \rightarrow \mathcal{S}_{mcf}(\mathbf{s}_t^i)$ . The best state of target object is estimated as

$$\mathbf{E}[\mathbf{s}_t | \mathbf{y}_{1:t}] \approx \sum_{i=1}^n w_t^i \mathcal{S}_{mcf}(\mathbf{s}_t^i). \quad (12)$$

As shown in Algorithm 1, we divide the proposed tracker into four main steps. The first step generates particles using the transition model  $p(\mathbf{s}_t | \mathbf{s}_{t-1})$  and re-samples them. The second step applies the proposed mixture correlation filter to each particle to make it have shifted and reached a stable position. The third step updates the weights using the responses of the mixture correlation filter. Finally, the fourth step calculates the weighted average to obtain the best state as in (12). Note that, the proposed correlation particle filter is different from the traditional particle filter, because it operates on particles that are close to the local maxima. Therefore, the approximation of (12) is performed using particles better representing the state space than those generated by the traditional particle filter. For this reason, these particles are more efficient than those in the traditional particle filter. Moreover, as particles are drawn at different scales, the proposed model can effectively handle scale variation. The proposed CPF algorithm is more than a simple running mixture correlation filter for multiple particles since the particles are filtered and selected depending on their likelihood. This feature gives to the proposed CPF the possibility to treat multi-modal posterior density functions.

#### IV. EXPERIMENTAL RESULTS

In this section, we first introduce an experimental setup. We then show the effect of particle numbers on tracking performance. Finally, we provide extensive comparisons with state-of-the-art trackers. Our algorithm is implemented in Matlab without optimization and runs at 2.5 frames per second on a 2.80 GHz Intel Core2 Duo machine with 16 GB RAM. We will make the source code available to the public.

##### A. Experimental Setup

1) *Features and Parameters*: We use HOG features for image representation as in the KCF tracker [2]. In the proposed mixture correlation filter,  $K$  is set to 3,  $\eta$  is set to 0.02,  $\pi_k$  is initialized as  $1/K$  and updated over time, and all the other

parameters are set to the same values as the KCF tracker. We use the same parameter values for all the experiments. Moreover, all the parameter settings are available in the source code to be released for accessible reproducible research.

2) *Datasets and Evaluation Metrics*: We evaluate the proposed method on the tracking benchmark [3] that includes 50 videos with comparisons to state-of-the-art trackers. The performance of our approach is quantitatively validated by three metrics used in [3] including center location error (CLE), distance precision (DP), and overlap precision (OP). The CLE is computed as the average Euclidean distance between the ground-truth and the estimated center location of the target. The DP is computed as the relative number of frames in the sequence where the center location error is smaller than a certain threshold. As in [3], the DP values at a threshold of 20 pixels are reported. The OP is defined as the percentage of frames where the bounding box overlap surpasses a threshold of 0.5, which correspond to the PASCAL evaluation criterion. We report results using median CLE, DP and OP over all 50 sequences. In addition, the results are also presented using precision and success plots [3]. In the legend, we report the average distance precision score at 20 pixels for each method. The average overlap precision is plotted in the success plot. We report the results at a threshold of 0.5, which corresponds to the PASCAL evaluation criteria. The area under the curve (AUC) is included in the legend.

##### B. Effect of Particle Number on Visual Tracking

In this section, we evaluate how the particle numbers effect the tracking performance as shown in Table II. The proposed CPF tracker is evaluated with different particle numbers on the 50 benchmark sequences, and the success rate and precision scores are shown for each testing. Moreover, we also report the speed of the trackers in average frames per second (FPS) over all the 50 sequences. Based on the results, it is clear that increasing the number of particles can improve tracking performance. However, the tracker becomes slower. To balance the accuracy and efficiency, we use 40 particles through all the experimental evaluations.

In the benchmark [3], there are several particle filters based sparse trackers, which have achieved the state-of-the-art performances, including L1APG [51], ASLA [52], and MTT [53], [54]. The details of the 3 trackers in the benchmark evaluation can be found in [3]. Since randomization is involved in all the above trackers, we repeat the experiments 5 times using the same initial initializations to evaluate the effects of randomization and report the average and standard deviation of tracking accuracy. The average distance precision and overlap success rate of these trackers are L1APG (48.1%, 37.8%), ASLA (60.2%, 47.5%), and MTT (47.8%, 37.2%), respectively. In addition, the FPS are 2.4, 4.5, and 1.0. Compared with these trackers, the proposed CPF method improves the tracking performance significantly with comparable FPS. For the proposed CPF method with 10 particles, both the speed and the accuracy are better than these particle filter based trackers [51]–[53]. Note that, in these trackers [51], [53], hundreds of particles are used. These results show that the

TABLE I

COMPARISON WITH STATE-OF-THE-ART TRACKERS ON THE 50 BENCHMARK SEQUENCES. OUR APPROACH PERFORMS FAVORABLY AGAINST EXISTING METHODS IN MEDIAN OVERLAP PRECISION (OP) (%) AT AN OVERLAP THRESHOLD 0.5, MEDIAN DISTANCE PRECISION (DP) (%) AT A THRESHOLD OF 20 PIXELS AND MEDIAN CENTER LOCATION ERROR (CLE) (IN PIXELS). THE TOP RANK 2 VALUES ARE HIGHLIGHTED BY BOLD WITH RED AND BLUE COLORS, RESPECTIVELY

Metrics	CPF	Struck [21]	CSK [25]	TLD [20]	VTD [9]	LOT [47]	DFT [48]	MEEM [22]	TGPR [49]	MUSTer [27]	DSST [1]	SCM [50]	MIL [19]	L1APG [51]	RPT [31]	KCF [2]
Median OP	<b>94.9</b>	57.5	36.8	51.7	48.4	31.6	38.2	83.8	85.8	<b>94.8</b>	78.9	80.7	29.4	33.3	82.5	69.2
Median DP	<b>97.9</b>	75.1	56.7	61.7	52.8	53.8	50.6	<b>96.5</b>	94.6	<b>97.9</b>	91.8	81.3	40.4	38.3	96.2	87.7
Median CLE	<b>5.3</b>	14.5	25.6	27.5	26.1	38.6	42.9	7.5	9.7	<b>6.9</b>	11.1	13.0	33.5	61.7	8.4	11.9

TABLE II

EFFECT OF PARTICLE NUMBERS ON VISUAL TRACKING PERFORMANCE. THE RESULTS ARE EVALUATED ON THE 50 SEQUENCES IN COMPARISON WITH PARTICLE FILTERS BASED SPARSE TRACKERS (L1APG [51], ASLA [52], AND MTT [53]). HERE, THE SUCCESS RATE, PRECISION, AND FPS ARE SHOWN WITH DIFFERENT PARTICLE NUMBERS. INCREASING THE NUMBER OF PARTICLES CAN IMPROVE TRACKING PERFORMANCE. HOWEVER, THE TRACKER BECOMES SLOWER

Method	# Particles	Success Rate	Precision	FPS
CPF	10	$58.5 \pm 0.83$	$79.4 \pm 0.82$	8.3
	30	$60.9 \pm 0.81$	$81.6 \pm 0.81$	2.8
	40	$62.6 \pm 0.81$	$84.4 \pm 0.82$	2.5
	50	$62.8 \pm 0.82$	$84.6 \pm 0.80$	1.9
L1APG	600	$37.8 \pm 0$	$48.1 \pm 0$	2.4
ASLA	600	$47.5 \pm 0.66$	$60.2 \pm 1.26$	4.5
MTT	600	$37.2 \pm 0.63$	$47.8 \pm 0.76$	1.0

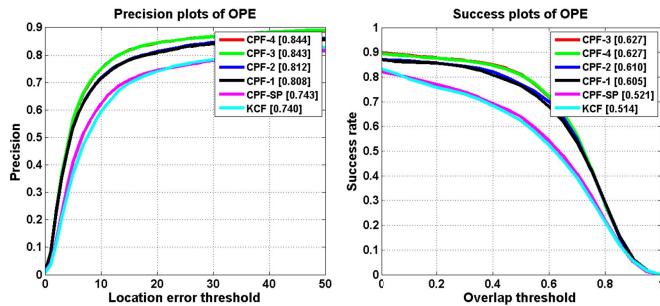


Fig. 3. Precision and success plots over all the 50 sequences [36] using one-pass evaluation (OPE). The legend contains the area-under-the-curve score for each tracker. Here, the CPF- $k$  method uses particle filtering with  $k$  correlation filters, and the tracker CPF-SP uses a mixture of correlation filters ( $k = 3$ ) without particle filtering. These results show that correlation filters and particle filters can enhance and complement each other. Please see the text for more details.

correlation filter can enhance and complement the particle filter, and help cover target object states well. Even with a fewer number of particles, the proposed CPF method can achieve better object tracking performance.

### C. Mixture Correlation Filters vs Particle Filters

Both mixture correlation filters and particle filters can improve object tracking performance. As in Figure 3, we show the experimental results with different number of correlation filters (denoted as CPF- $k$ ,  $k = 1, 2, 3, 4$ ). Here, the CPF- $k$  method uses particle filtering with  $k$  correlation filters. We also add another tracker CPF-SP, which uses a mixture of correlation filters ( $k = 3$ ) without particle filtering. The CPF-SP method is based on the KCF tracker with a mixture

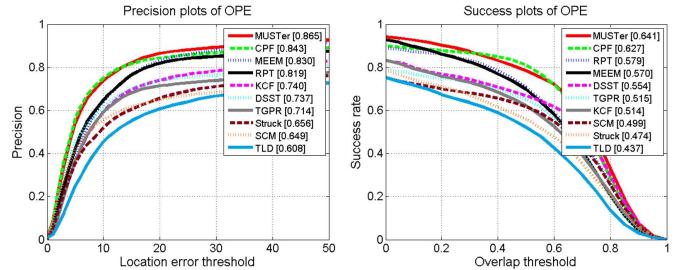


Fig. 4. Precision and success plots over all the 50 sequences using one-pass evaluation (OPE). The legend contains the area-under-the-curve score for each tracker. Among all the trackers, the proposed CPF method achieves the second best results on both the distance precision and overlap success rate. Compared with the MUSTer, the proposed CPF tracker achieves comparable results.

of correlation filters ( $k = 3$ ), and the CPF-1 method is based on the KCF tracker with particle filtering strategy.

From the ablation study, we have the following observations.

**(1) Mixture correlation filters are useful:** The tracking performances of the CPF- $k$  ( $k = 1, 2, 3, 4$ ) are improved with increasing number of correlation filters. Compared with CPF-1, CPF-4 achieves about 3.6% and 2.2% improvement on both the distance precision and overlap success rate. Furthermore, compared with KCF, CPF-SP achieves about 0.3% and 0.7% improvement on precision and overlap metrics.

These results show that the mixture correlation filters can improve tracking performance and the improvement of the mixture correlation filters with particle filters (CPF- $k$ ) is higher than the mixture correlation filters without particle filters (CPF-SP). (2) **Particle filters are useful:** Compared with CPF-SP, CPF-3 achieves much better performance with about 10.0% and 10.6% improvement on precision and overlap metrics. Compared with KCF, CPF-1 has about 6.8% and 9.1% improvement. These results show particle filters can complement correlation filters and significantly improve tracking performance. (3) The above results show that **particle filters are more important than mixture correlation filters**.

**Inters are more important than mixture correlation filters for improving tracking performance.** Let's take the trackers KCF, CPF-SP, and CPF-1 as an example. The CPF-SP and CPF-1 adopt a mixture of correlation filter ( $k = 3$ ) and particle filtering strategy to improve the base tracker KCF, respectively. Compared with the KCF tracker, the CPF-SP achieves about 0.3% and 0.7% improvement on precision and overlap metrics. However, the CPF-1 has about 6.8% and 9.1% improvement on these two metrics. These results agree with the empirical tracking results that KCF are less effective in handling scale variations and occlusions, which can be effectively dealt with

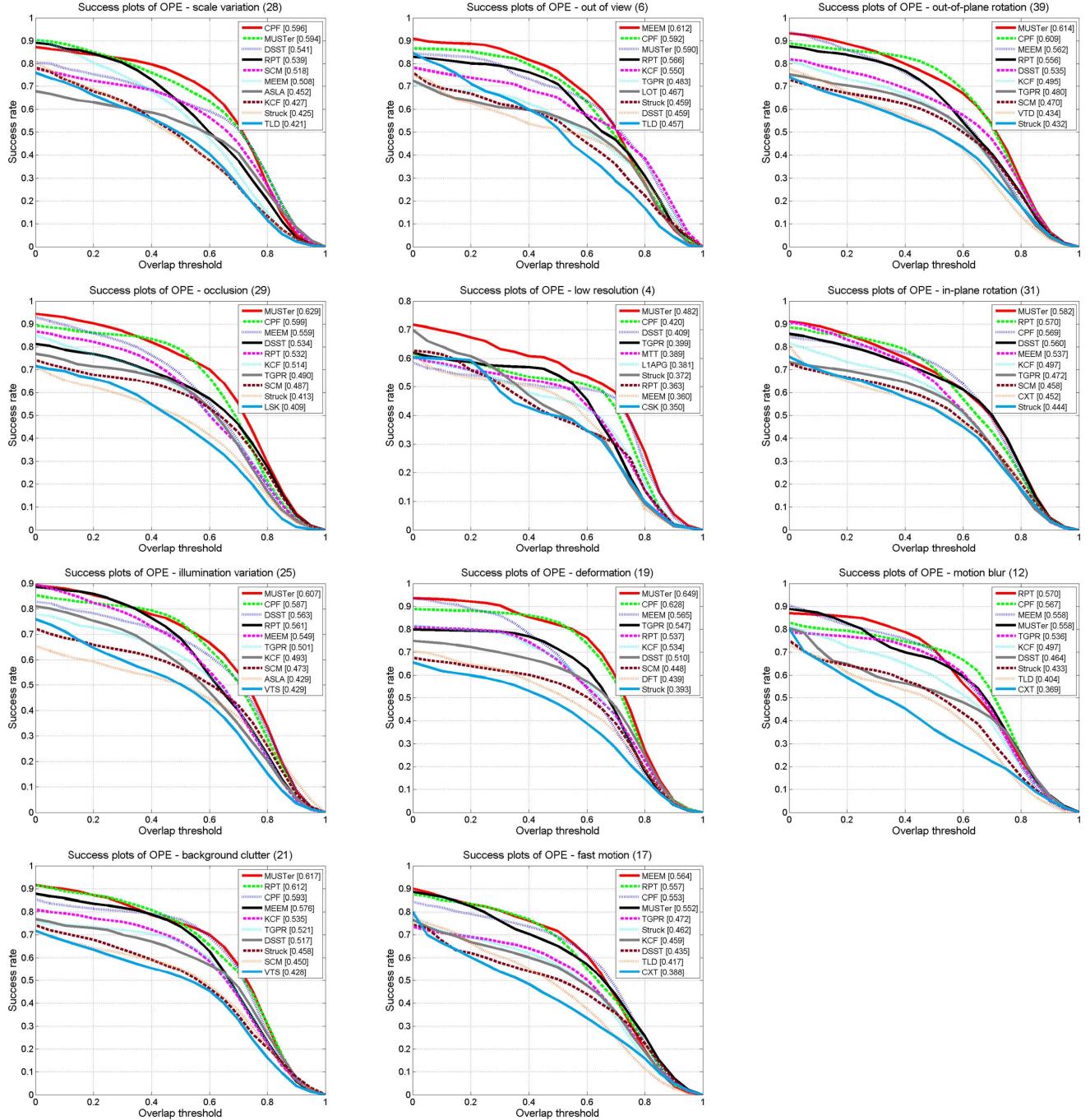


Fig. 5. Success plots over 11 tracking challenges of scale variation, out of view, out-of-plane rotation, low resolution, in-plane rotation, illumination, motion blur, background clutter, occlusion, deformation, and fast motion. The legend contains the AUC score for each tracker. Our CPF method performs favorably against the state-of-the-art trackers.

particle filters by maintaining multiple hypotheses. As such, we propose CPF to exploit and complement the strength of each one.

#### D. Comparison With State-of-the-Art

For more thorough evaluations, we test the proposed algorithm on the benchmark with comparisons to 34 trackers including 29 trackers in [3], and other 5 recently

published state-of-the-art trackers with their shared source code: MEEM [22], TGPR [49], RPT [31], MUSTer [27], DSST [1]. The details of the 29 trackers in the benchmark evaluation can be found in [3]. We present the results using median OP, DP, and CLE over all sequences in Table I, and report the results in one-pass evaluation (OPE) using the distance precision and overlap success rate in Figure 4. The attribute-based evaluation is shown in Figure 5. Furthermore, we compare the center location error and the overlap score

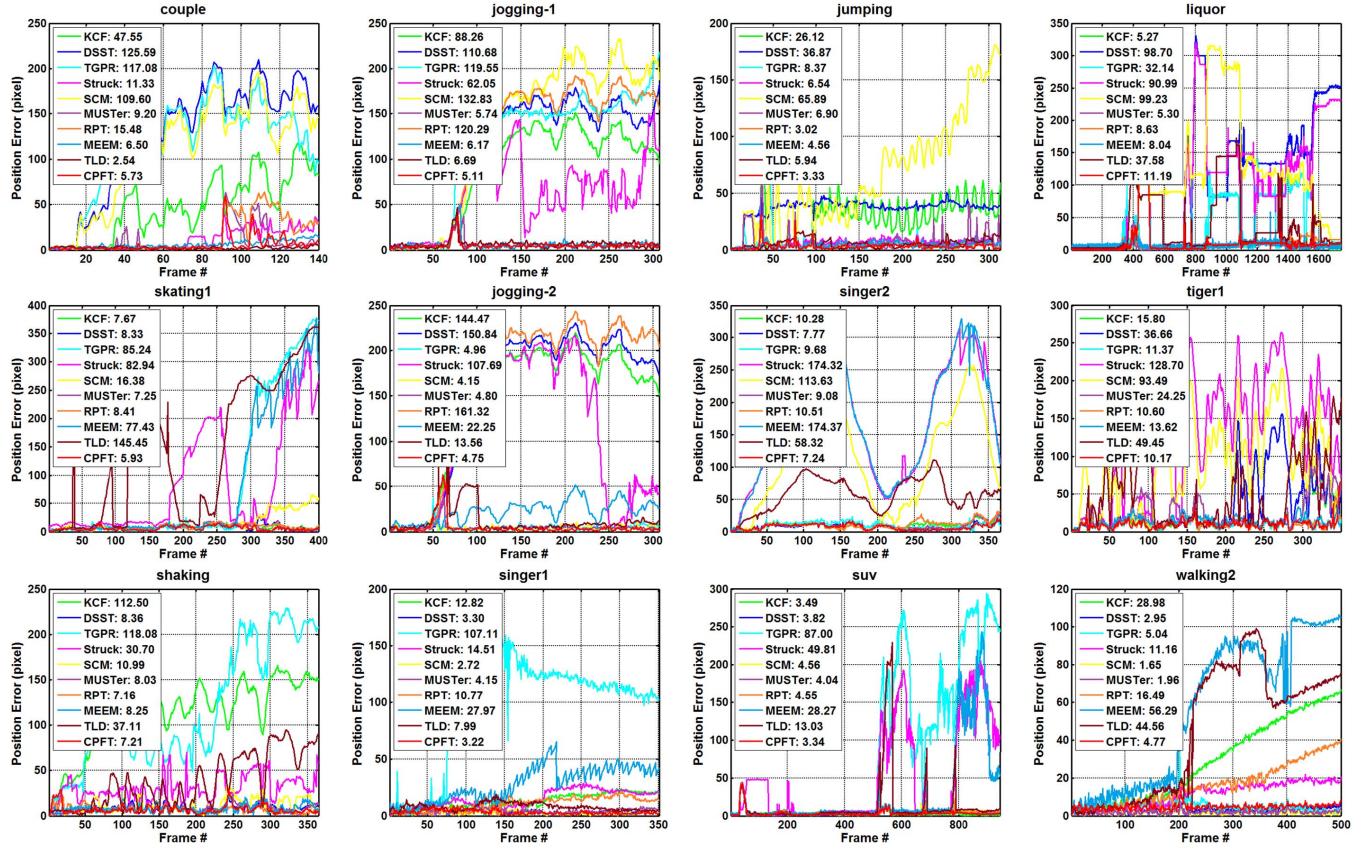


Fig. 6. Frame-by-frame comparison of center location errors (in pixels) on 12 challenging sequences. The average center location errors of the top 10 trackers (highlighted in different colors) for each image sequence are shown in the legend. Generally, our method is able to track targets accurately and stably.

frame-by-frame in Figure 6 and Figure 7. Finally, we show the qualitative comparison on several video sequences in Figure 8.

Table I shows that our algorithm performs favorably against state-of-the-art methods. Among the trackers in the literature, the MUSTer and MEEM methods achieve the first best results with median OP of 94.8%, median DP of 97.9% and median CLE of 6.9 pixels, and the second best results with median OP of 83.8%, median DP of 96.5%, and median CLE of 7.5 pixels, respectively. The proposed algorithm preforms well with median OP of 94.9%, median DP of 97.9%, and median CLE of 5.3 pixels. These results show that the proposed CPF tracker has slightly better tracking performance than the MUSTer and MEEM trackers. Overall, the proposed method achieves significant improvement than other existing trackers. The details are as follows. (1) The Struck, TLD, and SCM trackers are the top 3 methods among the 29 trackers in the benchmark evaluation [3]. The proposed tracking algorithm performs well against the Struck (by 37.4%, 22.8%, 9.2), TLD (by 43.2%, 36.2%, 22.2), and SCM (by 14.2%, 16.6%, 7.7) methods in terms of median OP, DP, and CLE, respectively. (2) Compared with the correlation filter trackers CSK, KCF, and DSST, the proposed CPF method performs well against the CSK (by 58.1%), DSST (by 16%) and KCF (by 25.7%) methods in terms of median OP, and achieves performance gain of 41.2%, 6.1%, and 10.2% in terms of median DP. In terms of average CLE, the proposed CPF method has about 20.3 pixels, 5.8 pixels, and 6.6 pixels improvement.

Figure 4 shows the precision and success plots illustrating the mean distance and overlap precision over all the 50 sequences. In both precision and success plots, our approach shows comparable results compared to the MUSTer and significantly outperforms the existing correlation filter methods (DSST and KCF). For example, the proposed CPF method performs well against the DSST (by 7.3%) and KCF (by 11.3%) methods in terms of success rate. In addition, the proposed tracking algorithm performs well against the DSST (by 10.6%) and KCF (by 10.3%) methods in terms of precision. These results show that particle filter strategy can significantly improve the tracking performances of the correlation filter based trackers. Moreover, in the VOT challenge [34], the DSST [1] and KCF [2] trackers achieve much better performances than existing trackers including particle filters based methods. However, the proposed CPF tracker achieves much better tracking performance than the DSST and KCF trackers. It shows that correlation filter strategy can also improve particle filter based trackers. In summary, the precision and success plots demonstrate that our approach performs well against the existing methods.

In Figure 5, we show the tracking performance based on attributes of image sequences [3], which annotates 11 attributes to describe the different challenges in the tracking problem, e.g., scale variation, out of view, occlusion, deformation. These attributes are useful for analyzing the performance of trackers in different aspects. Here, we present the success

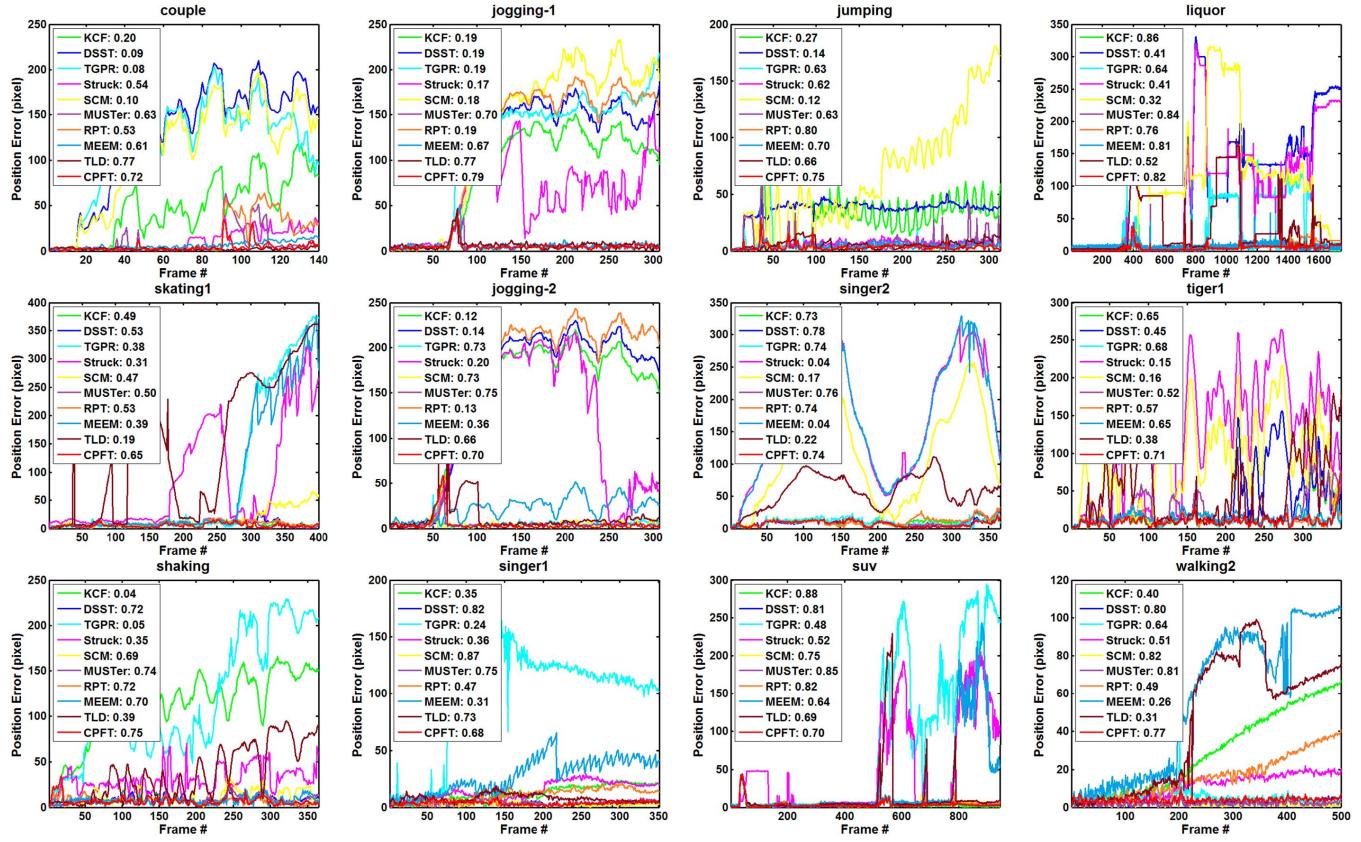


Fig. 7. Frame-by-frame comparison of overlap scores on 12 challenging sequences. The average overlap scores of the top 10 trackers (highlighted in different colors) for each image sequence are shown in the legend. Generally, our method is able to track targets accurately and stably.

plots of OPE for the 11 attributes in Figure 5. For presentation clarity, we present the top 10 performing methods in each plot. We note that the proposed tracking method performs well in dealing with challenging factors including scale variation, occlusion, deformation, fast motion, and out of view. As shown in Figure 5, the KCF tracker does not handle scale variation well, which achieves worse performances than the DSST and CPF methods. The proposed CPF performs better than the DSST tracker by 5.5% in terms of success rate. These results show that the proposed CPF tracker can handle scale variation well. For the occlusion attribute, the proposed CPF achieves the second best among all the trackers, and much better than the DSST and KCF trackers, which demonstrates that the proposed CPF method can handle occlusion problem much better than the two correlation filter based trackers. In details, the proposed CPF performs better than the DSST tracker by 6.5% in terms of success rate. Overall, these results demonstrate that the particle sampling strategy can improve correlation filter based trackers well in handling scale variation and partial occlusion.

We show the center location error and the overlap score frame-by-frame on the 12 sequences in Figure 6 and Figure 7, respectively. The average center location errors and the overlap scores of the top 10 trackers for each image sequence are shown in the legend. These results show our method performs well against the state-of-the-art trackers. Moreover, we compare our algorithm with the top 9 existing

trackers in our evaluation (TGPR [49], SCM [50], Struck [21], RPT [31], KCF [2], MEEM [22], DSST [1], MUSTer [27], and TLD [20]) on the 12 challenging sequences in Figure 8. Overall, these trackers perform well, but the existing trackers have the following issues: The TGPR tracker does not perform well in partial occlusion (*jogging-1*, *suv*) and fast motion (*couple*). The SCM method is not robust for fast motion (*jumping*, *couple*), and the Struck scheme does not handle partial occlusion well (*jogging-1*, *jogging-2*, and *suv*). On the other hand, the RPT approach does not perform well in scale variation (*singer1*, *walking2*). The correlation filter based trackers KCF and DSST drift when target objects undergo heavy occlusion (*jogging-1*, *jogging-2*) and fast motion (*couple*, *jumping*). The MEEM tracker does not handle partial occlusion well (*suv*, *walking2*, and *jogging-2*). We note that the MUSTer tracker drifts off the target object objects when fast motion occurs (*tiger1*, *couple*). The TLD method does not track targets undergoing significant deformation and fast motion (*tiger1* and *shaking*), and scale variation (*singer1*). Overall, our tracker performs well in tracking objects on these challenging sequences.

### E. Discussion

The above results clearly demonstrate the effectiveness and efficiency of the proposed CPF tracker, and correlation filters and particle filters can enhance and complement each other well. Here, we highlight the following conclusions.



Fig. 8. Tracking results of the top 10 trackers (denoted in different colors and lines) in our evaluation on 12 challenging sequences (from left to right and top to down are couple, jogging-1, jumping, liquor, shaking, singer1, skating1, jogging-2, singer2, tiger1, suv, walking2.).

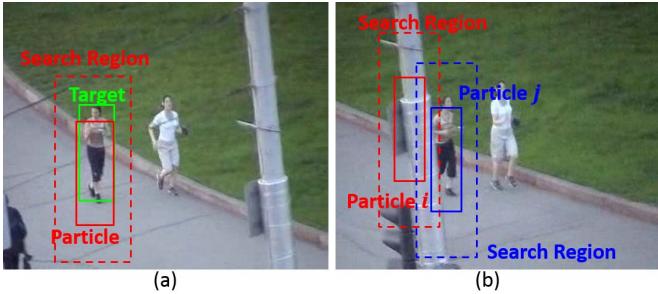


Fig. 9. Correlation filters and particle filters can enhance and complement each other. (a) Correlation filters can shepherd particles toward the modes of the target state distribution and make particle filters cover object state well using fewer particles. (b) Particle filters can improve correlation filters by handling scale variation, and make correlation filters maintain multiple hypotheses with multiple particles for occlusion handling.

**(1)** Particle filters based trackers need to densely sample particles to cover target object states as shown in Figure 2(a) and have high computational cost, which can be improved by correlation filters. Correlation filters can refine particles to cover target object states, thus, effectively reducing the number of particles needed for accurate tracking. This involves two strategies: search region padding and particle refinement. As shown in Figure 9 (a), for a particle (denoted in red bounding box), its search region (denoted in red bounding box with dashed line) is two times the size of the particle and determines the total number of possible circulant shifts of correlation filters. Even though this particle cannot cover

the target object (denoted in green bounding box), its search region can do well. With the correlation filter, the particle can be shepherded toward the modes of the target state distribution. As a result, though the sampled particles do not cover target object states well as shown in Figure 2(a), they will be shepherded toward the modes of the target state distribution to obtain much better object states as shown in Figure 2(b). Therefore, we do not need to draw particles *densely* to maintain multiple modes, because particles are refined by moving to local maxima actively after correlation filters analysis. Thanks to correlation filters, we are able to not only maintain multiple modes to cover object state well using fewer particles in comparison to the conventional particle filter, but also reduce the computational load of a particle-based tracking method depending heavily on the number of drawn particles.

**(2)** Correlation filters based trackers cannot deal with scale variation and partial occlusion, which can be improved by particle filters. **Scale Variation Handling:** Particle filters adopt dense sampling technique to cover the states for target object undergoing large-scale variation. Therefore, particle filters can effectively help correlation filter handle scale variation, which has been demonstrated in Figure 5 over tracking challenge of scale variation. **Occlusion Handling:** Particle filters can maintain multiple hypotheses to allow the tracker to handle clutters in the background, partial and total occlusions, and recovery from failure or temporary distraction. Therefore, particle filters can effectively help correlation filter handle

occlusion problem, which has been demonstrated in Figure 5 over tracking challenge of occlusion. As shown in Figure 9(b), for correlation filters based trackers, if their search region is only the search region of particle  $i$ , these trackers are prone to drift because target object is outside of the search region. However, particle filters can maintain multiple hypotheses with multiple particles, and recovery from failure by using the search region of particle  $j$ .

(3) The proposed CPF has the advantages of both particle filters and correlation filters, e.g., robustness to partial occlusion as well as scale variation, and computational efficiency. (a) As shown in Figure 5 for scale variation attribute, the proposed CPF tracker is robust to partial and total occlusions, and can recover from lost tracks by maintaining multiple hypotheses. (b) As shown in Figure 5 for occlusion attribute, the proposed CPF tracker can effectively overcome the scale variation problem via a particle sampling strategy as in traditional particle filter. (c) As shown in Table II, our CPF tracker can efficiently maintain multiple modes in the posterior density using fewer particles than conventional particle filters do, resulting in low computational cost. (d) As shown in Figure 4, our CPF tracker has robust tracking performance overall and achieves much better performance than correlation filters based trackers and particle filters based trackers.

## V. CONCLUSION

In this paper, we propose a novel correlation particle filter for robust visual tracking. The proposed correlation particle filter can effectively handle partial occlusion and scale variation, and efficiently maintain multiple modes in the posterior density using fewer particles than conventional particle filters. Moreover, it can shepherd particles toward the modes of the target state distribution to obtain robust tracking performance. Experimental results compared with several state-of-the-art methods on challenging sequences demonstrate the effectiveness and robustness of the proposed algorithm.

## REFERENCES

- [1] D. Martin, H. Gustav, K. Fahad, and F. Michael, “Accurate scale estimation for robust visual tracking,” in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–11.
- [2] 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.
- [3] Y. Wu, J. Lim, and M.-H. Yang, “Online object tracking: A benchmark,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2411–2418.
- [4] M. J. Black and A. D. Jepson, “Eigentracking: Robust matching and tracking of articulated objects using a view-based representation,” *Int. J. Comput. Vis.*, vol. 26, no. 1, pp. 63–84, 1998.
- [5] T. Zhang, C. Jia, C. Xu, Y. Ma, and N. Ahuja, “Partial occlusion handling for visual tracking via robust part matching,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1258–1265.
- [6] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-Based Object Tracking,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–575, May 2003.
- [7] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, “Robust online appearance models for visual tracking,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1296–1311, Oct. 2003.
- [8] T. Zhang, B. Ghanem, and N. Ahuja, “Robust multi-object tracking via cross-domain contextual information for sports video analysis,” in *Proc. Int. Conf. Acoust., Speech Signal Process.*, Mar. 2012, pp. 985–988.
- [9] J. Kwon, and K. M. Lee, “Visual tracking decomposition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1269–1276.
- [10] T. Zhang, S. Liu, N. Ahuja, M.-H. Yang, and B. Ghanem, “Robust visual tracking via consistent low-rank sparse learning,” *Int. J. Comput. Vis.*, vol. 111, no. 2, pp. 171–190, 2015.
- [11] X. Mei and H. Ling, “Robust visual tracking and vehicle classification via sparse representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2259–2272, Nov. 2011.
- [12] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, “Low-rank sparse learning for robust visual tracking,” in *Proc. ECCV*, 2012, pp. 470–484.
- [13] A. Adam, E. Rivlin, I. Shimshoni, “Robust fragments-based tracking using the integral histogram,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 798–805.
- [14] T. Zhang *et al.*, “Structural sparse tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 150–158.
- [15] T. Zhang, B. Ghanem, C. Xu, and N. Ahuja, “Object tracking by occlusion detection via structured sparse learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1033–1040.
- [16] R. T. Collins and Y. Liu, “On-line selection of discriminative tracking features,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 346–352.
- [17] S. Avidan, “Ensemble tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 494–501.
- [18] H. Grabner, M. Grabner, and H. Bischof, “Real-time tracking via on-line boosting,” in *Proc. Brit. Mach. Vis. Conf.*, 2006, pp. 1–10.
- [19] B. Babenko, M.-H. Yang, and S. Belongie, “Visual tracking with online Multiple Instance Learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 983–990.
- [20] 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.
- [21] S. Hare, A. Saffari, and P. H. S. Torr, “Struck: Structured output tracking with kernels,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 263–270.
- [22] J. Zhang, S. Ma, S. Sclaroff, “MEEM: Robust tracking via multiple experts using entropy minimization,” in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 188–203.
- [23] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550.
- [24] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, “Fast visual tracking via dense spatio-temporal context learning,” in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 127–141.
- [25] 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.*, 2012, pp. 702–715.
- [26] M. Danelljan, F. S. Khan, M. Felsberg, and J. van de Weijer, “Adaptive color attributes for real-time visual tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1090–1097.
- [27] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, “Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 749–758.
- [28] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, “Long-term correlation tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5388–5396.
- [29] T. Liu, G. Wang, and Q. Yang, “Real-time part-based visual tracking via adaptive correlation filters,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4902–4912.
- [30] T. Zhang, C. Xu, and M.-H. Yang, “Multi-task correlation particle filter for robust object tracking,” in *Proc. CVPR*, 2017, pp. 1–9.
- [31] Y. Li, J. Zhu, and S. C. H. Hoi, “Reliable patch trackers: Robust visual tracking by exploiting reliable patches,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 353–361.
- [32] S. Liu, T. Zhang, X. Cao, and C. Xu, “Structural correlation filter for robust visual tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4312–4320.
- [33] T. Zhang, A. Bibi, and B. Ghanem, “In defense of sparse tracking: Circulant sparse tracker,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3880–3888.
- [34] M. Kristan *et al.*, “The visual object tracking VOT2014 challenge results,” in *Proc. Eur. Conf. Comput. Vis. (ECCV) Vis. Object Tracking Challenge Workshop*, Zurich, Switzerland, 2014, pp. 191–217.
- [35] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

- [36] M. Isard and A. Blake, "CONDENSATION—conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.
- [37] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, 2006, Art. no. 13.
- [38] S. Salti, A. Cavallaro, and L. D. Stefano, "Adaptive appearance modeling for video tracking: Survey and evaluation," *IEEE Trans. Image Process.*, vol. 21, no. 10, pp. 4334–4348, Oct. 2012.
- [39] M. Kristan *et al.*, "The visual object tracking VOT2013 challenge results," in *Proc. ICCV Workshop*, Dec. 2013, pp. 98–111.
- [40] Y. Pang and H. Ling, "Finding the best from the second bests - inhibiting subjective bias in evaluation of visual tracking algorithms," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2784–2791.
- [41] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1442–1468, Jul. 2014.
- [42] M. Isard and A. Blake, "CONDENSATION—conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.
- [43] C. Yang, R. Duraiswami, and L. Davis, "Fast multiple object tracking via a hierarchical particle filter," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2005, pp. 212–219.
- [44] Z. Khan, T. Balch, and F. Dellaert, "A rao-blackwellized particle filter for eigentracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun./Jul. 2004, pp. II-980–II-986.
- [45] S. K. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1491–1506, Nov. 2004.
- [46] A. Doucet, N. De Freitas, and N. Gordon, in *Sequential Monte Carlo Methods in Practice*. New York, NY, USA: Springer-Verlag, 2001.
- [47] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan, "Locally orderless tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1940–1947.
- [48] L. Sevilla-Lara and E. Learned-Miller, "Distribution fields for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1910–1917.
- [49] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with Gaussian processes regression," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 188–203.
- [50] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparsity-based collaborative model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1838–1845.
- [51] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1830–1837.
- [52] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1822–1829.
- [53] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2042–2049.
- [54] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via structured multi-task sparse learning," *Int. J. Comput. Vis.*, vol. 101, no. 2, pp. 367–383, 2013.



**Tianzhu Zhang** (M'11) received the bachelor's degree in communications and information technology from the Beijing Institute of Technology, Beijing, China, in 2006, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011.

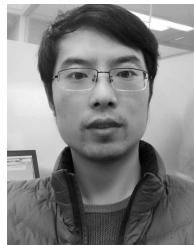
He is currently an Associate Professor with the Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision and multimedia, especially action recognition, object classification, object tracking, and social event analysis.



**Si Liu** (M'12) received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences. She was an Associate Professor with the Institute of Information Engineering, Chinese Academy of Sciences, and a Research Fellow with the Learning and Vision Research Group, Department of Electrical and Computer Engineering, National University of Singapore. She is currently an Associate Professor with the Beijing Key Lab Digital Media, School of Computer Science and Engineering, Beihang University. Her current research interests include object categorization, object detection, image parsing, and human pose estimation.



**Changsheng Xu** (M'97–SM'99–F'14) is a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, and the Executive Director of the China-Singapore Institute of Digital Media. His research interests include multimedia content analysis/indexing/retrieval, pattern recognition, and computer vision. He has held 30 granted/pending patents and published over 200 refereed research papers in these areas. He is an IAPR Fellow and an ACM Distinguished Scientist. He received the Best Associate Editor Award of the *ACM Transactions on Multimedia Computing, Communications and Applications* in 2012 and the Best Editorial Member Award of the *ACM/Springer Multimedia Systems Journal* in 2008. He served as the Program Chair of the ACM Multimedia 2009. He has served as an associate editor, a guest editor, the general chair, the program chair, the area/track chair, the special session organizer, the session chair, and a TPC member for over 20 IEEE and ACM prestigious multimedia journals, conferences, and workshops. He is an Associate Editor of the *IEEE TRANSACTIONS ON MULTIMEDIA*, the *ACM Transactions on Multimedia Computing, Communications and Applications*, and the *ACM/Springer Multimedia Systems Journal*.



**Bin Liu** received the bachelor's degree in electronic engineering from the Southwest University of Science and Technology in 2008 and the M.Sc. degree in electrical engineering from Shanghai Jiao Tong University, China, in 2011. He has a wide-ranging expertise in machine learning, computer vision, artificial intelligence, and computing in mathematics. His current research interests include the use of deep learning methodology for modeling, classification, and tracking.



**Ming-Hsuan Yang** (SM'06) received the Ph.D. degree in computer science from the University of Illinois at Urbana-Champaign in 2000. He is currently an Associate Professor of electrical engineering and computer science with the University of California at Merced, Merced. He is a Senior Member of the ACM. He received the NSF CAREER Award in 2012 and the Google Faculty Award in 2009. He served as an Associate Editor for the *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* from 2007 to 2011. He is an Associate Editor of the *International Journal of Computer Vision, Image and Vision Computing* and the *Journal of Artificial Intelligence Research*.