Structural Correlation Filters Combined with A Gaussian Particle Filter for Hierarchical Visual Tracking

Manna Dai[a,b,c,d], Gao Xiao[d,e,f,g], Shuying Cheng[a,*], Dadong Wang[c,*], Xiangjian He[b,*]

[a] College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, Fujian, P. R. China
[b] Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, NSW 2007, Australia
[c] Commonwealth Scientific and Industrial Research Organisation, Sydney, NSW 2122, Australia
[d] Division of Engineering in Medicine, Department of Medicine, Brigham and Women's Hospital, Harvard Medical School, Cambridge, MA 02139, USA
[e] John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, USA
[f] Wyss Institute for Biologically Inspired Engineering, Harvard University, Boston, Massachusetts 02115, USA.
[g] College of Environment and Resources, Fuzhou University, Fuzhou 350108, Fujian, P. R. China

Abstract

Visual tracking is a key problem for many computer vision applications such as human-computer interaction, intelligent medical diagnosis, navigation and traffic control management. Most of the existing tracking methods are mainly based on correlation filters. However, boundary effect, scale estimation and template updating have not been fully resolved. Herein, this paper presents a new hierarchical tracking method combining structural correlation filters with a Gaussian Particle Filter (GPF), named KCF-GPF. Weak KCF classifiers are constructed via an Lukas-Kanade (LK) method and the preliminary target location is presented as a weighted sum of these classifiers. Specially, a facile weight strategy is implemented to estimate the reliability of each weak classifier. On the basis of the preliminary target location, the GPF using features from a Convolutional Neural Network (CNN) is employed to predict the location and scale of a target. Extensive experiments with the OTB-2013 and the OTB-2015 databases demonstrate that the proposed algorithm performs favourably against state-of-the-art trackers.

Keywords: Structural correlation filter, Gaussian Particle Filter (GPF), Lukas-Kanade (LK), Reliability estimation, Convolutional Neural Network (CNN)

1. Introduction

Visual tracking is one of the most fundamental problems in computer vision due to its numerous applications such as video surveillance, motion analysis, vehicle navigation and human computer interactions[1, 2, 3, 4, 5]. Although a great progress has been seen on developing algorithms [6, 7, 8, 9, 10] and benchmark evaluations [11] for visual tracking, it is still a challenging problem in the situations of heavy illumination changes, pose deformations, partial and full occlusions, large scale variations, background clutter and fast motion.

Correlation Filters (CF) have recently attracted a great attention due to their high speed of calculation and robust tracking performance. Bolme et al. [12] proposed an adaptive correlation filer, called Minimum Output Sum of Squared Error (MOSSE), for producing ASEF-like filters by fewer training images. Henriques et al. [13] extended the CF-based trackers to kernel-based training, called the tracker with the circulant structure and kernels (CSK), to utilize a circulant structure of one image patch to conduct dense sampling, and then improved the kernelized correlation filter (KCF) tracker [14] by using multi-channel inputs and HOG descriptors. Danelljan et al. [15] developed the DSST method handling scale changes of a target, and Choi et al. [16] proposed a spatially attentional weight map to weight various CFs. Ma et al. [17] used the CF as a short-term tracker and an online random fern classifier for re-detection as a long-term memory system.

Although these CF-based trackers achieved appealing results both in precision and success rates with the OTB-

*Corresponding author
Email addresses: daimanna89@gmail.com (Manna Dai), gaoxiao@seas.harvard.edu (Gao Xiao), sycheng@fzu.edu.cn (Shuying Cheng), Dadong.Wang@data61.csiro.au (Dadong Wang), Xiangjian.He@uts.edu.au (Xiangjian He)

2013 [11] and the OTB-2015 databases [18], they still drifted or failed to track due to the following issues. (i) They still underwent boundary effects [19] during target tracking. When a target appeared near boundaries of a detection window of the CF, the boundary information was always discarded and interfered by a cosine window. In this way, boundary effects severely interfered with the progress of target detection in the case of fast motion and motion blur due to relative movements between foreground and background. To resolve the boundary effects, the SRDCF [19] added spatial regularization function that penalized filter coefficients residing outside the target region. (ii) A conventional CF used a fixed-size window to execute learning and detection process, and this kind of trackers lacked estimation on target scale. To handle this scale issue, the SRDCF used different spatial weights to enable the CF to learn and detect in larger image regions with different extents in each frame. As these spatial weights were based on the priori information about the spatial extent of the filter, these weights were not self-adaptive to real tracking scenarios so that the SRDCF also drifted or failed to track. (iii) As a CF-based tracker was a template-class method, it was easy to suffer from fast deformation due to lack of an appropriate strategy for template updating. As mentioned above, the SRDCF extracted different-size templates via priori spatial weights so that these updated templates were inappropriate for a particular application. In general, CF-based trackers are worthy of being studied and improved.

To resolve above issues, we develop a new hierarchical visual tracking method combining structural correlation filters with a Gaussian Particle Filter (GPF). The proposed method can be divided into two layers. The first layer is a homogeneous ensemble layer. An Lukas-Kanade optical flow method (LK) [20] is used to dynamically execute motion detection via estimating instantaneous image velocities between two sequential 2D images. By motion detection, potential locations of a target are preliminary to be found and further redetection will be implemented in these locations for reducing the influence of boundary effects (Issue i). Multiple weak classifiers based on structural correlation filters are generated in these potential locations and all weak classifiers are assembled as a strong classifier. Each weak classifier is a homogeneous base classifier and the weights of all weak classifiers are computed via two new reliability criteria. A preliminary tracking location is inferred by the strong classifier and the scale estimation is executed in the next layer. The second layer is a CNN-based GPF layer. The GPF is used to improve the preliminary visual tracking result by taking scale information of a target into account (Issue ii). The GPF can provide the estimations of location and scale at the same time. Features are extracted by the CNN, which are invariant to rotation, scaling, translation and deformation [21] (Issue iii). The GPF computes the final tracking result by the weighted sum of all particles.

The contributions in this paper are summarized as follows.

- A new homogeneous ensemble strategy is proposed to employ same-type structural correlation filters as homogeneous base classifiers and combine them into a strong classifier.
- The GPF is used for estimating the scale and the location of a target at the same time. A CNN-based feature extraction strategy is introduced into the GPF for reducing the influence of fast deformation of a target.
- Extensive experiments are conducted with the OTB-2013 [11] and the OTB-2015 databases [18] using 11 various attributes to demonstrate the outperformance of the proposed method in comparison with state-of-the-art trackers.

The paper is organized as follows. Section 2 introduces some preliminary methods related to our work for immediate reference. In Section 3, we provide detailed information on the proposed approach. Section 4 presents qualitative and quantitative comparisons of 16 state-of-the-art approaches and 4 baseline trackers with the OTB-2013 and the OTB-2015 databases. At last, some concluding remarks are demonstrated in Section 5.

2. Related Work

A comprehensive tracking review can be found in the previous literatures [11, 22, 23]. In this section, we discuss the methods closely related to this work, mainly regarding structural correlation filters, Gaussian Particle Filters (GPF) and ensemble trackers.

2.1. Structural Correlation Filters

Qi et al. [24] described the weak correlation filers on CNN features in each layer and Liu et al. [25] proposed the concept of the structural correlation filter.

In Qi's research [24], $X^k \in \mathbb{R}^{P\times Q\times D}$ denotes the feature map extracted from the k-th convolutional layer with Gaussian function label $Y \in \mathbb{R}^{P\times Q}$. Let $\mathcal{X}^k = \mathcal{F}(X^k)$

and $\mathcal{Y} = \mathcal{F}(Y)$, where $\mathcal{F}(\cdot)$ represents the discrete Fourier transformation (DFT). The objective function of correlation filter method [24] can be extended into its k-th filter modeled as

$$\mathcal{W}^k = \arg\min_{\mathcal{W}} \|\mathcal{Y} - \mathcal{X}^k \cdot \mathcal{W}\|_F^2 + \lambda \|\mathcal{W}\|_F^2, \quad (1)$$

where

$$\mathcal{X}^k \cdot \mathcal{W} = \sum_{d=1}^{D} \mathcal{X}^k_{*,*,d} \bigodot \mathcal{W}_{*,*,d}. \quad (2)$$

Here, the symbol $\bigodot$ is the element-wise product.

The optimization problem in Eq. 1 has a simple closed form solution, which can be efficiently computed in the Fourier domain by

$$\mathcal{W}^k_{*,*,d} = \frac{\mathcal{Y}}{\mathcal{X}^k \cdot \mathcal{X}^k + \lambda} \bigodot \mathcal{X}^k_{*,*,d}. \quad (3)$$

Given the testing data T^k from the output of the k-th layer, they transform it to the Fourier domain $\mathcal{T}^k = \mathcal{F}(T^k)$, and then the responses can be computed by

$$S^k = \mathcal{F}^{-1}(\mathcal{T}^k \cdot \mathcal{W}^k), \quad (4)$$

where $\mathcal{F}^{-1}$ denotes the inverse of DFT.

The k-th weak tracker outputs the target position with the largest response

$$l(x^k, y^k) = \arg\max_{x^k, y^k} S^k(x^k, y^k). \quad (5)$$

2.2. Gaussian Particle Filters (GPF)

Kotecha and Djuric [26] introduced a Gaussian Particle Filter (GPF), which is used for tracking filtering and predictive distributions encountered in Dynamic State-Space models (DSS) [27]. The DSS model represents the time-varying dynamics of an unobserved state variable. GPF is based on the Particle Filters (PFs) and Gaussian Filters (GFs) concepts. GFs provide Gaussian approximations to the filtering and predictive distributions, and they include Extended Kalman Filter (EKF) [28] and its variations [29, 30]. Unlike EKF, which assumes that predictive distributions are Gaussian and employs linearization of the functions in the process and observation equations, GPF updates the Gaussian approximations using particles. GPF only propagates the posterior mean and covariance of an unobserved state variable in a DSS model, and essentially importance sampling makes the procedure simple.

PF [31] uses Sequential Importance Sampling (SIS) [32] to update the posterior distributions. GPF is quite similar to SIS filters by the fact that Importance Sampling is used to obtain particles. However, a phenomenon called sample degeneration occurs wherein only a few particles representing the distribution have significant weights. A procedure called re-sampling [33] has been introduced to mitigate this problem, but re-sampling is computationally expensive and gives limited results. Since GPF approximates posterior distributions as Gaussians, unlike the SIS filters, particle re-sampling is not required. This results in a reduced complexity of GPF. Furthermore, Berzuini et al. [34] reported that re-sampling of SIS filters is a nonparallel operation. Fortunately, re-sampling would never occur in GPF simulation examples, and GPF is amenable to parallel implementation.

2.3. Ensemble trackers

Multiple component trackers have been combined with hand-crafted features to develop ensemble tracking methods [35, 36, 37] for visual tracking. For example, several ensemble methods [35, 36] using a boosting framework [38] constantly trained each component weak tracker to classify foreground objects and backgrounds. Wang and Yeung used a conditional particle filter to infer the target position and the reliability of each component tracker [37]. Qi et al. [24] treated tracking as a decision-theoretic online learning task and the tracked target was inferred by using decisions from multiple expert trackers. Similar to Qi's study [24], we considered visual tracking as a decision-theoretic online learning task [39], and used it in the structure of multiple correlation filters combining with a GPF. That is, in every round, each correlation filters makes a decision and the final decision is determined by a GPF.

3. Proposed Algorithm

In this section, we present the combination of structural correlation filters with a CNN-based Gaussian Particle Filter for a hierarchical tracking, namely KCF-GPF. The KCF method [14, 13] learns a single correlation filter with a fixed-size window. Different from the KCF method, KCF-GPF is proposed to construct multiple weak correlation filters in a more reliable search scope for dealing with fast motion, motion blur issues and bound effects in the conventional correlation filters. The GPF takes location as well as scale information into account at the same time, and jointly learns particle weights based on CNN features to make a final tracking result. Furthermore, our tracker can effectively handle scale variations via the sampling strategy of a Gaussian Particle Filter. Overall, the proposed ensemble method achieves the following two goals: 1) weak expert trackers are tuned to separate an object from background and

2) the ensemble as a whole ensures the temporal coherence of each part of the tracker.

3.1. Weak Classifiers Based on Structural Correlation Filters

Figure 1 shows a diagram of computing the optical flow using an Lucas-Kanade method (LK). I_x and I_y are an x-axis difference image and a y-axis difference image, respectively. They can be obtained by using the Scharr gradients on the input image. I_t is a time-axis difference image, which is obtained by computing the pixel value differences between two images. These three difference images are utilized to integrate an optical flow image at Frame 28 via the Least Square method (LS) [20]. Original pictures in the first column are from Frame 27 and Frame 28 of Sequence **BlurBody** in the OTB-2013 database [11].

As not all pixels in an image move in the same way between two successive frames in a sequence, we collect an x-axis velocity set of optical flow $OX_t = [ox_t^k]_1^K$ and a y-axis velocity set of optical flow $OY_t = [oy_t^k]_1^K$ consisting of velocity values that appear most often in the t-th frame.

Set $\boldsymbol{\mu}_{t-1} = (x^*_{t-1}, y^*_{t-1}, w^*_{t-1}, h^*_{t-1})$ as the last-frame tracking result, where x^*_{t-1}, y^*_{t-1}, w^*_{t-1} and h^*_{t-1} represent an x-axis position of a target, a y-axis position of a target, a target width and a target height, respectively, in the last frame. The k-th potential location of a target is defined as (x_t^k, y_t^k), which can be computed by

$$x_t^k = ox_t^k + x^*_{t-1}, \tag{6}$$

$$y_t^k = oy_t^k + y^*_{t-1}, \tag{7}$$

where $X_t = [x_t^k]_1^K$ is an x-axis position set and $Y_t = [y_t^k]_1^K$ is a y-axis position set.

In this way, a conventional single KCF is extended to multiple KCFs, and a conventional fixed-size detection window used in a single KCF is extended to reliable multiple detection windows for multiple KCFs.

Kernel Selection: We choose the Gaussian kernel in the existing correlation filter tracker [14].

Feature Representation: Similar to KCF [14], we use HOG features with 31 bins. However, our tracker is quite generic and any dense feature representation with arbitrary dimensions can be incorporated.

Compared to the HDT [24] and SCF [25] methods, which are similar to the proposed weak structural correlation filter, we demonstrate differences among these approaches as follows.

1. The features of HDT are extracted from one layer to build a weak tracker, and the part-based correlation filter SCF samples several parts of a target object to construct features, while KCF-GPF samples in a search scope based on the Lukas-Kanade optical flow method in the t-th frame.
2. In HDT, a target position is made by weighted decisions of all experts, and SCF solves the optimization problem using the fast first-order Alternating Direction Method of Multipliers (ADMM) [40]. Unlike them, KCF-GPF exploits Eq. 16 to infer the ultimate target position.

3.2. Strong Classification via a Homogeneous Ensemble Layer

As shown in Figure 2, an ensemble strategy is used to combine outputs of all weak classifiers to create a strong classifier to detect a target among patches. All used weak classifiers are same-type KCFs, and hence these weak classifiers as base classifiers are homogeneous.

The peak value and the fluctuation of the response map can reveal the confidence degree about the tracking results to some extent. The ideal response map should have only one sharp peak and be smooth in all other areas when the detected target is extremely matched to the correct target. The sharper the correlation peak is, the better the location accuracy is. Otherwise, the whole response map can fluctuate intensely, and its pattern is significantly different from normal response maps. If we continue to use uncertain samples to update the tracking model, it would be corrupted mostly.

The first criterion is called average peak-to-correlation energy (APCE) [8], in oder to measure the fluctuation degree of a response map and the reliability degree of a tracking result. On the basis of Eq. 5, APCE of the k-th KCF in the t-th frame can be defined as

$$APCE_t^k = \frac{|S_t^{k+} - S_t^{k-}|^2}{\text{mean}\left(\sum(S_t^k(x_t^k, y_t^k) - S_t^{k-})^2\right)} \tag{8}$$

where S_t^{k+}, S_t^{k-} denote the maximum and minimum of the response $S_t^k(x_t^k, y_t^k)$ in the t-th frame, respectively. They are defined as below

$$S_t^{k+} = \max_{x_t^k, y_t^k} S_t^k(x_t^k, y_t^k), \tag{9}$$

$$S_t^{k-} = \min_{x_t^k, y_t^k} S_t^k(x_t^k, y_t^k), \tag{10}$$

where $S_t^k(x_t^k, y_t^k)$ is referred to that in Eq. 4 and Eq. 5.

For sharper peaks and less noise, in the case that the target fully appearing in a tracking region, APCE becomes greater and the response map becomes smoother except for only one sharp peak. On the other hand, APCE is small if an object is occluded or missing.

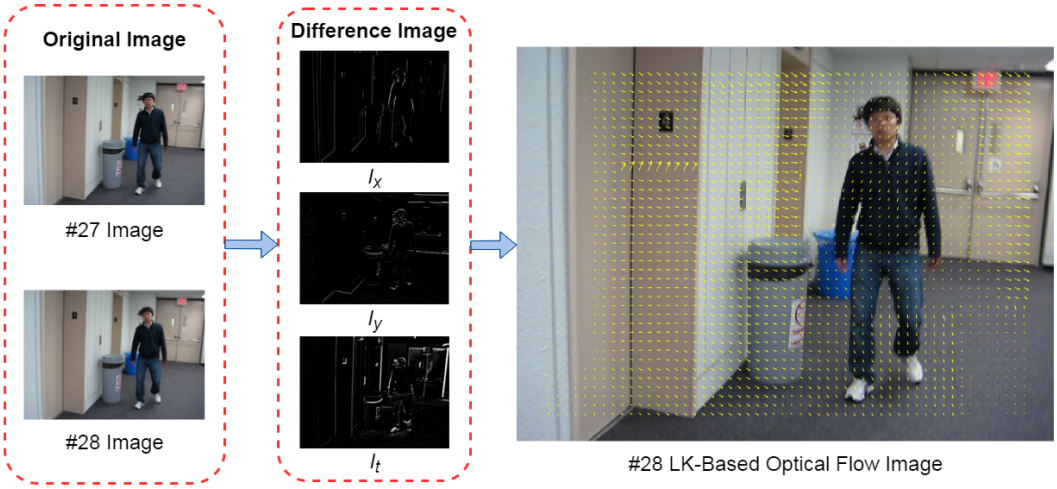


Figure 1: Diagram of computing the optical flow using an Lucas-Kanade method (LK) [20] between two sequential images. The first column shows original full images at Frame 27 and Frame 28 from Sequence **BlurBody** in the OTB-2013 database [11]. The second column denotes an x-axis difference image I_x, a y-axis difference image I_y and a time-axis difference image I_t. I_x and I_y are obtained using the Scharr gradients on the input image. I_t is obtained by computing the pixel value differences between two images. As shown in the third column, these three output images of difference are employed to obtain an optical flow image at Frame 28 via the Least Square method (LS) [20].

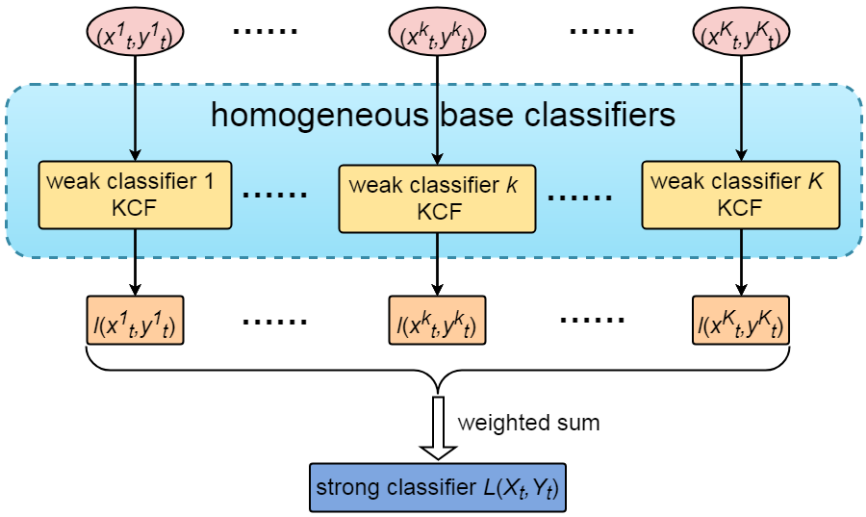


Figure 2: Diagram of the homogeneous ensemble layer. A sample set is generated via Eq. 6 and Eq. 7. Multiple Structural correlation filters are regarded as same-type weak classifiers, which are also called homogeneous base classifiers. Then, all weak classifiers are assembled as a strong classifier via a facile weighted sum strategy based on reliability estimation in Eq. 11.

The second criterion is the maximum response score S_t^{k+} inferred by Eq. 9.

Given reliability degrees of weak classifiers, each weak classifier will be assigned a weight w_t^k to reveal its reliability of tracking performance in the current round. In Eq. 11, w_t^k is the weight of the k-th KCF in the t-th frame and it is defined as

$$w_t^k = w_t^{'} \times w_t^{''}, \tag{11}$$

where $w_t^{'}$ and $w_t^{'}$ are the weights of the k-th weak classifier in the t-th frame in terms of the largest response and APCE, respectively. They are defined as follows

$$w_t^{'} = 1 - \text{sign}(|S_t^{k+} - \max_{1 \le i \le t-1} S_i^{+}|), \tag{12}$$

$$w_t^{''} = 1 - \text{sign}(|APCE_t^k - \max_{1 \le i \le t-1} APCE_i^{+}|), \tag{13}$$

where $|\cdot|$ denotes the absolute value and 'sign' represents the signum function.

As shown in Figure 2, a strong classifier $L(X_t, Y_t)$ is made by the weighted sum of K weak classifier outputs

$$L(X_t, Y_t) = \sum_{k=1}^{K} l(x_t^k, y_t^k) \cdot \hat{w}_t^k, \tag{14}$$

$$\hat{w}_t^k = \frac{w_t^k}{\sum_{k=1}^{K} w_t^k}, \tag{15}$$

where $l(x_t^k, y_t^k)$ is referred to Eq. 5 and $\hat{w}_t^k$ is obtained by normalization of w_t^k.

In the homogeneous ensemble layer, the target position in the t-th frame is inferred as

$$(x'_t, y'_t) = L(X_t, Y_t). \tag{16}$$

3.3. Gaussian Particle Filter Using CNN Features

Figure 3 shows a diagram of tracking using a Gaussian Particle Filter based on CNN features. The GPF in the t-th frame approximates the posterior mean $\boldsymbol{\mu}_t$ and covariance $\boldsymbol{\Sigma}_t$ of the unknown state variable $\mathbf{x}_t$ using Bayesian importance sampling. Samples $\{\mathbf{x}_t^j\}_{j=1}^M$ in the t-the frame are drawn from the importance function $\pi(\cdot)$ by using

$$\pi(\mathbf{x}_t|\mathbf{y}_{0:t}) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_{t-1}, \boldsymbol{\Sigma}_{t-1}), \tag{17}$$

$$\boldsymbol{\mu}_{t-1} = (x'_t, y'_t, w^*_{t-1}, h^*_{t-1}). \tag{18}$$

Here, $\mathbf{y}_{0:t}$ is the observations from the first frame to the t-th frame, and $\mathcal{N}(\cdot)$ represents a Gaussian function.

The respective weights are computed by

$$w_t^j = \frac{p(\mathbf{y}_t|\mathbf{x}_t^j)\mathcal{N}(\mathbf{x}_t = \mathbf{x}_t^j; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)}{\pi(\mathbf{x}_t^j|\mathbf{y}_{0:t})}, \tag{19}$$

where the distribution $p(\mathbf{y}_t|\mathbf{x}_t^j)$ represents the observation equation $\mathbf{y}_t$ conditioned on the unknown state variable $\mathbf{x}_t^j$ in the t-th frame.

Eq. 12 can be rewritten as follows from Eq. 13:

$$w_t^j \propto p(\mathbf{y}_t|\mathbf{x}_t^j). \tag{20}$$

In this paper, we adopt a pre-trained VGG-Net [41] to extract CNN features. Then, we set $p(\mathbf{y}_t|\mathbf{x}_t^j) = |f^* - f(\mathbf{x}_t^j)|$, where $|\cdot|$ denotes the absolute value, $f(\mathbf{x}_t^j)$ is the CNN features of the j-th particle in the t-th frame, and f^* represents the CNN features of the template. Hence, each Gaussian particle weight can be calculated with

$$w_t^j \propto |f^* - f(\mathbf{x}_t^j)|. \tag{21}$$

Normalize the weights as

$$\bar{w}_t^j = \frac{w_t^j}{\sum_{j=1}^{M} w_t^j}. \tag{22}$$

The mean and the covariance in the t-th frame are estimated by

$$\boldsymbol{\mu}_t = \sum_{j=1}^{M} \bar{w}_t^j \mathbf{x}_t^j, \tag{23}$$

$$\boldsymbol{\Sigma}_t = \sum_{j=1}^{M} \bar{w}_t^j (\boldsymbol{\mu}_t - \mathbf{x}_t^j)(\boldsymbol{\mu}_t - \mathbf{x}_t^j)^H, \tag{24}$$

where H represents the Hermitian Matrix.

The mean results $\boldsymbol{\mu}_t$ of all particles with location and scale information are regarded as the final tracking result in the t-th frame. There is

$$(x_t^*, y_t^*, w_t^*, h_t^*) = \boldsymbol{\mu}_t. \tag{25}$$

3.4. KCF-GPF Tracker

Figure 4 illustrates the flowchart of the proposed algorithm. The process can be divided into two layers, namely the homogeneous ensemble layer and the CNN-based GPF layer. In the homogeneous ensemble layer, we execute motion detection using an LK optical flow method to find the potential locations of a target, generate weak classifiers in this potential locations, and assemble the weak classifiers to construct a strong classifier to obtain a target location. In the CNN-based GPF layer, samples are generated in the target location and CNN features are extracted from each sample via a pre-trained VGG-Net. The weight of each sample is measured. Samples with weights are combined to predict the final location and scale of a target.

An overview of the proposed method is summarized in Algorithm 1.

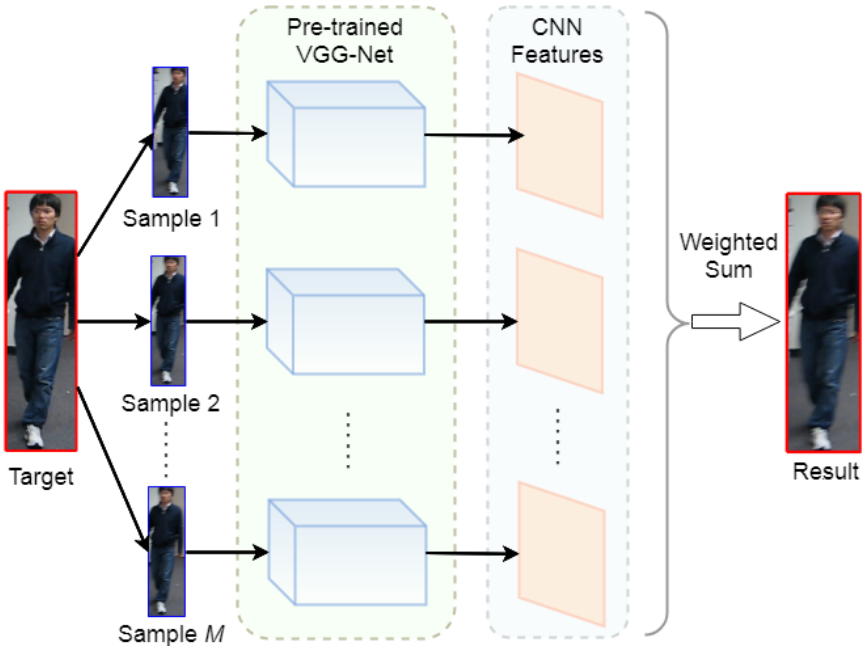


Figure 3: Diagram of tracking using a Gaussian Particle Filter based on CNN features. The first column is an object in the last frame, and the second column denotes M Gaussian random samples with different target locations and different target scales in the current frame. Then, CNN features are extracted from each sample by using a pre-trained VGG-Net. Finally, a weighted sum strategy is employed to obtain the location and scale of a target in the current frame.

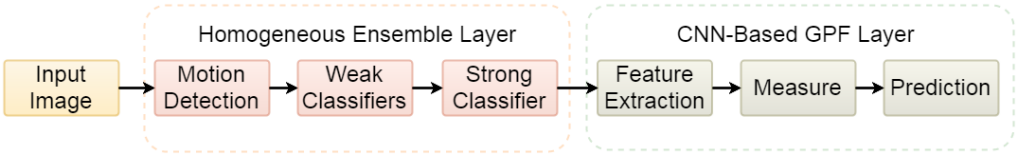


Figure 4: Diagram of the architecture of the proposed KCF-GPF method. In the homogeneous ensemble layer, we execute motion detection using an LK optical flow method to find the potential locations of a target, generate weak classifiers in this potential locations, and assemble the weak classifiers to construct a strong classifier to obtain a target location. In the CNN-based GPF layer, samples are generated in the target location and CNN features are extracted from each sample via a pre-trained VGG-Net. The weight of each sample is measured. Samples with weights are combined to predict a final location and scale of a target.

Algorithm 1: KCF-GPF tracking algorithm

Input: Frames $\{\mathbf{I}_t\}_1^T$;
Output: Target location and target scale in each frame $\boldsymbol{\mu}_t = (x_t^*, y_t^*, w_t^*, h_t^*)$.

for *Frame* $t = 1 : T$ **do**
  if $t = 1$ **then**
    Initialize the target location and the target scale $(x_1^*, y_1^*, w_1^*, h_1^*)$;
  else
    Homogeneous Ensemble Layer:
    Generate a potential location set of a target based on an LK optical flow method via Eq. 6 and Eq. 7, where $k = 1, 2, \ldots, K$;
    Construct K KCFs as weak classifiers based on the location set in the last step and compute their responses using Eq. 4;
    Output K target positions inferred by KCFs using Eq. 5;
    Calculate weights of all KCFs via Eq. 11;
    Construct a strong classifier and output target location via Eq. 16;
    CNN-Based GPF Layer:
    Extract samples of a GPF via Eq. 17;
    Extract CNN features via a VGG-Net;
    Calculate weight of each sample via Eq. 22;
    Estimate the target location and the target scale $(x_t^*, y_t^*, w_t^*, h_t^*)$ using Eq. 25;
  end
end

4. Experiments

Here, we present qualitative and quantitative comparisons of 16 state-of-the-art approaches and 4 baseline trackers with the OTB-2013 database and the OTB-2015 database.

4.1. Experimental Setups

Implementation Details. The conventional features used for KCF-GPF are composed of HOG features and CNN features. Our tracker is implemented on MATLAB on a PC with a 2.40 GHz CPU and achieves 0.18 frame per second (FPS) in Table 1.

Databases. Experimental evaluation is based on the OTB-2013 database [11] consisting of 50 sequences and the OTB-2015 database [18] consisting of 100 sequences. The images are annotated with ground truth bounding boxes and 11 various visual attributes include scale variation, out of view, out-of-plane rotation, low resolution, in-plane rotation, illumination, motion blur, background clutter, occlusion, deformation, and fast motion. In this paper, we show the results based on OTB-2013 and OTB-2015 databases.

Evaluation Metrics. We compare the proposed method with state-of-the-art tracking methods using evaluation metrics and code provided by the respective benchmark datasets. For testing on OTB datasets, we employ the one-pass evaluation (OPE) and use two metrics: precision and success plots. The precision metric computes the rate of frames whose center location is within some certain distance from the ground truth location. The success metric computes the overlap ratio between the tracked and ground truth bounding boxes. In the legend, we report the area under curve (AUC) of success plot and precision score at a 20 pixel threshold (PS) corresponding to the one-pass evaluation for each tracking method.

4.2. Comparison with State-of-the-Art

We evaluate KCF-GPF with the OTB-2013 dataset [11] and compare it with 16 state-of-the-art trackers including LMCF [8], CFNet [7], CFN [42], CFN_ [42], CNT [9], BIT [43], SINT [44], SCT [16], Staple [45], SiamFC [46], SRDCF [19], DSST [47], MEEM [48], KCF [14], TLD [49] and Struck [50]. Among them, LMCF, CFN, CFN_, Staple, KCF, SRDCF, DSST, CFNet and SCT are CF based algorithms. SINT, SiamFC, CFNet, CNT and BIT are convolutional network based algorithms. MEEM is developed based on regression and multiple trackers. TLD is based on an

Table 1: Tracking results of all 17 evaluated trackers over all 50 sequences using OPE evaluation with the OTB-2013. The entries in red denote the best results and the ones in blue indicate the second best.

	LMCF[8]	CFNet[7]	CFN[42]	CFN_ [42]	CNT[9]	BIT[43]
precision	0.842	0.803	0.813	0.784	0.723	0.816
success	0.800	0.775	0.675	0.630	0.656	0.745
	SINT[44]	SCT[16]	Staple[45]	SiamFC[46]	SRDCF[19]	DSST[47]
precision	0.851	0.836	0.793	0.809	0.838	0.740
success	0.791	0.730	0.754	0.783	0.781	0.670
	MEEM[48]	KCF[14]	TLD[49]	Struck[50]	KCF-GPF(ours)	mean FPS(ours)
precision	0.840	0.740	0.608	0.656	0.857	0.18
success	0.706	0.623	0.521	0.559	0.805	

ensemble classifier, and Struck is based on a structured SVM.

4.3. Quantitative Comparison

The characteristics and tracking results are summarized in Table 1. The mean FPS here is estimated on all sequences in the OTB-2013 and achieves 0.18 fps. LMCF achieves the second best performance in terms of the success metric and SINT shows the second best performance in terms of precision metric. Figure 6 illustrates the precision and success plots of all trackers under all challenging attributes in the OTB-2013. KCF-GPF is also superior to other up-to-date trackers with precision and success evaluation metrics using the OTB-2013 benchmark.

For detailed analyses, we also evaluate KCF-GPF with state-of-the-art trackers on various challenging attributes in the OTB-2013 benchmark database and the results are shown in Figure 5. The results demonstrate that KCF-GPF is ranked on top three in each attribute and achieves the best performances in the general success plots. Besides that, the proposed method outperforms other trackers in terms of deformation, out-of-plane rotation and occlusion attributes.

4.4. Qualitative Comparison

To demonstrate the effect of the proposed KCF-GPF algorithm, we make a qualitative comparison with above trackers using the OTB-2013 with 11 different attributes. As shown in Figure 7, these trackers perform well, but the existing trackers have the following issues.

SCT: This tracker cannot work well for the attribute of scale variation in Liquor, Woman and Dog1. This is because that SCT lacks estimation of the scale of a target.

CFNet: CFNet cannot handle occlusion (e.g., in Lemming, Skating 1, Subway, Singer 2, Suv, Liquor, Woman and Soccer), background clutters (e.g., in Skating 1, Subway, Singer 2, Suv, Liquor and Soccer), and deformation (e.g., in Skating1, Subway, suv, Singer 2 and Woman) , and out-of-plane rotation (e.g., in Lemming, Skating1, Singer 2, Liquor, Woman and Soccer). This is due to a lack of reliability estimation on tracking results, and hence CFNet has a large tracking error when a target has a big appearance change.

KCF: KCF drifts when illumination variation occurs (e.g., in Shaking, Lemming and Woman), fast motion (e.g., in Woman and Soccer) and scale variation (e.g., in Shaking, Lemming, Woman and Dog1), and out-of-plane rotation (e.g., in Shaking, Woman, Soccer and Lemming). This is because that KCF suffers from the boundary effect, and a tracking box with fixed size can also limit performance of feature extraction of a target.

TLD: TLD is susceptible to illumination variation (e.g., in Shaking, Skating 1, Singer 2 and Soccer), occlusion (e.g., in Lemming, Subway, Singer 2, Suv, Liquor, Woman and Soccer), and scale variation (e.g., in Lemming and Dog 1). This is because that the used target feature is based on gray images and this feature is susceptible to illumination variation. Meanwhile, the used normalized cross correlation works well in the calculation of overlap rate between a template and a sample when a target does not change a lot. However, this normalized cross correlation fails to align a template with a sample when a target suffers from a severe occlusion and scale variation. Hence, TLD fails to make an accurate tracking.

Struck: Struck is difficult to deal with illumination variation (e.g., in Shaking, Skating 1, Singer 2 and Soccer), occlusion (e.g., in Lemming, Subway, Singer 2, Suv, Liquor, Woman and Soccer), and scale variation

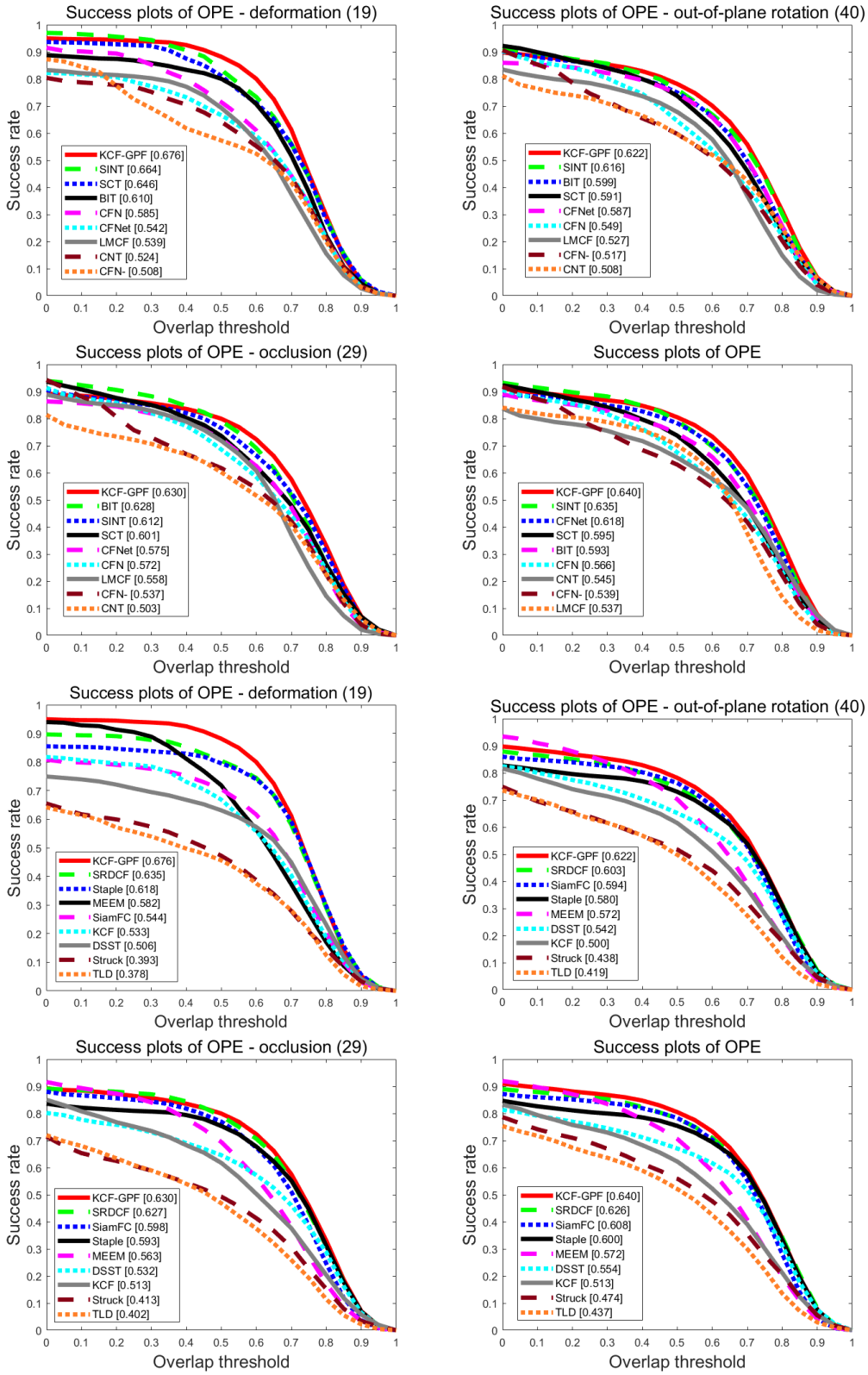


Figure 5: Success plots over all 50 sequences using OPE evaluation with the OTB-2013 dataset. The evaluated trackers are LMCF, CFNet, CFN, CFN_, CNT, BIT, SINT, SCT, Staple, SiamFC, SRDCF, DSST, MEEM, KCF, TLD, Struck and KCF-GPF. All 11 tracking challenges include scale variation, out of view, out-of-plane rotation, low resolution, in-plane rotation, illumination, motion blur, background clutter, occlusion, deformation, and fast motion. The numbers in the legend indicate the average AUC scores for success plots. Our KCF-GPF method performs favorably against the state-of-the-art trackers.

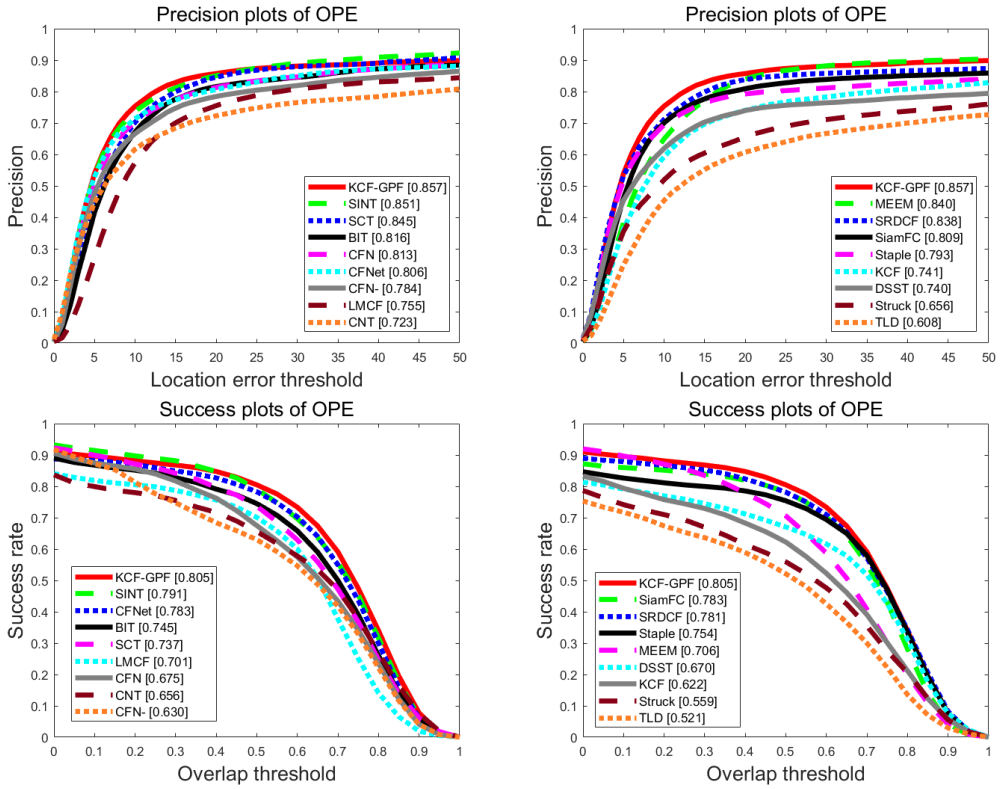


Figure 6: Precision and success plots over all 50 sequences using OPE evaluation with the OTB-2013 database. The numbers in the legend indicate the average precision scores for precision plots and the average AUC scores for success plots. Our KCF-GPF method performs favorably against the state-of-the-art trackers.

(e.g., in Lemming and Dog 1). This is because that the Struck adopts a haar feature, which is susceptible to illumination variation. Due to the limit of detection scope of a tracker, a large scale of orderless movement makes a tracker fail. Therefore, the Struck based on a multi-scale traversal search method is unable to search for a target with a great scale change.

KCF-GPF: In the aspect of general comparison based on 50 videos from the OTB-2013 database, KCF-GPF is superior to other state-of-the-art tracking approaches. This is because that the adopted multiple structural correlation filters can execute a large scale of target search to enable KCF-GPF to be free from the interference of the boundary effect, so as to make KCF-GPF perform well in the scenarios with abrupt large-scale motion. In addition, KCF-GPF also introduces a reliability assessment on each weak classification result for a GPF to execute re-location and scale estimation of a target. Therefore, KCF-GPF has a higher accuracy.

In conclusion, the designed KCF-GPF in this paper can effectively deal with fast motion, background clutter, scale variation and so on. Particularly, KCF-GPF outperforms other 16 representatives of the well-known tracking algorithms in terms of target deformation, out-of-plane rotation and occlusion. This is mainly due to the following factors: (1) KCF-GPF uses multiple weak KCFs to detect a target in a large scale search area, and executes a reliability evaluation on each weak KCF. The most reliable KCF is chosen to provide its tracking result as a reference for a GPF to make a further tracking. The above process not only enables KCF-GPF to make a strong suggestion, but also infers a final location of a target in scenarios with fast motion, deformation, appearance variation and occlusion. (2) KCF-GPF employs an LK optical flow method to perform motion detection so as to provide preliminary target locations for further re-detection. (3) KCF-GPF integrates a GPF method to match a target with various scales, and hence the designed KCF-GPF can reduce the interference of target scale variation. (4) KCF-GPF method uses CNN features, and hence this tracker is invariant to deformation.

4.5. Comparison with baseline trackers

In this section, we show the contribution of each part of the proposed tracking method. The proposed tracking method is the basic framework, and experimental methods are with or without an LK optical flow method, GPF with HOG features and GPF with CNN features.

As showed in Table 2 and Figure 8, the proposed method with an LK method and GPF with CNN features achieves the best results. For a real-time application, the proposed method with an LK method and GPF with HOG features is the best choice, since it achieves the second best results and can satisfy the need for real time.

From the results, an LK method can tackle the motion issues, such as motion blurs and fast motions. GPF can further estimate the location and size of a tracked target.

For detailed analyses, we also evaluate KCF-GPF with state-of-the-art trackers on various challenging attributes in the OTB-2015 database and the results are shown in Figure 8. Results demonstrate that LK+KCF+GPF+CNN is ranked on top three in each attribute and achieves the best performances in the general success plots.

5. Conclusion

In this paper, a new tracker has been proposed to combine multiple structural correlation filters with a Gaussian Particle Filter, namely KCF-GPF. The proposed method has exploited motion detection in successive frames to provide potential tracking locations in order to generate weak classifiers. KCF-GPF has taken multiple structural KCFs as weak classifiers to construct a homogeneous ensemble layer. The reliability degree of each weak classifier has been introduced in experiments as a weight to be assigned to each weak classifier. The ensemble layer has made a preliminary tracking result for GPF via using weighted sum of all results of weak classifiers. As a result, the proposed KCF-GPF has the advantages of the existing correlation filter trackers, such as, computational efficiency and robustness. Moreover, KCF-GPF can deal with scale variations because GPF has taken location and scale estimations into account at the same time. In addition, CNN features have been integrated with GPF so that the KCF-GPF has been invariant to rotation, scaling, translation, and deformation. The proposed KCF-GPF tracking algorithm has outperformed the state-of-the-art methods with the OTB-2013 and OTB-2015 benchmarks in terms of qualitative and quantitative evaluations.

[1] X. Lu, B. Ni, C. Ma, X. Yang, Learning transform-aware attentive network for object tracking, Neurocomputing (2019).

[2] S. Zhai, P. Shao, X. Liang, X. Wang, Fast rgb-t tracking via a cross-modal correlation filters, Neurocomputing 334 (2019) 172–181.

[3] H. Kashiani, S. B. Shokouhi, Visual object tracking based on adaptive siamese and motion estimation network, Image and Vision Computing (2019).

[4] M. Li, Z. Peng, Y. Chen, X. Wang, L. Peng, Z. Wang, G. Yuan, Y. He, A novel reverse sparse model utilizing the spatio-temporal relationship of target templates for object tracking, Neurocomputing 323 (2019) 319–334.

Figure 7: Comparisons of the proposed tracker with the state-of-the-art trackers (SCT [16], CFNet [7], KCF [14], [49] and Struck [50]) in our evaluation on 10 challenging sequences (from left to right and top to down are **Shaking**, **Lemming**, **Skating1**, **Subway**, **Singer2**, **Suv**, **Liquor**, **Woman**, **Soccer**, **Dog1**, respectively).

Table 2: Tracking results of all 4 evaluated trackers over all 100 sequences using OPE evaluation with the OTB-2015. The entries in red denote the best results and the ones in blue indicate the second best.

	KCF	LK+KCF	LK+KCF+GPF+HOG	LK+KCF+GPF+CNN
Success rate	0.475	0.494	0.550	0.563
FPS	152	55	23	0.18

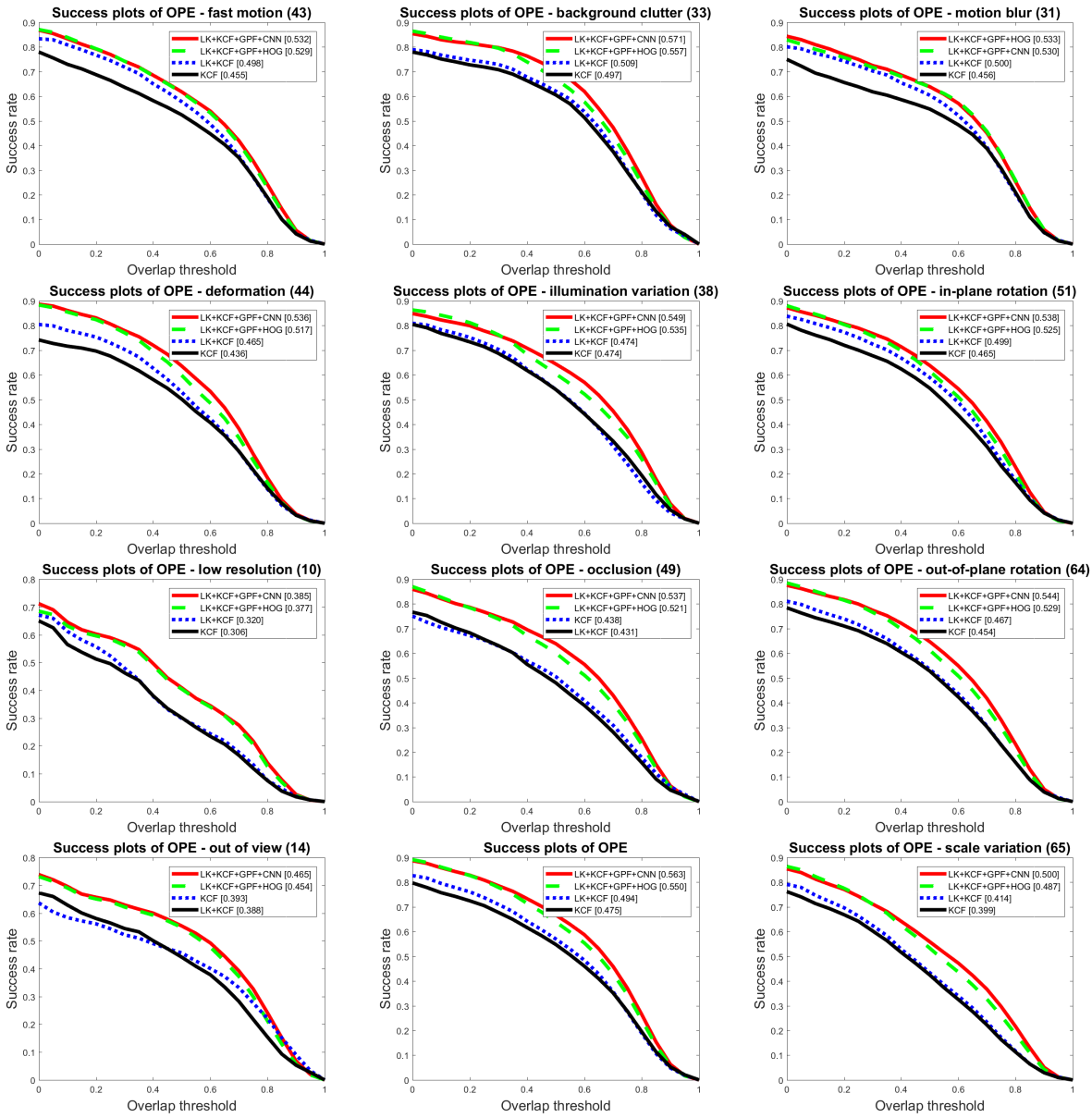


Figure 8: Success plots over all 100 sequences using OPE evaluation in the OTB-2015 database. Our method performs favorably against the state-of-the-art trackers.

[5] A. S. Mendes, G. Villarrubia, J. Caridad, H. Daniel, J. F. De Paz, Automatic wireless mapping and tracking system for indoor location, Neurocomputing (2019).

[6] A. Assa, F. Janabi-Sharifi, K. N. Plataniotis, Sample-based adaptive kalman filtering for accurate camera pose tracking, Neurocomputing 333 (2019) 307–318.

[7] J. Valmadre, L. Bertinetto, J. F. Henriques, A. Vedaldi, P. H. S. Torr, End-to-end representation learning for correlation filter based tracking (2017).

[8] M. Wang, Y. Liu, Z. Huang, Large margin object tracking with circulant feature maps (2017).

[9] K. Zhang, Q. Liu, Y. Wu, M. H. Yang, Robust visual tracking via convolutional networks without training, IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society 25 (2016) 1779.

[10] S. Xu, Y. Ou, J. Duan, X. Wu, W. Feng, M. Liu, Robot trajectory tracking control using learning from demonstration method, Neurocomputing 338 (2019) 249–261.

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

[12] D. S. Bolme, J. R. Beveridge, B. A. Draper, Y. M. Lui, Visual object tracking using adaptive correlation filters, in: Computer Vision and Pattern Recognition, pp. 2544–2550.

[13] J. Henriques, C. Rui, P. Martins, J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, in: European Conference on Computer Vision, pp. 702–715.

[14] J. F. Henriques, C. Rui, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (2015) 583–596.

[15] M. Danelljan, G. Hager, F. S. Khan, M. Felsberg, Discriminative scale space tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (2016) 1561–1575.

[16] J. Choi, H. J. Chang, J. Jeong, Y. Demiris, Y. C. Jin, Visual tracking using attention-modulated disintegration and integration, in: Computer Vision and Pattern Recognition, pp. 4321–4330.

[17] C. Ma, X. Yang, C. Zhang, M. H. Yang, Long-term correlation tracking, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5388–5396.

[18] Y. Wu, J. Lim, M. Yang, Object tracking benchmark, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (2015) 1834–1848.

[19] M. Danelljan, G. Hager, F. S. Khan, M. Felsberg, Learning spatially regularized correlation filters for visual tracking, in: IEEE International Conference on Computer Vision, pp. 4310–4318.

[20] A. Bruhn, J. Weickert, C. Schnörr, Lucas/kanade meets horn/schunck: Combining local and global optic flow methods, International journal of computer vision 61 (2005) 211–231.

[21] Y. Gong, L. Wang, R. Guo, S. Lazebnik, Multi-scale orderless pooling of deep convolutional activation features, in: European conference on computer vision, Springer, pp. 392–407.

[22] A. Yilmaz, Object tracking: A survey, Acm Computing Surveys 38 (2006) 13.

[23] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, M. Shah, Visual tracking: An experimental survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (2014) 1442–1468.

[24] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, M. H. Yang, Hedged deep tracking, in: Computer Vision and Pattern Recognition, pp. 4303–4311.

[25] S. Liu, T. Zhang, X. Cao, C. Xu, Structural correlation filter for robust visual tracking, in: Computer Vision and Pattern Recognition, pp. 4312–4320.

[26] J. H. Kotecha, P. M. Djuric, Gaussian particle filtering, IEEE Transactions on Signal Processing 51 (2003) 2592–2601.

[27] P. J. Harrison, C. F. Stevens, Bayesian forecasting. discussion, Journal of the Royal Statistical Society 38 (1976).

[28] S. J. Julier, J. K. Uhlmann, Unscented filtering and nonlinear estimation, Proceedings of the IEEE 92 (2004) 401–422.

[29] H. W. Sorenson, Recursive estimation for nonlinear dynamic systems, Bayesian Analysis of Time (1988).

[30] G. Welch, G. Bishop, An introduction to the kalman filter, University of North Carolina at Chapel Hill 8 (2010) 127–132.

[31] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, P. J. Nordlund, Particle filters for positioning, navigation, and tracking, IEEE Transactions on Signal Processing 50 (2002) 425–437.

[32] J. S. Liu, R. Chen, T. Logvinenko, A Theoretical Framework for Sequential Importance Sampling with Resampling, Springer New York, 2001.

[33] J. S. Liu, R. Chen, Sequential monte carlo methods for dynamic systems, Journal of the American Statistical Association 93 (1998) 1032–1044.

[34] C. Berzuini, N. G. Best, W. R. Gilks, C. Larizza, Dynamic conditional independence models and markov chain monte carlo methods, Journal of the American Statistical Association 92 (1997) 1403–1412.

[35] S. Avidan, Ensemble tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007) 261–271.

[36] Q. Bai, Z. Wu, S. Sclaroff, M. Betke, C. Monnier, Randomized ensemble tracking, in: IEEE International Conference on Computer Vision, pp. 2040–2047.

[37] N. Wang, D. Y. Yeung, Ensemble-based tracking: aggregating crowdsourced structured time series data, in: International Conference on International Conference on Machine Learning, pp. II–1107.

[38] Y. Freund, R. E. Schapire, A desicion-theoretic generalization of on-line learning and an application to boosting, in: European Conference on Computational Learning Theory, pp. 23–37.

[39] K. Chaudhuri, Y. Freund, D. Hsu, A parameter-free hedging algorithm, Computer Science (2009) 297–305.

[40] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Foundations and Trends in Machine Learning 3 (2011) 1–122.

[41] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil in the details: Delving deep into convolutional nets, arXiv preprint arXiv:1405.3531 (2014).

[42] J. Choi, H. J. Chang, S. Yun, T. Fischer, Y. Demiris, Y. C. Jin, Attentional correlation filter network for adaptive visual tracking, in: IEEE Conference on Computer Vision and Pattern Recognition.

[43] B. Cai, X. Xu, X. Xing, K. Jia, J. Miao, D. Tao, Bit: Biologically inspired tracker, IEEE Transactions on Image Processing 25 (2016) 1327–1339.

[44] R. Tao, E. Gavves, A. W. M. Smeulders, Siamese instance search for tracking, in: Computer Vision and Pattern Recognition, pp. 1420–1429.

[45] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P. H. S. Torr, Staple: Complementary learners for real-time tracking 38 (2015) 1401–1409.

[46] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, P. H. S. Torr, Fully-convolutional siamese networks for object tracking (2016) 850–865.

[47] M. Danelljan, G. Hager, F. S. Khan, M. Felsberg, Accurate scale estimation for robust visual tracking, in: British Machine Vision Conference, pp. 65.1–65.11.

[48] J. Zhang, S. Ma, S. Sclaroff, Meem: Robust tracking via mul-

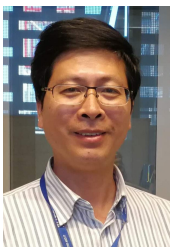
tiple experts using entropy minimization, in: European Conference on Computer Vision, pp. 188–203.
[49] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-learning-detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2012) 1409.
[50] S. Hare, A. Saffari, P. H. S. Torr, Struck: Structured output tracking with kernels, in: IEEE International Conference on Computer Vision, pp. 263–270.

Manna Dai received the Ph.D. degree in computer science from Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. She was affiliated with Data61, Commonwealth Scientific and Industrial Research Organization, Australia. She is currently a Postdoctoral Researcher in Division of Engineering in Medicine, Department of Medicine, Brigham and Women's Hospital, Harvard Medical School, USA. Her research interests include target tracking, machine learning, deep learning, computer vision and image processing.

Gao Xiao received the B.S. and Ph.D. degrees from Sichuan University, Chengdu, China in 2008 and 2014, respectively. Then, he worked as an Assistant Professor at College of Environment and Resources, Fuzhou University, Fuzhou, China. Since 2017, he has been working as a Senior Postdoctoral Researcher in John A. Paulson School of Engineering and Applied Sciences, Harvard University, USA. He is also affiliated with Wyss Institute for Biologically Inspired Engineering, Harvard University, and Brigham and Women's Hospital, Harvard Medical School, USA. His research interests include bio-imaging science, intelligent systems for molecular biology, nanomedicine and medical image processing.

Shuying Cheng received the B.S. and M.D. degrees from Shandong University, Shandong, China in 1988 and in 1991, respectively. Her Ph.D. degree was awarded at the Institute of Fujian Material Structures, Chinese Academy of Science in 2002. She was a post-doctoral fellow in Fuzhou University. She was a Full Professor in Fuzhou University, Fuzhou, China after her master graduation. She was a visiting scholar at the University of New South Wales in Australia. She is currently a professor working with Fuzhou University. Her research interests include object detection, object classification and visual tracking.

Dadong Wang He is a Principal Research Scientist and the leader of the CSIRO Quantitative Imaging Research Team, part of the CSIRO Data61. He is also an adjunct Professor at the University of Technology Sydney and a Conjoint Associate Professor at the University of New South Wales. He has published over 90 research papers, book chapters and reports. He is a senior member of the IEEE. His main research interests include image analysis, computer vision, artificial intelligence, signal processing and software engineering.



Xiangjian He received the Ph.D. in computing sciences from University of Technology Sydney, Australia, in 1999. He had had several hundred of publications including papers appearing in various Elsevier's journals and IEEE Transactions. Since 1999, he has been with University of Technology, Sydney. He is currently the director of Computer Vision and Recognition Laboratory and a full professor at University of Technology Sydney. He is an IEEE Senior Member and has been an IEEE Signal Processing Society Student Committee member. His research interests mainly focuses on image processing, network security, pattern recognition, computer vision, machine learning and deep learning.