Vehicle Tracking at Nighttime by Kernelized Experts With Channel-Wise and Temporal Reliability Estimation

Wei Tian, Long Chen, Ke Zou, and Martin Lauer

***Abstract*— Despite the fact that in recent years, vision-based tracking approaches have made significant progress, the task of tracking vehicles at night still remains challenging. Visual information is strongly deteriorated or at least degraded due to poor illumination conditions. This reduces the perceptive ability of vision systems significantly and can even lead to target loss, resulting in false estimation and/or false prediction of object behavior. In this paper, we propose a novel online-learning method to track vehicles at night. Our method is based on the kernelized correlation filter and assembles different feature channels to kernelized experts. By estimating their reliabilities, we force the appearance model to focus on the most discriminative visual features to accomplish the classification. In addition, a temporal optimization step in conjunction with a memory model is used to remove outliers and keep the most reliable samples to train the tracker models. Experiments over various daytime and weather conditions show that our approach outperforms existing trackers at night and in case of bad weather while offering state-of-the-art performance in more favorable situations. As our tracker has only little computational cost, it is appropriate for use cases with real-time requirements like in automotive or industrial applications.**

***Index Terms*— Nighttime vehicle tracking, correlation filter, kernelized expert, reliability estimation, weather robustness.**

I. Introduction

In recent years, traffic safety, especially on the road, has been increasingly drawing public attention, as road traffic injury is ranked as one of the top causes of human casualties [1]. Simultaneously, advanced driver assistance systems and autonomous car technology are employed to reduce traffic accidents. One key technique of these systems is to detect and track other traffic participants in order to analyze and predict their behaviors. Thus, tracking approaches, especially those based on vision techniques, become more and more popular. Compared to other sensor setups, vision-based systems are relative cheap, lightweight and flexible. Moreover, they can provide rich visual information that can be used to recognize numerous object classes. These properties have led to a considerable improvement of tracking performance [2], [3].

One representative of these methods is the discriminative tracking approach [4], which incorporates machine-learning strategies. Such kind of tracker trains a classifier for each captured object based on the extracted image features. The classifier is applied on the next image to search the candidate with the most similar appearance. This new candidate is then utilized to update both the trained model and the object states (e.g., location in the image). As powerful classifiers like the support vector machine (SVM) can be deployed, high precision in tracking various object classes (including vehicles) can be achieved [5].

While most trackers are dedicated to object tracking with good illumination, e.g., in daylight, tracking objects in poorly illuminated cases such as in the night can be troublesome. The main problem is the degraded contrast between the background and the foreground objects, as reported in [6]. In low exposed images, the conventional visual features used for object recognition, such as the size, shape and color of a vehicle, suffer from different fading effects and are only partially available (Fig. 1 (a)). The most distinguishable parts of a vehicle at night are the brightest ones, mostly coming from the head- and taillights, turn signals, warning lights and stripes, or from other areas which can reflect the light from the street or car lamps (Fig. 1 (b)). Moreover, considering weather factors, e.g., in wet nights, the camera imaging can be interfered by unclear vision conditions such as mist or rain drops, resulting in visual contamination of tracked targets by scattered bright areas in the image, making object recognition much more difficult (Fig. 1 (c)-(d)).

Regardless of hardware upgrades such as improving the exposure quality of cameras, the remaining task in tracking algorithms is how to build trackers which are robust to adverse conditions. In this paper, we tackle this problem on the feature levels. Based on one discriminative tracking approach, i.e., the kernelized correlation filter (KCF), we propose a new kind of online-learning method. In this method, we assemble different feature channels in several kernelized experts. The appearance model for tracked objects is built with respect to their corresponding reliability estimation. Additionally, we conduct an optimization step in temporal domain to remove outliers

Manuscript received April 23, 2017; revised August 11, 2017; accepted November 2, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61773414. The Associate Editor for this paper was S. S. Nedevschi. *(Corresponding author: Long Chen.)*

W. Tian and M. Lauer are with the Institute of Measurement and Control Systems, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany (e-mail: wei.tian@kit.edu; martin.lauer@kit.edu).

L. Chen and K. Zou are with the School of Data and Computer Science, Sun Yat-Sen University, 510006 Guangzhou, China (e-mail: chenl46@mail.sysu.edu.cn; zouk@mail2.sysu.edu.cn).

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/TITS.2017.2771410

Fig. 1. Image samples of vehicles during the night. Image (a) represents a black car with fading color due to low illumination. The only recognizable parts are its taillights, license plate and wheels. Image (b) shows two trucks which can be distinguished by the white trunk and the warning stripes respectively. Images (c)-(d) show the visual contamination in the image by raindrops on the frontal wind shield, with the camera directly mounted behind it. The light from the car lamps scatters to several bright areas in the image, which are much larger than their original size. This scattered light contaminates other parts of the vehicle and makes the recognition of tracked object much more difficult.

and keep the most reliable samples. Based on these two procedures, a classifier is trained, which not only focuses on the discriminative image features under low illumination but is also free from visual contamination by unreliable object candidates, e.g., caused by weather factors. By elaborating the implementation, especially by transferring the major computation task into frequency domain, we demonstrate that our tracker can provide real-time performance.

II. Related Works

As visual object tracking has been an active area among the research community, drastic advances especially in feature selection and model building have been witnessed in the last decade [3], [7]–[9].

According to appearance models, trackers generally can be categorized into two groups: generative and discriminative approaches [4]. In the first category, the object appearance is modeled by generative features such as templates or sparse coding. For instance, [10], [11] extract color histograms from small image patches to represent pedestrians. Additionally, [12], [13] deploy point features to boost the performance in matching objects. In comparison, discriminative approaches reinterpret tracking as a classification problem by training a classifier just in time from previous observations to find the target in the following image. Such kind of a concept establishes the bridge to introduce numerous machine learning technologies into conventional tracking research.

In prior work, classification methods like boosting [14], [15] and random forests [16], [17] have been introduced to build discriminative trackers. As they are ensemble learning approaches, in spite of high tracking precision, the sampling in large data sets brings inevitable heavy computational burdens. Conversely, by integrating localization and classification in the same scheme, significant improvements on computational efficiency has been achieved by structured SVMs [18]. Leveraging the strength of structured SVMs, the correlation filter (CF) [19], [20] has also become attractive in tracking research, as both of them share similar structures.

The CF-based tracking approach has met tremendous progress in recent years. For precision improvement, correlation filters with multiple channels [21], [22] are designed to adopt more discriminative features such as histogram of oriented gradients (HOG) and color attributes. By exploiting training set with circulant structures[1] and introducing the kernel tricks [23], both tracking precision and runtime performance have been further improved. To enhance the classifier training, several strategies are proposed such as spatial regularizations [24], [25], estimating spatial priors [26], learning support vectors [27], multi-memory stores [28] and adaptive training set managements [29]. As deep learning has become more and more popular among the computer vision community, it has also been incorporated in tracker models to obtain more powerful features [30], [31], yet at the cost of decreased computational efficiency. Due to these efforts, CF-based trackers currently achieve top performances for various object classes (including vehicles). However, most of these tracked objects are still observed with relative good illumination, e.g., in daylight.

As visual features are weakened in badly illuminated cases, to track vehicles in the night, most of the research works prefer to locate the bright areas of a vehicle in the image. Representatively, clustering processes on bright objects have been applied in [6] and [32] to grab the head- and taillight patterns. Similar recognition approaches can be seen in [33] but are augmented by geometric and motion pairing. Aside from that, shapes of lights have been also incorporated in [34] to eliminate false objects caused by on-road reflections. In contrast, [35] focuses on the recognition of turn signals and utilizes the Nakagami distribution to build scattering models. Although vehicle lamps appear as quite distinguishable in the night, they may not cover the whole visible area of a vehicle. Other parts with clear contour or color are also worth being considered as appearance features in order to improve the tracking performance. However, such kind of deep digging on available visual information of a tracked target is rare to be seen in most research work. Furthermore, some tracking approaches are based on specific hardware. For instance, a specifically configured camera is utilized in [36] to control the exposure and color processing operations. In [37], tracked objects are verified by an additional sonar sensor, yet with a limited range of measurement. Instead of that, thermal cameras are adopted in [38] to enhance the contrast of objects in the image. However, the performance of thermal sensors can also be interfered by unexpected heat sources such as bonfires.

Regarding these factors, in this paper, we propose a new discriminative approach based on the correlation filter to track vehicles at night. Contributions of this paper are as follows:

- By reliability estimation within kernelized experts, we seek the most discriminative features to build appearance models for targets. Up to our knowledge, we are

[1]Training samples are obtained by cyclic shifting an image patch. Hence, the training set can be considered as a circulant matrix with each sample regarded as a column vector. For details, please refer to [23].

the first that attempt to tackle the problem of tracking vehicles at night on this feature level.
- In order to reduce the contamination of training samples by falsely detected objects, a temporal optimization is conducted in combination with memory models to remove the outliers.
- Based on extensive experiments, we demonstrate that our tracker not only performs robust against weakened visual conditions such as low illumination or bad weathers but also provides real time performance.

III. The Tracking Approach

Here we build our tracker based on the kernelized correlation filter due to its high tracking accuracy and fast processing speed. Furthermore, we decompose the appearance features into different kernelized experts. Based on the reliability of each expert we employ the best features to build appearance models. By estimating the reliability of training samples in the time domain, we reinforce the classification power of the most reliable samples.

A. Kernelized Correlation Filter

For a better understanding of this work, we would like to give a short review of the kernelized correlation filter. In this method, the training samples for the classifier are extracted based on an image patch $\mathbf{x}$ of $M \times N$ pixels with the target located in its center. Instead of sliding windows, each sample $\mathbf{x}_i$ is obtained by circularly shifting the image patch with $i \in \{0, \cdots, M-1\} \times \{0, \cdots, N-1\}$ pixels. Its label $y_i \in [0, 1]$ is calculated by a Gaussian function with respect to the Euclidean distance between the centers of image patch $\mathbf{x}$ and the shifted version $\mathbf{x}_i$. Following the rules of structured SVMs, the classifier can be trained by minimizing the regression errors, formulated as

$$\arg\min_{\mathbf{w}} \sum_{i}^{n} |f(\mathbf{x}_i) - y_i|^2 + \lambda \|\mathbf{w}\|^2, \tag{1}$$

where $\mathbf{w}$ corresponds to the coefficient vector of the classifier $f(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i$ and the non-negative value λ denotes the regularization parameter. The number of samples is denoted by $n = M \cdot N$. Leveraging the circular property of the training set, the objective function (1) can be reformulated as

$$\arg\min_{\mathbf{w}} \|\mathbf{w} * \mathbf{x} - \mathbf{y}\|^2 + \lambda \|\mathbf{w}\|^2, \tag{2}$$

where $\mathbf{y}$ denotes the label vector and the symbol $*$ represents the convolution operation.

For a non-linear classifier model, kernels are introduced into the coefficient vector $\mathbf{w} = \sum_i^n \alpha_i \varphi(\mathbf{x}_i)$ in terms of a non-linear mapping φ. Hence, the coefficient vector $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_n]^T$ becomes the key to solving problem (1). As training samples are extracted in a circular fashion, for shift-invariant kernels, e.g., the Gaussian kernel, according to [23] and [39], the solution $\boldsymbol{\alpha}$ can be obtained in a closed form as

$$\boldsymbol{\alpha} = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathbf{y})}{\mathcal{F}(\mathbf{k}^{\mathbf{xx}}) + \lambda}\right), \tag{3}$$

where $\mathbf{k}^{\mathbf{xx}}$ is the kernelized autocorrelation with elements calculated by the kernel function $k_i^{\mathbf{xx}} = \langle \varphi(\mathbf{x}), \varphi(\mathbf{x}_i) \rangle$. Symbols $\mathcal{F}$ and $\mathcal{F}^{-1}$ respectively indicate the Discrete Fourier transform (DFT) and its inverse. The fraction operation is conducted element-wisely.

For simplicity, we use the symbol ^ to denote the DFT of a vector, e.g., $\hat{\mathbf{x}} = \mathcal{F}(\mathbf{x})$. If a sample $\mathbf{x}$ consists of C feature channels, we denote it as $\mathbf{x} = [\mathbf{x}^1; \ldots; \mathbf{x}^C]$. Thus, for a Gaussian kernel with the standard deviation σ, the kernelized correlation $\mathbf{k}^{\mathbf{xx}'}$ between two samples $\mathbf{x}$ and $\mathbf{x}'$ can be rewritten as

$$\mathbf{k}^{\mathbf{xx}'} = \exp\left(-\frac{1}{\sigma^2} \sum_{c}^{C} h(\mathbf{x}^c, \mathbf{x}'^c)\right) \tag{4}$$

subject to the channel-wise operation

$$h(\mathbf{x}^c, \mathbf{x}'^c) = \|\mathbf{x}^c\|^2 + \|\mathbf{x}'^c\|^2 - 2\mathcal{F}^{-1}(\hat{\mathbf{x}}^{c*} \odot \hat{\mathbf{x}}'^c), \tag{5}$$

where $\odot$ indicates the Hadamard product and superscript * denotes the complex conjugation.

Applying the classification function on an image patch $\mathbf{z}$ of $M \times N$ pixels, the evaluation on all circularly shifted versions of $\mathbf{z}$ can be interpreted as

$$\mathbf{f}(\mathbf{z}) = \mathcal{F}^{-1}(\hat{\mathbf{k}}^{\mathbf{xz}} \odot \hat{\boldsymbol{\alpha}}), \tag{6}$$

where $\mathbf{f}(\mathbf{z})$ consists of all the filter responses and has the same size as $\mathbf{z}$. Therefore, the most similar candidate $\mathbf{x}_z$ can be found at the peak value of $\mathbf{f}(\mathbf{z})$. Afterwards, it is utilized to update the appearance model $\mathbf{x}_{t-1}$ and coefficient vector $\boldsymbol{\alpha}_{t-1}$ from the frame $t-1$ by

$$\begin{aligned} \mathbf{x}_t &= (1-\beta)\mathbf{x}_{t-1} + \beta \mathbf{x}_z \\ \boldsymbol{\alpha}_t &= (1-\beta)\boldsymbol{\alpha}_{t-1} + \beta \boldsymbol{\alpha}_z \end{aligned} \tag{7}$$

with a small positive learning rate β. Term $\boldsymbol{\alpha}_z$ indicates the currently learned coefficient vector based on $\mathbf{x}_z$. As most of the calculations are done by element-wise operations in frequency domain, leveraging the power of Fast Fourier Transform (FFT), the processing speed of KCF is significantly improved. For further details about the implementation of the KCF tracker, we refer the reader to [23].

B. Channel-Wise Reliability Estimation by Kernelized Experts

According to equation (4), the term $\mathbf{k}^{\mathbf{xx}'}$ is calculated by accumulating the correlation of each feature channel, regarding them with equal contributions in measuring similarities between image samples $\mathbf{x}$ and $\mathbf{x}'$. Although this assumption works well in tracking objects in scenarios of good lighting conditions, it can be troublesome to deal with objects in low illuminated cases, particularly in the night. As each feature channel may represent one type of visual features, they can suffer from different fading effects caused by low illumination (Fig. 1 (b)). Thus, treating them equally can weaken the classification power of discriminative channels in the averaged filter response and make matching results vulnerable to noises from non-discriminative ones. Moreover, features from different channels may be extracted in various scales, a direct accumulation can also cause unbalanced weighting of different feature types.

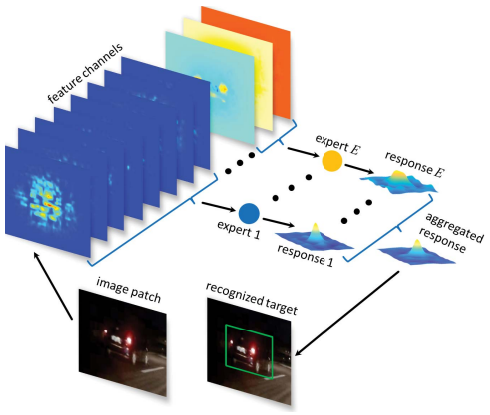


Fig. 2. The classifier is decomposed into a number E of kernelized experts, each focusing on a few feature channels. The filter response of each expert is weighted by its corresponding coefficient vector and aggregated into the final response map, with its peak to indicate the location of the most similar candidate of the target.

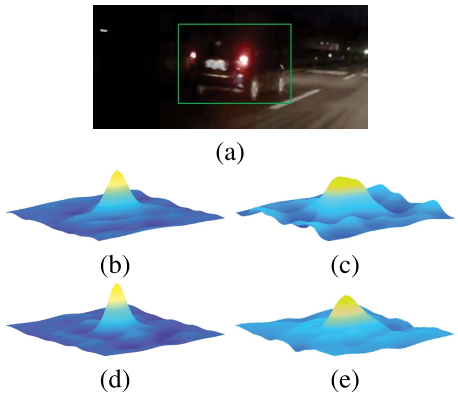


Fig. 3. In this example, the appearance model consists of two feature types: the FHOG features [40] and the color attributes [22], which are respectively aggregated into expert 1 and 2. The tracking target is indicated by a green box in image patch (a). Image (b) and (c) represent the weighted filter responses for each expert. As the color of this black vehicle suffers from great fading effect due to low illumination, the gradient features become more discriminative, which is implied by the sharp peak in response map of the first expert. As a result, the aggregated response of both experts in image (d) also exhibits a much sharper form than that of the normal KCF approach in image (e), which means the object becomes more distinguishable.

To solve these problems, we decompose the classifier $\mathbf{f}(\mathbf{z})$ into a number E of experts $\mathbf{f}_e(\mathbf{z})$ as

$$\mathbf{f}(\mathbf{z}) = \sum_{e}^{E} \mathbf{f}_e(\mathbf{z}), \tag{8}$$

each with a form of

$$\mathbf{f}_e(\mathbf{z}) = \mathcal{F}^{-1}(\hat{\mathbf{k}}_e^{\mathbf{xz}} \odot \hat{\boldsymbol{\alpha}}_e), \tag{9}$$

where the parameters of expert $\mathbf{f}_e(\mathbf{z})$ are indicated by the subscript $_e$. The idea is that we force each expert to focus only on a small number C_e of feature channels (Fig. 2), which are within the same type or scale, so that the corresponding kernelized correlation $\mathbf{k}_e^{\mathbf{xz}}$ can be restricted to these features, which can be expressed as

$$\mathbf{k}_e^{\mathbf{xz}} = \exp(-\frac{1}{\sigma^2}\sum_{c}^{C_e} h(\mathbf{x}^c, \mathbf{z}^c)) \quad \text{w.r.t.} \quad \sum_{e}^{E} C_e = C. \tag{10}$$

As it is further weighted by the coefficient vector $\boldsymbol{\alpha}_e$ in Fourier domain, the resulted filter response can be considered as a kind of reliability measurement, with great values assigned to discriminative experts (Fig. 3). In this way, feature channels can be fairly aggregated according to their classification powers. Hence, the remaining problem is to train a feasible coefficient vector for each of the experts.

Following Parsevaal's theorem and the definition of kernel functions, problem (2) can equivalently be solved in frequency domain as

$$\arg\min_{\hat{\mathbf{w}}} \|\mathrm{diag}(\hat{\mathbf{x}})\hat{\mathbf{w}} - \hat{\mathbf{y}}\|^2 + \lambda\|\hat{\mathbf{w}}\|^2 \tag{11}$$

$$= \arg\min_{\hat{\boldsymbol{\alpha}}} \|\mathrm{diag}(\hat{\mathbf{k}}^{\mathbf{xx}})\hat{\boldsymbol{\alpha}} - \hat{\mathbf{y}}\|^2 + \lambda\hat{\boldsymbol{\alpha}}^H \mathrm{diag}(\hat{\mathbf{k}}^{\mathbf{xx}})\hat{\boldsymbol{\alpha}}, \tag{12}$$

where superscript ^{H} denotes the Hermitian transpose and $\mathrm{diag}(\hat{\mathbf{k}}^{\mathbf{xx}})$ stands for a diagonal matrix with its diagonal formed by vector $\hat{\mathbf{k}}^{\mathbf{xx}}$. Here we use the equal sign to indicate the equivalence between these two optimization problems. Replacing the classifier with multiple experts, objective function (12) can be rephrased as

$$\arg\min_{\hat{\boldsymbol{\alpha}}_e} \|\sum_{e}^{E} \mathrm{diag}(\hat{\mathbf{k}}_e^{\mathbf{xx}})\hat{\boldsymbol{\alpha}}_e - \hat{\mathbf{y}}\|^2 + \lambda \sum_{e}^{E} \hat{\boldsymbol{\alpha}}_e^H \mathrm{diag}(\hat{\mathbf{k}}_e^{\mathbf{xx}})\hat{\boldsymbol{\alpha}}_e$$

$$= \arg\min_{\hat{\mathbf{a}}} \|\hat{\mathbf{K}}_1\hat{\mathbf{a}} - \hat{\mathbf{y}}\|^2 + \lambda\hat{\mathbf{a}}^H\hat{\mathbf{K}}_2\hat{\mathbf{a}} \tag{13}$$

with reshaped vector $\hat{\mathbf{a}} = [\hat{\boldsymbol{\alpha}}_1; \ldots; \hat{\boldsymbol{\alpha}}_E]$, matrices $\hat{\mathbf{K}}_1 = [\mathrm{diag}(\hat{\mathbf{k}}_1^{\mathbf{xx}}), \ldots, \mathrm{diag}(\hat{\mathbf{k}}_E^{\mathbf{xx}})]$ and $\hat{\mathbf{K}}_2 = \mathrm{diag}([\hat{\mathbf{k}}_1^{\mathbf{xx}}; \ldots; \hat{\mathbf{k}}_E^{\mathbf{xx}}])$. As vector $\mathbf{k}_e^{\mathbf{xx}}$ corresponds to the kernelized autocorrelation for specific feature channels of sample $\mathbf{x}$, according to the Wiener-Khinchin Theorem, its Fourier transform $\hat{\mathbf{k}}_e^{\mathbf{xx}}$, as well as matrices $\hat{\mathbf{K}}_1$ and $\hat{\mathbf{K}}_2$, only consists of non-negative real values. In this way, function (13) is convex due to its quadratic form of vector $\hat{\mathbf{a}}$. Hence, the optimum can be obtained at the point with zero gradient, which can be expressed as $\mathbf{A}\hat{\mathbf{a}} - \mathbf{b} = \mathbf{0}$ with respect to

$$\begin{cases} \mathbf{A} = \hat{\mathbf{K}}_1^H\hat{\mathbf{K}}_1 + \lambda\hat{\mathbf{K}}_2 \\ \mathbf{b} = \hat{\mathbf{K}}_1^H\hat{\mathbf{y}}. \end{cases} \tag{14}$$

Such optimization problem can be casted as solving the reformed equation set

$$\mathbf{A}\hat{\mathbf{a}} = \mathbf{b}. \tag{15}$$

Here we exploit the approach of Successive Over-Relaxation (SOR) to solve this expression of vector $\hat{\mathbf{a}}$. In this method, matrix $\mathbf{A}$ is decomposed in a form of $\mathbf{A} = \mathbf{D} + \mathbf{L} + \mathbf{U}$ with a diagonal matrix $\mathbf{D}$ as well as strictly lower and upper triangular matrices $\mathbf{L}$ and $\mathbf{U}$. The approximation errors of solution $\hat{\mathbf{a}}$ can be iteratively reduced by conducting following operations

$$(\mathbf{D} + \omega\mathbf{L})\hat{\mathbf{a}}^{(j+1)} = \omega\mathbf{b} - (\omega\mathbf{U} + (\omega - 1)\mathbf{D})\hat{\mathbf{a}}^{(j)} \tag{16}$$

in each iteration j with a constant relaxation factor ω. This procedure stops when either the maximum iteration number

$N_{\mathcal{J}}$ is reached or the approximation error converges to a predefined span of $\|\mathbf{A}\hat{\mathbf{a}}^{(j+1)} - \mathbf{b}\|^2 \leq \varepsilon_a$. As both matrices $\hat{\mathbf{K}}_1$ and $\hat{\mathbf{K}}_2$ are block-wise diagonal, the matrix $\mathbf{A}$ should also share a similar sparse structure. Thus, calculations are only performed on a few matrix elements. Leveraging this kind of sparsity in combination with forward substitution of intermediate solutions, the equation set (15) can be solved efficiently.

C. Temporal Reliability Estimation for Training Set

In the KCF approach, depending on the selected value of the learning rate β, the updated tracker model by equation (7) can heavily rely on the quality of sample extracted from the current image. As the behavior of an object is unpredictable, the target appearance can encounter abrupt changes, e.g., caused by occlusions, false classifications or unclear vision conditions such as bad weather (Fig. 1 (c)-(d)). Even if these appearance changes may only happen in a very short length of time, the learned model could become inconsistent with the real target if these corrupted image samples are added into the training dataset. This could further lead to drifted object predictions and even tracking failures.

Additionally, the fixed learning rate β is not sufficient to deal with the trade-off between samples from different frames. For instance, in cases of tracking static objects, the old samples should not be quickly removed from the training set, as they can also provide precious information in reidentifying the target, which is out of occlusions. However, to track objects with rapid deformation or rotation, the recent samples should exert more influence on training the model so that it can fit the current object appearance. Therefore, a preferred solution is the dynamic weighting of historical samples.

In an effort to tackle these problems, we apply a joint learning approach, which can train the tracker model and evaluate the reliability of samples concurrently. For simplicity yet without loss of generality, we discuss the case of only one expert, so that $\hat{\boldsymbol{\alpha}} = \hat{\boldsymbol{\alpha}}_e$ with $e = 1$. Here we rewrite the problem (12) as minimizing the loss function

$$\mathcal{J}(\hat{\boldsymbol{\alpha}}, \mathbf{x}_t) = \|\mathrm{diag}(\hat{\mathbf{k}}^{\mathbf{x}_t\mathbf{x}_t})\hat{\boldsymbol{\alpha}} - \hat{\mathbf{y}}\|^2 + \lambda\hat{\boldsymbol{\alpha}}^H\mathrm{diag}(\hat{\mathbf{k}}^{\mathbf{x}_t\mathbf{x}_t})\hat{\boldsymbol{\alpha}} \tag{17}$$

with respect to the vector $\hat{\boldsymbol{\alpha}}$ and image sample $\mathbf{x}_t$ at frame t. By introducing weights for samples in an interval of T frames, the minimization of function (17) can be extended to a joint optimization problem of

$$\arg\min_{\boldsymbol{\theta},\hat{\boldsymbol{\alpha}}} \mathcal{L}(\boldsymbol{\theta}, \hat{\boldsymbol{\alpha}}) = \arg\min_{\boldsymbol{\theta},\hat{\boldsymbol{\alpha}}} \sum_{t=1}^{T} \theta_t \mathcal{J}(\hat{\boldsymbol{\alpha}}, \mathbf{x}_t) + \sum_{t=1}^{T} \frac{\theta_t^2}{p_t} \tag{18}$$

subject to

$$\sum_{t=1}^{T} \theta_t = 1 \wedge \theta_t \geq 0, \quad \forall t \in [1, T], \tag{19}$$

where $\mathcal{L}$ is a joint loss function in terms of the coefficient vector $\hat{\boldsymbol{\alpha}}$ and weight vector $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_T]$. The term $\mathbf{p} = [p_1, \ldots, p_T]$ is a vector of positive priors to regularize the distribution of sample weights. Constraint (19) implies that samples should be assigned with non-negative weights, with their sum normalized to 1.

According to the memory model of human brain [41], the recently captured objects should be preserved, as they with high probability will appear again, while the old ones without presence are not so important and can be forgotten gradually. In the light of this model, the prior p_t for a historical sample $\mathbf{x}_t$ can be calculated in a similar way. Here we deploy the forgetting curve [42] to initialize a sample prior by

$$p_t = \mu \exp(-\frac{T - t}{Th}) \tag{20}$$

with a positive parameter h to control the strength of memory. The multiplier μ is a constant value and guarantees that the prior sum is normalized with $\sum_t^T p_t = 1$. The exponential function presents the decline of memory retention over time. Hence, the strongest retention is obtained at current time T.

Introducing equation (20) into loss function $\mathcal{L}$, we employ the approach of Alternate Convex Search (ACS) to solve the joint optimization problem (18), which can be demonstrated as biconvex [29]. In this method, we iteratively solve two subproblems with fixed values of either $\boldsymbol{\theta}$ or $\hat{\boldsymbol{\alpha}}$. Thus, in each iteration, we proceed with the following two steps:

1) *Step of Updating Coefficient Vector $\hat{\boldsymbol{\alpha}}$:* The weight vector $\boldsymbol{\theta}$ is initialized with an equal distribution at the very beginning. For each iteration l, since sample weights are fixed in the first updating step, the subproblem can be described as minimizing the loss

$$\mathcal{L}(\hat{\boldsymbol{\alpha}}^{(l)}) = \arg\min_{\hat{\boldsymbol{\alpha}}^{(l)}} \sum_{t=1}^{T} \theta_t \mathcal{J}(\hat{\boldsymbol{\alpha}}^{(l)}, \mathbf{x}_t) \tag{21}$$

with respect to the coefficient vector $\hat{\boldsymbol{\alpha}}^{(l)}$. As this simplification can be considered as a sum of weighted loss functions $\mathcal{J}(\hat{\boldsymbol{\alpha}}^{(l)}, \mathbf{x}_t)$ for each sample $\mathbf{x}_t$, it can be solved by the same approaches (14)-(15) yet with minor modifications of

$$\begin{cases} \mathbf{A} = \sum_{t=1}^{T} \theta_t (\hat{\mathbf{K}}_{1,t}^H \hat{\mathbf{K}}_{1,t} + \lambda \hat{\mathbf{K}}_{2,t}) \\ \mathbf{b} = \sum_{t=1}^{T} \theta_t \hat{\mathbf{K}}_{1,t}^H \hat{\mathbf{y}}, \end{cases} \tag{22}$$

where the subscript $_t$ indicates that matrices $\hat{\mathbf{K}}_{1,t}$ and $\hat{\mathbf{K}}_{2,t}$ are calculated by the image sample $\mathbf{x}_t$ at frame t. The obtained solution $\hat{\boldsymbol{\alpha}}^{(l)}$ is then passed on to the next step of updating sample weights $\boldsymbol{\theta}^{(l)}$.

2) *Step of Updating Weight Vector $\boldsymbol{\theta}$:* In this second update step, for a given vector $\hat{\boldsymbol{\alpha}} = \hat{\boldsymbol{\alpha}}^{(l)}$, the output of function $\mathcal{J}(\hat{\boldsymbol{\alpha}}^{(l)}, \mathbf{x}_t)$ for each individual sample $\mathbf{x}_t$ is constant. Therefore, the optimization problem of (18)-(19) is only dependent on the weight vector $\boldsymbol{\theta}^{(l)}$, so that the parameters of joint loss $\mathcal{L}(\boldsymbol{\theta}^{(l)}, \hat{\boldsymbol{\alpha}}^{(l)})$ can be reduced to $\mathcal{L}(\boldsymbol{\theta}^{(l)})$. As this kind of simplified function is expressed in a quadratic form of vector $\boldsymbol{\theta}^{(l)}$ subject to constrains (19), it can be solved by the Quadratic Programming approach. The solution $\boldsymbol{\theta}^{(l)}$ is reused to update the coefficient vector $\hat{\boldsymbol{\alpha}}^{(l+1)}$ in the next iteration $l + 1$.

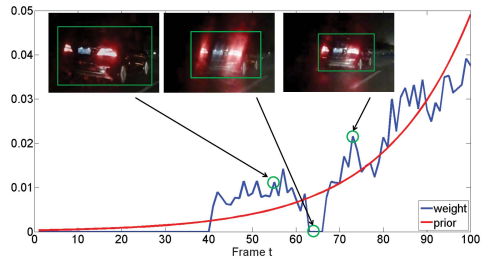


Fig. 4. In this example, a black car is tracked in a wet night. Here we keep $T = 60$ historical examples to train the classifier. The sample weights and their priors are respectively denoted in blue and red. As samples at frames 56 and 73 are extracted with a clear vision, they are assigned with great weights. For the sample at frame 64, as the vision is contaminated by the raindrops, it is inconsistent with the real target appearance. Therefore, its weight value is low.

This loop terminates if either the maximum iteration $N_{\mathcal{L}}$ is reached or the joint loss $\mathcal{L}$ converges, interpreted as

$$\|\mathcal{L}(\boldsymbol{\theta}^{(l+1)}, \hat{\boldsymbol{\alpha}}^{(l+1)}) - \mathcal{L}(\boldsymbol{\theta}^{(l)}, \hat{\boldsymbol{\alpha}}^{(l)})\|^2 \leq \varepsilon_{\mathcal{L}}, \tag{23}$$

where $\varepsilon_{\mathcal{L}}$ indicates the predefined upper limit of the approximation error. After the training procedure, we sort the samples according to their weights in an ascending order. If the amount of training set exceeds its upper limit T, we eliminate the samples with the smallest weights, so that both computational burden and memory consumption are limited. Analogously, the current appearance model $\mathbf{x}_z$ is also updated by the aggregation of weighted samples, interpreted as

$$\mathbf{x}_z = \sum_{t=1}^{T} \theta_t \mathbf{x}_t. \tag{24}$$

The whole procedure of this approach is illustrated by an example in Fig 4. Here we keep $T = 60$ historical examples. The calculated weights present a similar trend with the prior except at the frames 63-65. As these frames exhibit unclear vision, which is caused by raindrops, their samples are assigned with almost zero weights. After the image quality is restored by cleaning the raindrops, high weight values are assigned to samples again, e.g., at frame 73. In this way, only the most reliable samples are utilized to train the classifier, which improves the robustness of tracking, especially under bad weather conditions.

IV. Experiments and Results

To evaluate the proposed method, we test it on the night traffic dataset provided by [35], which is recorded by a wide range CCD camera mounted directly behind the windshield of a moving vehicle during the night. This dataset consists of 119 videos with an entire length of about 3.5 hours and a resolution of 856 × 480 pixels. The sequences are recorded with different time slots (e.g., from nightfall to morning), road conditions (e.g., from highways to crowded streets) and weather factors (e.g., from clear nights to the ones with raindrops). Besides, lots of vehicles with a rich diversity in color, size, class and behavior can be seen, which makes it possible for us to evaluate the tracker in various scenarios.

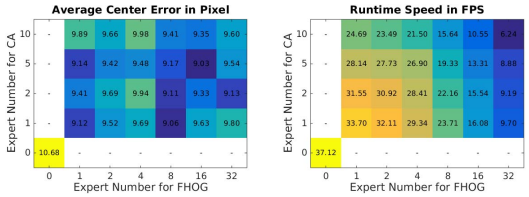


Fig. 5. Heat maps (a) and (b) respectively show the tracking accuracy and runtime speed of our approach in dependence of expert numbers. The first metric is defined by the average center distance between estimated targets and their groundtruths while the second one is given in frames per second (FPS). As our tracker consists of 32 FHOG feature channels [40] and 10 color attributes (CA) [22], we only choose the expert number for each feature type so that the channels can be divided equally. All values are calculated with $N_{\mathcal{L}} = 4$, which is proven to be sufficient in Section IV-B. The value at zero coordinates corresponds to the original KCF tracker.

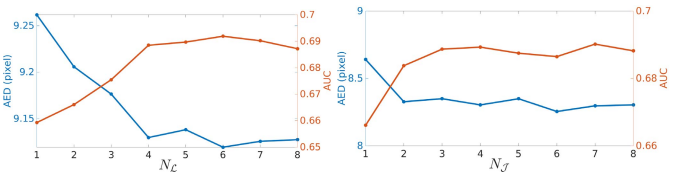


Fig. 6. Plots (a) and (b) respectively show the test results on different iteration numbers of parameter $N_{\mathcal{L}}$ and $N_{\mathcal{J}}$. The tracking precision is presented by two metrics: the AED and the AUC curves, which are denoted in blue and orange respectively.

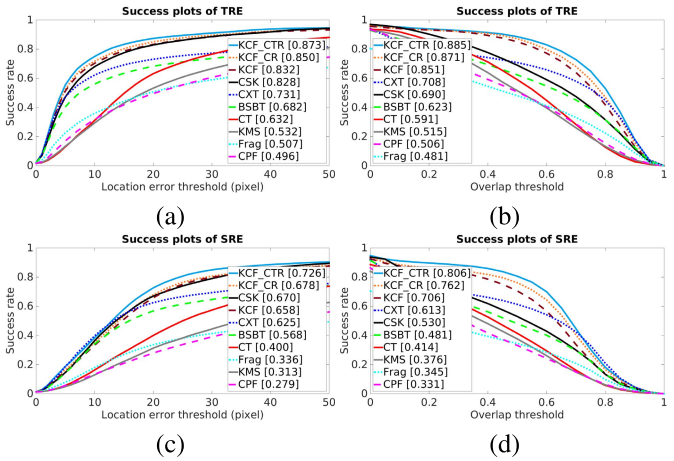


Fig. 7. Performance evaluation of different trackers based on metrics of TRE and SRE with each presented in one row. The success rate in the first column is calculated in terms of location error while in the second column based on the overlap ratio between the predicted target and its groundtruth. Additionally, trackers are ranked according to their precision values at the threshold of a location error of 20 pixels and an overlap ratio of 50%.

TABLE I

Evaluation of Runtime Performance on Each Tracker in Frames Per Second (FPS)

Method	CT	BSBT	CXT	CSK	CPF	Frag	KMS	KCF	KCF_CR	KCF_CTR
FPS	28.0	1.6	9.1	109.3	18.7	2.0	451.9	37.1	33.7	21.4

A. Parameter Settings

As different procedures of reliability estimation are incorporated in our approach, for a better understanding of their performance, we prepare three versions of KCF-based trackers. The first one is the original KCF tracker, whose code is

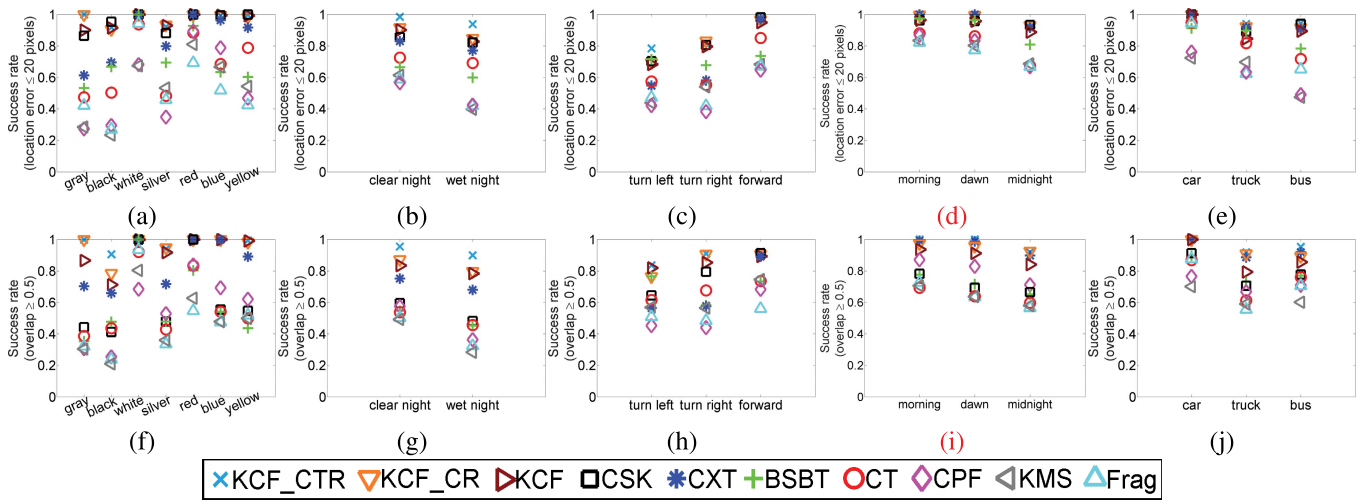


Fig. 8. Evaluation of nighttime tracking performance in attributes of color, weather, behavior, time and vehicle class. The first row presents the precision of each tracker in terms of a location error threshold of 20 pixels. The second row indicates the tracker precision with more than 50% of overlap ratio between predicted target and its groundtruth.

provided by [23]. The next one is denoted as KCF_CR, which is a combination with channel-wise reliability estimation by aggregated experts. The last tracker is an integration of both channel-wise and temporal reliability estimation, denoted as KCF_CTR. The appearance model is built by two different feature types: FHOG features [40] and color attributes (CA) [22], each with a cell size of 4 pixels and integrated into one expert. We utilize two different experts because the deployment of more experts only yields minor improvement of the tracking accuracy but significantly decreases the runtime speed, as illustrated in Fig. 5. The constant learning rate β for the first two trackers is equal to 0.025. All image patches utilized for training are rescaled to a unified square of 128 pixels. The relaxation factor is empirically set to $\omega = 1.1$. For the temporal reliability estimation, we keep at most $T = 60$ historical samples. The memory strength is set to $h = 0.5$. The upper bound of approximation error ε_α and $\varepsilon_\mathcal{L}$ are equally set to 10^{-4}. To track objects in varied sizes, we deploy five different scales with a scaling step of 1.1. All other parameters of the KCF tracker follow their default settings in [23]. The tests are performed on a laptop platform with an Intel i7-3740QM CPU of 2.7GHz and a memory of 8GB. The proposed approach is implemented in C++ and runs in a single thread with an average memory usage of less than 50MB.

B. Exploration on Iteration Number

As both the channel-wise and the temporal reliability estimation are iterative procedures, in the first experiment we would like to show the influence of the iteration numbers $N_\mathcal{L}$ and $N_\mathcal{J}$ on tracking precision, which is measured by two metrics: the Average Euclidean Distance (AED) and the Area Under Curve (AUC). The first one calculates the distance between the centers of tracked objects and their groundtruths. The upper limit is set to 20 pixels, so that the influence of outliers caused by loss of targets can be reduced. The second one accumulates the areas under ROC-curves. For each of those two parameters we conduct tests within a small range of $1 \leq N_\mathcal{L}, N_\mathcal{J} \leq 8$, so that the tolerance area of approximation error is not reached.

The test results with respect to the iteration numbers are plotted in Fig. 6. Obviously, for the channel-wise reliability estimation, the AED curve decreases greatly in the first half range while it settles down afterwards. Similarly, the increasing trend of its AUC curve also slows down after the 4th iteration. As more iterations are not beneficial for the tracking precision and only increase the computational burden, the value of $N_\mathcal{L}$ is set to 4 in further experiments. In comparison, the AED curve for temporal reliability estimation already converges after the second iteration while its AUC value only becomes stable when reaching the point of $N_\mathcal{J} = 3$. Therefore, we set the parameter $N_\mathcal{J}$ to 3.

C. Comparison With State-of-the-Art

In the next experiment, we compare the performance of our proposed approach with that of seven other state-of-the-art trackers, i.e., CPF [11], Frag [43], KMS [10], CT [22], BSBT [44], CXT [12], CSK [45]. The parameters of those trackers are set according to their original papers. Here we follow the one-pass evaluation (OPE) protocol [7] which initializes the tracker at the first frame with a given bounding box and thereupon records its predicted target location and size over the test sequence. To explore the performance of our proposed tracker in general cases, we employ two more metrics for precision measurement: the temporal robustness evaluation (TRE) and the spatial robustness evaluation (SRE). In the first metric, the tracking is triggered at equally distributed frames in the temporal domain to simulate sequences with varied lengths. In the second metric, the tracker is initialized at the first frame yet with shifted and scaled bounding boxes, which can be considered as resulted from imperfect object detectors. The tracking precision is measured by the success rate in terms of location error and overlap ratio between the predicted target and its groundtruth with varied upper limits.

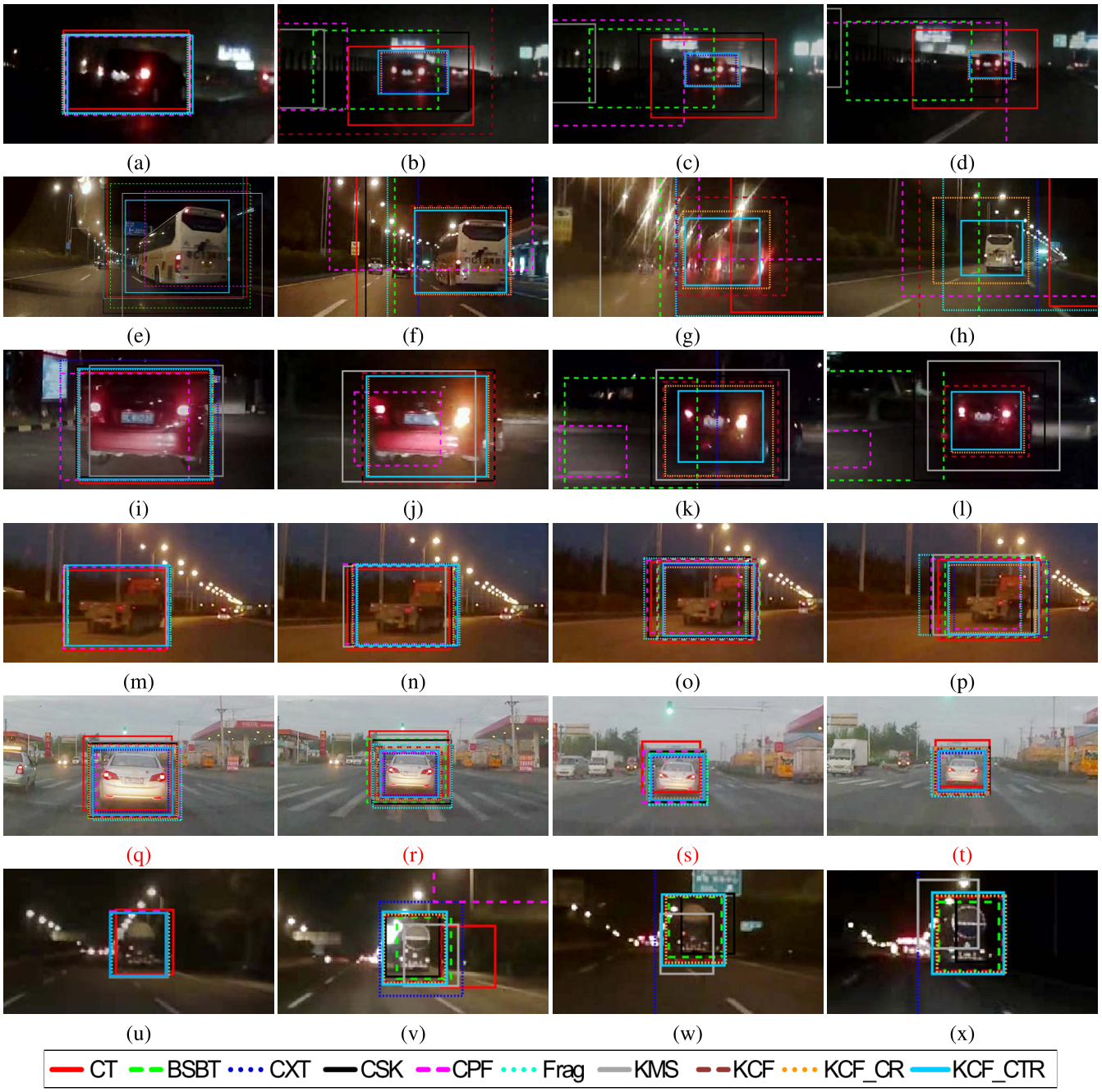


Fig. 9. Examples of nighttime vehicle tracking in attributes of color, weather, behavior, time and vehicle class, each displayed in one row respectively.

For a fair comparison, we follow [7] to rank the trackers at the threshold of a location error of 20 pixels and an overlap of 50%, illustrated in Fig. 7. In the ranking, it is clear that the KCF-based trackers already outperform most other state-of-the-art approaches. Other well-performing ones are only the CSK and CXT trackers. The first one also shares a correlation framework but extracts features from raw pixels while the second one employs additional feature points to support the location inference of tracked targets. Although both of them achieve comparable location precision with that of the naive KCF approach, as their utilized features are more prone to noises in low exposed images, the precision of their estimated object sizes is not high (Fig. 7 (b) and (d)). Compared with the naive KCF tracker, the precision is boosted by 2 $\sim$ 6% in the KCF_CR approach, due to a better weighting of different feature types. If it is integrated with temporal reliability estimation, i.e., the KCF_CTR tracker, its precision can be further improved by about 2 $\sim$ 5%.

In Table I we report the runtime performance of the tested trackers. On the top of this list is the KMS approach with a speed of 451.9 frames per second (fps), owing to its light-weighted models and features, yet at the cost of vulnerability to low exposed images. By integration of channel-wise and temporal reliability estimation, although our

method KCF_CTR is about 15 fps slower than the naive KCF approach, it shows strong robustness in badly illuminated cases and still runs at a speed of 21.4 fps, which can fulfill most real time requirements.

D. Evaluation Based on Attributes

For a systematical analysis of the influence on tracker performance by different factors, we conduct 5 additional experiments according to the attributes of color, weather, behavior, time and vehicle class. For a fair comparison, we only change one attribute in each experiment setting while the other ones are kept the same or equivalent. For each test scenario, sequences are selected with a length of more than 200 frames and contain up to 1000 frames. The number of tracked targets is in the range of 30 to 50. The precision is also measured by the success rate at the threshold of a location error of 20 pixels and an overlap ratio of 50%. Detailed results are plotted in Fig. 8. A comprehensive description is as follows.

1) *Color:* The low illumination in the night leads to the fact that bright colors are more recognizable than the dark ones. Thus, the average tracking precision on white or red vehicles is much higher than that with gray or black colors, as presented in Fig. 8 (a) and (f). Among the trackers, the approaches of KCF_CTR and KCF_CR always exhibit an outstanding performance, especially in tracking objects with strongly faded colors (Fig. 9 (a)-(d)). This demonstrates the advantage of feature weighting in building efficient tracker models.
2) *Weather:* Here we discuss about two cases: clear and wet night. In the second case, the vision condition is interfered by raindrops, resulting in contaminated image areas (Fig. 9 (g)). Although the corrupted object appearance is cumbersome for most of the trackers, by removing the outliers through temporal reliability estimation, our approach KCF_CTR can still successfully track the targets, which is illustrated in Fig. 9 (e)-(h).
3) *Behavior:* As presented in Fig. 8 (c) and (h), tracking of turning vehicles is much more difficult than that of forward driving ones. The main problem is the turning signal, which scatters into bright image areas in dark nights (Fig. 9 (j)). As the signal lamp blinks at a specific frequency, this scattering effect only appears at non-consecutive frames, resulting in frequent change of object appearance. Nevertheless, it can be handled by carefully building models according to the channel-wise and temporal reliabilities. Hence, both KCF_CTR and KCF_CR perform well in this case.
4) *Time:* Here we focus on three scenarios: morning, dawn and midnight. The first case can be considered as regular daily test (Fig. 9 (q)-(t)). As the illumination conditions are relatively good, all trackers achieve their highest accuracy in this scenario, with our tracker ranked at the top. In the second case, the captured images are a little bit dim. However, the contour, color and shape of an object are almost fully visible (Fig. 9 (m)-(p)). Thus, the average tracking precision is still higher than that in midnight cases, where the visual feature is strongly deteriorated by darkness (Fig. 9 (a)-(d)).
5) *Vehicle Class:* Test results in Fig. 8 (e) and (j) have shown that tracking trucks and buses is more difficult than cars in the night. An explanation is that buses are usually painted with various patterns and colors, such as ads, on different sides. This property leads to a significant appearance change especially in passing-by situations. For trucks, the trouble comes from its rear part. If the distance to a tracked truck varies over time, its reflection stripes may not always be reached by the headlight of the ego-vehicle, resulting in changed patterns (Fig. 9 (u)-(x)). Although both cases are difficult for most trackers, our tracker still exhibits strong robustness.

V. Conclusion

In this paper we present a novel method to track vehicles under low illumination conditions, e.g., at night. The tracker is tailored from the baseline framework of the kernelized correlation filter. By exploring the reliability in feature levels with the help of aggregated experts we are able to extract the most discriminative visual features. Additionally, with the help of picking reliable training samples in the time domain by integrating the memory model of human brain, we are capable to deal with tracking under deteriorated vision conditions. These two steps are successfully incorporated in a learning framework using the approach of Alternate Convex Search. Extensive experimental results demonstrate the performance of our tracker, improving state of the art. Furthermore, we present a systematical analysis on tracking vehicles at night with various challenging factors such as color, weather, time, behavior and vehicle class. Leveraging an elaborate design, especially by transferring most of the computation to the frequency domain, a real-time performance is also available with our method. Although in this paper, we focus on vehicle tracking under night conditions, as our approach optimizes the classifier in feature levels, it can also be extended to deal with tracking under other difficult situations, such as in whitened scenarios like foggy or snowy days. Furthermore, as our tracker is built independently from object classes or numbers, it is possible to be integrated into multi-object tracking approaches. Exploring these points is part of our future works.

References

[1] *Injuries and Violence: The Facts 2014*, World Health Org., Geneva, Switzerland, 2014.
[2] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.
[3] 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.
[4] N. Wang, J. Shi, D.-Y. Yeung, and J. Jia, "Understanding and diagnosing visual tracking systems," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3101–3109.
[5] M. Kristan *et al.*, "The thermal infrared visual object tracking VOT-TIR2016 challenge results," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 824–849.
[6] Y.-L. Chen, Y.-H. Chen, C.-J. Chen, and B.-F. Wu, "Nighttime vehicle detection for driver assistance and autonomous vehicles," in *Proc. IEEE Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2006, pp. 687–690.

[7] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2411–2418.

[8] A. Roy, X. Zhang, N. Wolleb, C. P. Quintero, and M. Jägersand, "Tracking benchmark and evaluation for manipulation tasks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 2448–2453.

[9] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5630–5644, Dec. 2015.

[10] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

[11] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2002, pp. 661–675.

[12] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1177–1184.

[13] G. Nebehay and R. Pflugfelder, "Clustering of static-adaptive correspondences for deformable object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2784–2791.

[14] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 983–990.

[15] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2006, pp. 1–7.

[16] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof, "On-line random forests," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV)*, Sep./Oct. 2009, pp. 1393–1400.

[17] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof, "PROST: Parallel robust online simple tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 723–730.

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

[19] 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. (CVPR)*, Jun. 2010, pp. 2544–2550.

[20] V. N. Boddeti, T. Kanade, and B. V. K. V. Kumar, "Correlation filters for object alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2291–2298.

[21] H. K. Galoogahi, T. Sim, and S. Lucey, "Multi-channel correlation filters," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 3072–3079.

[22] 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. (CVPR)*, Jun. 2014, pp. 1090–1097.

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

[24] H. K. Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4630–4638.

[25] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4310–4318.

[26] A. Lukežič, T. Vojíř, L. Čehovin, J. Matas, and M. Kristan. (Nov. 2016). "Discriminative correlation filter with channel and spatial reliability." [Online]. Available: https://arxiv.org/abs/1611.08461

[27] W. Zuo, X. Wu, L. Lin, L. Zhang, and M.-H. Yang. (Jan. 2016). "Learning support correlation filters for visual tracking." [Online]. Available: https://arxiv.org/abs/1601.06032

[28] 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. (CVPR)*, Jun. 2015, pp. 749–758.

[29] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1430–1438.

[30] Y. Li, Y. Zhang, Y. Xu, J. Wang, and Z. Miao, "Robust scale adaptive kernel correlation filter tracker with hierarchical convolutional features," *IEEE Signal Process. Lett.*, vol. 23, no. 8, pp. 1136–1140, Aug. 2016.

[31] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 472–488.

[32] K. Robert, "Night-time traffic surveillance: A robust framework for multi-vehicle detection, classification and tracking," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveill.*, Sep. 2009, pp. 1–6.

[33] Q. Zou, H. Ling, S. Luo, Y. Huang, and M. Tian, "Robust nighttime vehicle detection by tracking and grouping headlights," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2838–2849, Oct. 2015.

[34] H. Hajimolahoseini, R. Amirfattahi, and H. Soltanian-Zadeh, "Robust vehicle tracking algorithm for nighttime videos captured by fixed cameras in highly reflective environments," *IET Comput. Vis.*, vol. 8, no. 6, pp. 535–544, 2014.

[35] L. Chen, X. Hu, T. Xu, H. Kuang, and Q. Li, "Turn signal detection during nighttime by CNN detector and perceptual hashing tracking," *IEEE Trans. Intell. Transp. Syst.*, to be published, doi: 10.1109/TITS.2017.2683641.

[36] R. O'Malley, E. Jones, and M. Glavin, "Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 453–462, Jun. 2010.

[37] S.-Y. Kim *et al.*, "Front and rear vehicle detection and tracking in the day and night times using vision and sonar sensor fusion," in *Proc. World Congr. Intell. Transp. Syst.*, Aug. 2005, pp. 2173–2178.

[38] C. Fries and H.-J. Wuensche, "Autonomous convoy driving by night: The vehicle tracking system," in *Proc. IEEE Int. Conf. Technol. Pract. Robot Appl. (TePRA)*, May 2015, pp. 1–6.

[39] R. Rifkin *et al.*, "Regularized least-squares classification," in *NATO Science Series, III: Computer and Systems Sciences*. Amsterdam, The Netherlands: IOS Press, 2003.

[40] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Intell. Transp. Syst.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

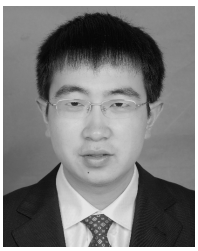
[41] R. C. Atkinson and R. M. Shiffrin, "Human memory: A proposed system and its control processes," in *Psychology of Learning and Motivation*, vol. 2. New York, NY, USA: Academic, 1968, pp. 89–195, doi: 10.1016/S0079-7421(08)60422-3.

[42] H. Ebbinghaus, "Memory: A contribution to experimental psychology," *Ann. Neurosci.*, vol. 20, no. 4, pp. 155–156, 2013.

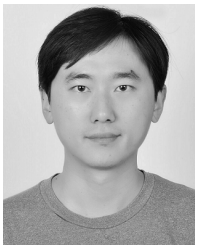
[43] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2006, pp. 798–805.

[44] S. Stalder, H. Grabner, and L. van Gool, "Beyond semi-supervised tracking: Tracking should be as simple as detection, but not simpler than recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Sep./Oct. 2009, pp. 1409–1416.

[45] 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. (ECCV)*, 2012, pp. 702–715.



Wei Tian received the B.Sc. degree in mechatronics engineering from Tongji University, Shanghai, China, in 2010, and the M.Sc. degree from the Department of Electrical Engineering and Information Technology, KIT, Karlsruhe, Germany, in 2013. He is currently pursuing the Ph.D. degree with the Institute of Measurement and Control Systems, KIT. He is interested in research areas of robust object detection and tracking.



Long Chen received the B.Sc. degree in communication engineering and the Ph.D. degree in signal and information processing from Wuhan University, Wuhan, China, in 2007 and 2013, respectively. From 2010 to 2012, he was a co-trained Ph.D. Student with the National University of Singapore. From 2008 to 2013, he was in charge of environmental perception system for autonomous vehicle SmartV-II with the Intelligent Vehicle Group, Wuhan University. He is currently an Associate Professor with the School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China. His areas of interest include perception system of intelligent vehicle.

Ke Zou is currently pursuing the master's degree with Sun Yat-sen University, where she is majoring in software engineering in China. Her interest in computer vision began in 2015 when she became a member of the CPS Laboratory under the instruction of L. Chen. She mainly focuses on target tracking.

Martin Lauer received the Diploma degree in computer science from Karlsruhe University and the Ph.D. degree in computer science from Osnabrück University in 2004. He was a Post-Doctoral Researcher with Osnabrück University in the areas of machine learning and autonomous robots. Since 2008, he has been leading a Research Group with the Karlsruhe Institute of Technology. His main research interests are in the areas of machine vision, autonomous vehicles, and machine learning.