A Scale Adaptive Kernel Correlation Filter Tracker with Feature Integration

Yang Li and Jianke Zhu(✉)

College of Computer Science, Zhejiang University Zhejiang, Hangzhou, China
jkzhu@zju.edu.cn

Abstract. Although the correlation filter-based trackers achieve the competitive results both on accuracy and robustness, there is still a need to improve the overall tracking capability. In this paper, we presented a very appealing tracker based on the correlation filter framework. To tackle the problem of the fixed template size in kernel correlation filter tracker, we suggest an effective scale adaptive scheme. Moreover, the powerful features including HoG and color-naming are integrated together to further boost the overall tracking performance. The extensive empirical evaluations on the benchmark videos and VOT 2014 dataset demonstrate that the proposed tracker is very promising for the various challenging scenarios. Our method successfully tracked the targets in about 72% videos and outperformed the state-of-the-art trackers on the benchmark dataset with 51 sequences.

Keywords: Visual Tracking · Correlation Filter · Kernel Learning

1 Introduction

Visual tracking is one of the fundamental research problem in computer vision community for its various applications in video surveillance, robotics, human computer interaction and driverless vehicle. Although great progress has been made in the past decade, the model-free tracking is still a tough problem due to illumination changes, geometric deformations, partial occlusions, fast motions and background clutters.

Recently, correlation filter is introduced into visual community, which has already been applied in many applications [2] [10] [13] [27]. As described in Convolution Theorem, the correlation in time domain corresponds to an element-wise multiplication in Fourier domain. Thus, the intrinsic idea of correlation filter is that the correlation can be calculated in Fourier domain in order to avoid the time-consuming convolution operation. Meanwhile, the correlation filter is treated as similarity measure between the two signals in signal processing, which gives a reliable distance metric and explains the reason of the promising performance achieved by the previous approaches. Bolme et al. [7] and Henriques et al. [13] introduce the correlation filter into the tracking application. Although achieved the appealing results both in accuracy and robustness, these correlation

L. Agapito et al. (Eds.): ECCV 2014 Workshops, Part II, LNCS 8926, pp. 254–265, 2015.
DOI: 10.1007/978-3-319-16181-5_18

filter-based trackers employ the template with the fixed size, which is not able to handle the scale changes of a target.

In this paper, we propose a novel scale adaptive kernelized correlation filter tracker with multiple feature integration. The proposed approach overcomes the limitations of the conventional correlation filter trackers by a multiple scales searching strategy. To solve the scale change issue in object tracking, we sample the target with different scales, and resize the samples into a fixed size to compare with the leant model at each frame. Meanwhile, we adopt a multiple feature integration scheme, which employs the raw pixel, Histogram of Gradient [9] and color-naming [32] to further enhance the proposed tracker for dealing with the more challenge scenarios. Our experimental evaluation demonstrates that the proposed scale adaptive and multiple feature integration method achieves a significant performance gain (over 10%) comparing the state-of-the-art approach. Moreover, our method successfully tracks the targets in almost 72% sequences in the benchmark [33] with 51 videos in total.

The main contributions of this paper can be summarized as follows. Firstly, we extend the correlation filter-based tracker with the capability of handling scale changes, which obtains an impressive performance gain in accuracy. Secondly, we conduct the extensive experiments to compare the previous studies of the correlation filter-based trackers [14] [4] [12] with our proposed method that includes multiple features integration, scale adaptive scheme and a full system. These experiments reveals the underline clues on the importance of the different components for a modern tracking-by-detection tracker. Finally, the proposed tracker achieved a very appealing performance both in accuracy and robustness against the state-of-the-art trackers.

2 Related Works

Tracking-by-detection trackers [11] [1] [16] [34] are very popular due to its high performance and efficiency. As these methods usually employ the binary classifier to distinguish the tracked object from the background, which are usually denoted as the discriminative methods. Struck [11] is one the most representative discriminative trackers, which employs the structured Support Vector Machine(SVM) to directly link the target's location space with the training samples. It achieves the appealing result in the recent benchmark [33]. TLD [16] exploits a set of structural constraints with a sampling strategy using boosting classifier. The re-detection function makes the TLD method more robust in the challenge videos. Inspired by the compressive sensing techniques, Zhang et al. [34] train a Naive Bayes classifier with the compressive features projected from the original space. MIL [1] explores the idea of a bag of positive samples with a boosting variant algorithm to construct the tracker. Meanwhile, generative model-based trackers [22] [15] [21][29] [3] [30] [24] aim to build the metric model to search the most similar patches for the tracked object. SCM [36] combines the discriminative classifier and generative model to achieve the high accuracy and robustness. However, it involves with the heavy computational cost, which hinders its capability on real-time applications. Additionally, some trackers [5] [35] employ the

structure information in the scene to enhance the tracking performance while others [31] exploits the deep learning techniques in the object tracking task.

Our proposed approach is closely related to the correlation filter-based trackers [14] [4] [12] [7] [6], which adopt the correlation filter in traditional signal processing technique into the tracking applications. CSK [12] is proposed to explore the structure of the circulant patch to enhance the classifier by the augmentation of negative samples, which employs the kernel correlation filter to achieve the high efficiency. Based on CSK [12], KCF [14] adopts the HoG feature [9] instead of raw pixel to improve both the accuracy and robustness of the tracker. To further boost the performance of CSK tracker, Danelljan et al. [4] adopt the color-naming feature into the object tracking task, which is a powerful feature for the color objects [17] [19] [18]. Meanwhile, MOSSE [7] formulates the problem in the view of learning a filter .

3 The Tracker

In this section, we firstly review the kernelized correlation filter (KCF) tracker [14], and then introduce the powerful features used in our approach. Moreover, a scale adaptive scheme is presented to improve the correlation filter-based trackers.

3.1 The KCF Tracker

Our approach is built on KCF tracker [14], which achieves very impressive results on Visual Tracker Benchmark [33]. Although the idea of KCF tracker is very simple, it achieves the fastest and highest performance among the recent top-performing trackers. The key of KCF tracker is that the augmentation of negative samples are employed to enhance the discriminative ability of the track-by-detector scheme while exploring the structure of the circulant matrix for the high efficiency. In the following, we briefly review the main idea of KCF tracker [14].

In KCF [14], Henriques et al. assume that the cyclic shifts version of base sample is able to approximate the dense samples over the base sample. Suppose that we have a one-dimensional data $\mathbf{x} = [x_1, x_2, ..., x_n]$, a cyclic shift of $\mathbf{x}$ is $\mathbf{Px} = [x_n, x_1, x_2, ..., x_{n-1}]$. The experiments show that such assumption is held reasonably in most of cases. Therefore, all the cyclic shift visual samples, $\{\mathbf{P^u x} | u = 0...n-1\}$, are concatenated to form the data matrix $\mathbf{X} = C(\mathbf{x})$. As the data matrix is purely generated by the cyclic shifts of $\mathbf{x}$, it is called *circulant matrix*. It has an intriguing property [28] that all the circulant matrices can be expressed as below:

$$\mathbf{X} = \mathbf{F}^H \mathbf{diag}(\mathbf{Fx})\mathbf{F} \tag{1}$$

where $\mathbf{F}$ is known as the DFT matrix, which transforms the data into Fourier domain, and $\mathbf{F}^H$ is the Hermitian transpose of $\mathbf{F}$. The decomposition of circulant matrix can be employed to simplify the solution of linear regression. The

objective function of linear ridge regression can be formulated as follows:

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

where the function f can be written as the linear combination of basis samples: $f(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\mathbf{x}$. The ridge regression has the close-form solution, $\mathbf{w} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$. Substituted by Eqn.1, we have the solution $\hat{\mathbf{w}}^* = \frac{\hat{\mathbf{x}}^* \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda}$, where $\hat{\mathbf{x}} = \mathbf{F}\mathbf{x}$ denotes the DFT of $\mathbf{x}$, and $\hat{\mathbf{x}}^*$ denotes the complex-conjugate of $\hat{\mathbf{x}}$. Compared with the prevalent method, this solution saves the computational cost of both extracting patches explicitly and solving a general regression problem [14]. In the case of no-linear regression, kernel trick, $f(\mathbf{z}) = \mathbf{w}^T\mathbf{z} = \sum_{i=1}^{n} \alpha_i \mathcal{K}(\mathbf{z}, \mathbf{x}_i)$, is applied to allow more powerful classifier. For the most commonly used kernel functions, the circulant matrix trick can also be used [14]. The dual space coefficients $\boldsymbol{\alpha}$ can be learnt as below

$$\hat{\boldsymbol{\alpha}}^* = \frac{\hat{\mathbf{y}}}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda} \tag{3}$$

where $\mathbf{k}^{\mathbf{xx}}$ is defined as *kernel correlation* in [14]. Similar to the linear case, the dual coefficients are learnt in Fourier domain. The inference is valid for the case that kernel function treats each dimension of the data equally [14]. In this paper, we adopt the Gaussian kernel which can be applied the circulant matrix trick as below:

$$\mathbf{k}^{\mathbf{xx}'} = \exp(-\frac{1}{\sigma^2}(||\mathbf{x}||^2 + ||\mathbf{x}'||^2) - 2\mathbf{F}^{-1}(\hat{\mathbf{x}} \odot \hat{\mathbf{x}}'^*)) \tag{4}$$

As the algorithm only requires dot-product and DFT/IDFT, the computational cost is in $O(n \log n)$ time. The training label $\mathbf{y}$ is a Gaussian function, which decays smoothly from the value of one for the centered target to zero for other shifts. As zero means the negative sample, we need to enlarge the original target bounding box to enclose the negative samples. In this paper, we employ the window with the size of 2.5 times larger than its original target box for training. Although the cyclic shift lost lots of information on the original frame, the classifier obtains the dense samples to fit the model more precisely.

The circulant matrix trick can also be applied in detection to speed up the whole process. The patch $\mathbf{z}$ at the same location in the next frame is treated as the base sample to compute the response in Fourier domain,

$$\hat{\mathbf{f}}(\mathbf{z}) = (\hat{\mathbf{k}}^{\tilde{\mathbf{x}}\mathbf{z}})^* \odot \hat{\boldsymbol{\alpha}} \tag{5}$$

where $\tilde{\mathbf{x}}$ denotes the data to be learnt in the model. When we transform $\hat{\mathbf{f}}(\mathbf{z})$ back into the spatial domain, the translation with respect to the maximum response is considered as the movement of the tracked target. The motion model implied that the searching range is the window size for the base patch. Although the whole model follows the tracking-by-detection scheme, there are only two samples in the process, both at the same position sampled in the last frame and current frame. Intuitively, it is more like a similarity metric in Fourier domain. In addition, Bolme et al. [7] give another interpretation on the whole process. For the more detailed formulation, please refer to [14] [7].

3.2 Multiple Feature Integration

Since the kernel correlation function only needs to calculate the dot-product and vector norm, multiple channels can be applied for the image features. Suppose the multiple channels of the data representation are concatenated into a vector $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_C]$. Eqn. 4 can be rewritten as follows:

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = \exp\left(-\frac{1}{\sigma^2}(||\mathbf{x}||^2 + ||\mathbf{x}'||^2) - 2\mathbf{F}^{-1}\left(\sum_{\mathbf{C}} \hat{\mathbf{x}}_{\mathbf{c}} \odot \hat{\mathbf{x}}_{\mathbf{c}}'^{*}\right)\right) \tag{6}$$

which allows us to use the more strong features rather than the raw greyscale pixels. Moreover, we can employ various powerful features to exploit the advantages of feature fusion. There are three types of features used in our proposed tracker. Besides the raw greyscale pixel of the original image, we adopt two commonly used features in visual tasks.

Histogram of Gradient (HoG) is one of the most popular visual features in vision community, since it is very effective in practical applications and can be computed very efficiently. The feature extracts the gradient information from a cell, which is a range of pixels. HoG counts the discrete orientation to form the histogram. As in [9], we employ the 31 gradient orientation bins variant in our method.

Color-naming or color attributes, is a perspective space, which is the linguistic color label assigned by human to describe the color. Being better than the RGB space, the distance in color label space is more similar to human sense. As achieved the promising results in other visual tasks such as object recognition, object detection and action recognition [17] [19] [18], we employ the mapping method described in [32] to transform the RGB space into the color names space, which is an 11 dimensional color representation. Color names provide the perception of object color, which usually contains the important information on the target.

The two features are complementary to each other. HoG puts emphasis on the image gradient while color naming focuses on the color information. In section 4.2, we will testify the efficacy of these features separately. Although the idea is quite straightforward, the performance gain is very promising. Note that the feature sizes do not consist with each other at first and alignment should be applied for the features data for the correlation filter.

3.3 Multiple Scale Kernelized Correlation Filter

As described in Section 3.1, the whole process is straightforward. Moreover, KCF is unable to deal with the scale changes in videos. To this end, we propose a scale adaptive method to enable the naive correlation filter tracker to deal with the scale variations.

In Section 3.1, the searching strategy is implied in the kernel correlation filter. We employ the bilinear interpolation to enlarge the image representation space from the countable integer space into the uncountable float space. We fix the template size as $\mathbf{s_T} = (s_x, s_y)$, and define a scaling pool $\mathbf{S} = \{t_1, t_2, ..., t_k\}$. Suppose that the target window size is $\mathbf{s_t}$ in the original image space. For the current frame, we sample k sizes in $\{t_i\mathbf{s_t}|t_i \in \mathbf{S}\}$ to find the proper target. Note that the dot-product in kernel correlation function needs the data with the fixed size. In this paper, we employ bilinear-interpolation to resize the samples into the fixed template size $\mathbf{s_T}$, and the final response is calculated by

$$\arg\max \mathbf{F}^{-\mathbf{1}}\hat{\mathbf{f}}(\mathbf{z}^{t_i}) \tag{7}$$

where $\mathbf{z}^{t_i}$ is the sample patch with the size of $t_i\mathbf{s_t}$, which is resized to $\mathbf{s_T}$. Since the response function obtains a vector, the max operation is employed to find its maximum scalar. As the target movement is implied in the response map, the final displacement needs to be tuned by t to get the real movement bias.

Note that all the templates are registered to the same size. Thus, the update procedure is straightforward. There are two sets of coefficients should be updated. One is the dual space coefficients $\boldsymbol{\alpha}$, and another is the base data template $\widetilde{\mathbf{x}}$. As in [14], we linearly combine the new filter with the old one as below:

$$\bar{\mathbf{T}} = \theta\mathbf{T}_{new} + (1-\theta)\bar{\mathbf{T}} \tag{8}$$

where $\mathbf{T} = [\boldsymbol{\alpha}^T, \widetilde{\mathbf{x}}^T]^T$ is the template to be updated. With the scale adaptive scheme, the proposed tracker is able to deal with the size changes. The overall algorithm is summarized into Algorithm 1.

Algorithm 1. Overall algorithm of SAMF

Require:
The template for the tracked target, $\widetilde{\mathbf{x}}$;
The dual space coefficient, $\boldsymbol{\alpha}$;
The newly arrived observation, $\mathbf{y}$;
The last frame position, $\mathbf{p}_{old}$;

Ensure:
The updated template for the tracked target, $\widetilde{\mathbf{x}}$;
The updated dual space coefficient, $\boldsymbol{\alpha}$;
The new position, $\mathbf{p}_{new}$;

1: **for** every t_i in **S do**
2: Sample the new patch $\mathbf{z}^{t_i}$ based on size $t_i\mathbf{s_t}$ and resize it to $\mathbf{s_T}$ with multiple features.
3: calculate the response $\hat{\mathbf{f}}(\mathbf{z}^{t_i})$ with Equation 5 and 6.
4: **end for**
5: Get final position $\mathbf{p}_{new}$ and size $t_i\mathbf{s_t}$ according to Equation 7
6: Get $\widetilde{\mathbf{x}}_{new}$ based on new position $\mathbf{p}_{new}$ and size $t_i\mathbf{s_t}$, and calculate $\boldsymbol{\alpha}_{new}$ with Equation 3.
7: Use Equation 8 to update $\widetilde{\mathbf{x}}$ and $\boldsymbol{\alpha}$ with $\widetilde{\mathbf{x}}_{new}$ and $\boldsymbol{\alpha}_{new}$.
8: **return** updated $\widetilde{\mathbf{x}}$ and $\boldsymbol{\alpha}$;

4 Experiments

We conduct three experiments to evaluate the efficacy of our proposed tracker. Firstly, we implemented three trackers with various settings, including Multiple Features tracker (MF), Scale Adaptive tracker (SA) and the proposed Scale Adaptive with Multiple Features tracker (SAMF). We compare them with other correlation filter-based trackers. Secondly, we evaluate our proposed tracker against the state-of-the-art trackers to show the effectiveness of our proposed SAMF tracker. Additionally, we report the detailed evaluation on VOT 2014 dataset.

4.1 Experimental Setup and Methodology

We implemented the proposed tracker by native Matlab without optimization. All the experiments are conducted on an Intel i5-760 CPU (2.80 GHz) PC with 16 GB memory. Our proposed SAMF tracker runs at about 7 fps. The σ used in Gaussian function is set to 0.5. The cell size of HoG is 4×4 and the orientation bin number of HoG is 9. The learning rate θ is set to 0.01. We use the scaling pool $\mathbf{S} = \{0.985, 0.99, 0.995, 1.0, 1.005, 1.01, 1.015\}$. All parameters are same for all following experiments.

In all the experiments, two evaluation criteria are used. The first one is mean center location error (CLE). CLE is the difference between the center of tracked results and the ground truth, where the smaller value means the more accurate result. The second criteria is the Pascal VOC overlap ratio (VOR) [8]. It is defined as $VOR = \frac{Area(B_T \cap B_G)}{Area(B_T \cup B_G)}$, where B_T is the tracking bounding box, and B_G is the ground truth bounding box. The larger value means the more accurate result.

To make comprehensive evaluation on the proposed approach, we employ the whole 51 video sequence in the benchmark [33] for the first two experiments. Moreover, we run the proposed tracker on VOT 2014 dataset containing 25 sequences. In VOT 2014 challenge, the accuracy is measured by the VOR score. The robustness indicates the failing time for a tracker on the sequence.

4.2 Experiment 1: Comparison between Correlation Filter-based Trackers

To evaluate the performance gain of our proposed scale adaptive scheme with multiple features, we run six variants of trackers on the benchmark [33], including SAMF, MF, SA, KCF, CN and CSK. All of these trackers takes advantage of the circulant matrix or kernel correlation filter. Table 1 summarizes the difference for these trackers. Figure 2 shows the CEL curves and VOR curves for those trackers. Although their ideas are very similar, the tracking performances are quite different. This indicates that the visual features and search strategy are essentially important to the visual tracking tasks. CSK only employs the raw pixel, whose rank is the lowest one among the compared trackers. CN adopts both

Table 1. The difference among six trackers

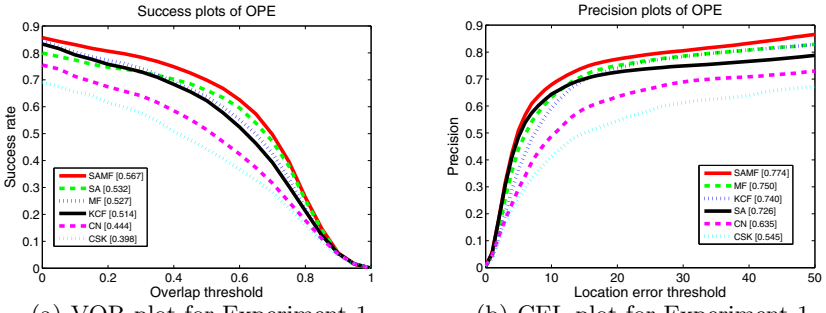
name	Features	Scale adaptive
SAMF	raw pixel, HoG, Color label	Yes
SA	HoG	Yes
MF	raw pixel, HoG, Color label	No
KCF [14]	HoG	No
CN [4]	raw pixel, Color label	No
CSK [12]	raw pixel	No

color names and raw pixel as features, and achieves a few improvement upon CSK. MF outperforms the KCF by augmenting the features space with color information and raw pixel. As shown in VOR curve, SA obtains a large improvement in accuracy shows. However, the robustness is decayed in the CEL curve. This demonstrates that expanding the search range will lead to the problem of local maximum. By taking advantage of the fusion features and the proposed scale adaptive scheme, SAMF tracker achieved the best performance in both VOR and CEL metrics.

The results from our experiment shows that our proposed tracker is very promising both in robustness and accuracy. The experiment also suggests that the feature and search strategy play very important role in visual tracking. Comparing to KCF, the VOR performance gains of SA and MF are 3.8% and 2.7% respectively while the SAMF gets a 10.6% improvement upon KCF. This indicates that the SAMF is not just the simple combination of the MF and SA, which can effectively capture the color information while accurately estimating the size of object.

4.3 Experiment 2: Comparison with the State-of-art Trackers

Table 2 illustrates the overall performance for the six trackers compared with the top two trackers reported in benchmark [33]. In the experiments, we observe



(a) VOR plot for Experiment 1 (b) CEL plot for Experiment 1

Fig. 1. The benchmark overall plot of the six kernel correlation filter based trackers

that the mean VOR score will be below 50% when the tracker loses the target in the sequence. Therefore, we define the successfully tracked sequence for a given tracker when the mean overlap of the whole sequence is above 0.5. The total number of the successfully tracked sequences can be viewed as a comprehensive metric of the tracker. The trackers with HoG feature achieved the very appealing performance compared against SCM and Struck in all the methods. SAMF achieves the best performance in terms of both mean CLE and mean VOR. Impressively, our approach achieves 57.4% in mean VOR overall, which is 10% improvement over the KCF tracker. In addition, the proposed tracker successfully tracked 37 of 51 sequences in the benchmark. This demonstrates that 72.5% of the sequences in the benchmark can be tracked, which is a big improvement for the visual object trackers.

Figure 2 shows the detailed report of SAMF compared with the top rank trackers, KCF [14], SCM [36], Struck [11], CN [4], TLD [16], ASLA [15], CXT [5], VTS [20], DFT [29], CPF [26], LSK [23], LOT [25] and VTD [21] in the benchmark. SAMF ranks the first with a large margin comparing to other trackers. Although the SAMF is not specially designed for occlusions, deformations and out-of-plane rotations, surprisingly, the proposed tracker obtains very appealing performances on these challenging video sequences. These promising results

Table 2. Overall comprehensive evaluation

	SAMF	SA	MF	KCF	CN	CSK	SCM	Struck
mean CEL	30.09	39.91	34.55	35.49	64.68	88.78	54.13	50.57
mean VOR	0.574	0.539	0.533	0.519	0.448	0.401	0.505	0.478
Passed Num.	37	32	32	31	23	18	28	28

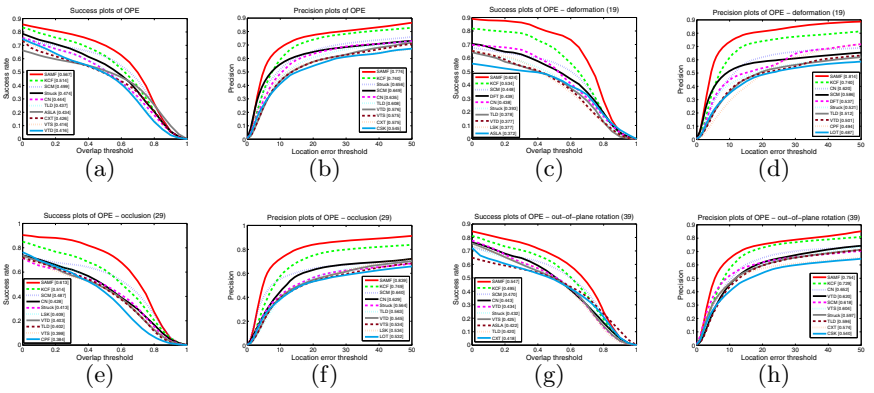


Fig. 2. The plot curves for the proposed tracker compared with 9 state-of-art trackers in the benchmark. (a)-(h) indicate the VOR and CEL of overall, deformation, occlusion and out-of-plane rotation, respectively.

Table 3. The results of VOT 2014

	accuracy						robustness					
	SAMF	KCF	NCC	SAMFn	KCFn	NCCn	SAMF	KCF	NCC	SAMFn	KCFn	NCCn
ball	0.772	0.702	0.740	0.738	0.640	0.633	1	1	30	0.47	1	26.1
basketball	0.748	0.574	0.573	0.640	0.562	0.577	0	2	30	0	2	30.7
bicycle	0.613	0.454	0.717	0.659	0.516	0.678	0	1	9	0.13	0.4	9.93
bolt	0.562	0.522	0.206	0.555	0.510	0.447	2	3	33	1.93	2.6	32.7
car	0.508	0.421	0.708	0.521	0.402	0.646	0	0	6	0.07	0	6.07
david	0.817	0.746	0.691	0.763	0.691	0.623	0	0	16	0	0	14.9
diving	0.245	0.233	0.269	0.209	0.226	0.233	4	5	8	4.4	4.8	6.87
drunk	0.568	0.434	0.364	0.542	0.481	0.423	0	0	4	0	0.53	4.07
fernando	0.394	0.402	0.575	0.393	0.393	0.331	1	1	15	1	1.13	13.3
fish1	0.495	0.438	0.564	0.472	0.445	0.541	3	3	16	2.73	3.27	16.5
fish2	0.296	0.299	0.265	0.294	0.257	0.189	5	4	14	4.80	5.47	12.4
gymnastics	0.536	0.528	0.663	0.467	0.489	0.402	2	3	8	2.47	2.2	7.4
hand1	0.544	0.389	0.515	0.417	0.408	0.378	3	6	13	5.33	4.8	14.8
hand2	0.462	0.438	0.275	0.400	0.443	0.230	5	8	15	7.07	7.87	16.8
jogging	0.819	0.760	0.795	0.674	0.655	0.696	1	1	3	0.93	1.07	3.33
motocross	0.400	0.372	0.326	0.351	0.349	0.208	4	5	9	3.4	4	9.07
polarbear	0.708	0.662	0.750	0.672	0.649	0.620	0	0	3	0	0	2.6
skating	0.452	0.488	0.675	0.526	0.530	0.563	0	0	26	0.07	0.4	26.7
sphere	0.879	0.713	0.643	0.796	0.664	0.674	0	0	1	0	0	2.27
sunshade	0.758	0.761	0.775	0.684	0.718	0.723	0	0	5	0	0	5.33
surfing	0.800	0.797	0.889	0.728	0.738	0.793	0	0	0	0	0	0
torus	0.840	0.757	0.507	0.752	0.687	0.376	0	0	17	0.07	0.27	15.9
trellis	0.814	0.546	0.600	0.732	0.506	0.525	0	0	29	0	0	27.5
tunnel	0.545	0.318	0.719	0.494	0.292	0.639	0	0	10	0	0	9.33
woman	0.758	0.755	0.745	0.734	0.687	0.611	1	2	23	1	2.07	21.5
Mean	0.613	0.540	0.582	0.569	0.518	0.510	1.28	1.80	13.72	1.43	1.75	13.44

suggest that the effective features and proper search strategy are more effective than the complicated models for deformations and occlusions.

4.4 Experiment 3: VOT 2014

Finally, we evaluate our proposed tracker on VOT 2014 dataset. The results are summarized into Table 3. Compared against KCF [14] and the baseline NCC tracker provided by the VOT organizer [1], SAMF achieves the higher performance both in accuracy and robustness. NCC performs quite well in accuracy but poor in the robustness. This is because NCC obtains more ground truth labels when it fails to track the target. Benefited from the correlation filter, KCF achieves an appealing score in robustness, however, it ranks at the last place in the accuracy due to the template with the fixed size. The proposed SAMF achieves the best results on both the accuracy and robustness. It can be seen that our proposed SAMF tracker performs especially well in case of robustness meanwhile it maintains the highest accuracy compared with other two trackers. This consists with the experimental results illustrated in Section 4.3.

[1] http://votchallenge.net/vot2014/index.html

5 Conclusions

This paper presented a very effective tracker based on the framework of correlation filter. We proposed the scale adaptive scheme to deal with the problem of the fixed template size in the conventional kernel correlation filter tracker. Moreover, the powerful features including HoG and color naming are fused together to further boost the overall performance for the proposed tracker. The extensive empirical evaluations on the benchmark videos and VOT 2014 dataset demonstrate that the proposed method is very promising for the various challenging scenarios. Our method successfully tracked the targets in about 72% videos and outperformed the state-of-the-art trackers on the benchmark dataset with 51 sequences.

Acknowledgments. The work was supported by National Natural Science Foundation of China under Grants (61103105 and 91120302).

References

1. Babenko, B., Yang, M.H., Belongie, S.: Robust object tracking with online multiple instance learning. TPAMI **33**(8), 1619–1632 (2011)
2. Boddeti, V.N., Kanade, T., Kumar, B.V.: Correlation filters for object alignment. In: CVPR (2013)
3. Chen, D., Yuan, Z., Wu, Y., Zhang, G., Zheng, N.: Constructing adaptive complex cells for robust object tracking. In: ICCV (2013)
4. Danelljan, M., Khan, F.S., Felsberg, M., van de Weijer., J.: Adaptive color attributes for real-time visual tracking. In: CVPR (2014)
5. Dinh, T.B., Vo, N., erard Medioni, G.: Context tracker: Exploring supporters and distracters in unconstrained environments. In: CVPR (2011)
6. D.S.Bolme, B.A.Draper, J.R.Beveridge: Average of synthetic exact filters. In: CVPR (2009)
7. D.S.Bolme, J.R.Beveridge, B.A.Draper, Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: CVPR (2010)
8. Everingham, M., Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classese(voc) challenge. IJCV **88**(2), 303–338 (2010)
9. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. TPAMI (2010)
10. Galoogahi, H.K., Sim, T., Lucey, S.: Multi-channel correlation filters. In: ICCV (2013)
11. Hare, S., Saffari, A., Torr, P.H.S.: Struck: Structured output tracking with kernels. In: ICCV (2011)
12. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 702–715. Springer, Heidelberg (2012)
13. Henriques, J.F., Carreira, J., Caseiro, R., Batista, J.: Beyond hard negative mining: Efficient detector learning via block-circulant decomposition. In: ICCV (2013)
14. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. TPAMI (2014)

15. Jia, X., Lu, H., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: CVPR, pp. 1822–1829. Providence, June 2012
16. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. In: PAMI (2011)
17. Khan, F.S., Anwer, R.M., van de Weijer, J., Bagdanov, A., Lopez, A., Felsberg, M.: Coloring action recognition in still images. IJCV 105(3), 205–221 (2013)
18. Khan, F.S., Anwer, R.M., van de Weijer, J., Bagdanov, A., Vanrell, M., Lopez, A.: Color attributes for object detection. In: CVPR (2012)
19. Khan, F.S., van de Weijer, J., Vanrell, M.: Modulating shape features by color attention for object recognition. IJCV 98(1), 49–64 (2012)
20. Kwon, J., Lee, K.M.: Tracking by sampling trackers. In: ICCV (2011)
21. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: CVPR (2010)
22. Lim, J., Ross, D., Lin, R.S., Yang, M.H.: Incremental learning for visual tracking. In: NIPS (2004)
23. Liu, B., Huang, J., Yang, L., Kulikowsk, C.: Robust tracking using local sparse appearance model and k-selection. In: CVPR (2011)
24. Mei, X., Ling, H.: Robust visual tracking using l1 minimization. In: ICCV (2009)
25. Oron, S., Bar-Hillel, A., Levi, D., Avidan, S.: Locally orderless tracking. In: CVPR (2012)
26. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-Based Probabilistic Tracking. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 661–675. Springer, Heidelberg (2002)
27. Revaud, J., Douze, M., Cordelia, S., Jgou, H.: Event retrieval in large video collections with circulant temporal encoding. In: CVPR (2013)
28. Gray, R.M.: Toeplitz and circulant matrices: A review. Now Publishers **77**(1–3), 125–141 (2006)
29. Sevilla-Lara, L., Learned-Miller, E.: Distribution fields for tracking. In: CVPR (2012)
30. Wang, D., Lu, H., Yang, M.H.: Least soft-thresold squares tracking. In: CVPR. Portland, June 2013
31. Wang, N., Yeung, D.Y.: Learning a deep compact image representation for visual tracking. In: NIPS (2013)
32. van de Weijer, J., Schmid, C., Verbeek, J.J., Larlus, D.: Learning color names for real-world applications. TIP 18(7), 1512–1524 (2009)
33. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: CVPR (2013)
34. Zhang, K., Zhang, L., Yang, M.-H.: Real-Time Compressive Tracking. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part III. LNCS, vol. 7574, pp. 864–877. Springer, Heidelberg (2012)
35. Zhang, L., van der Maaten, L.: Structure preserving object tracking. In: CVPR (2013)
36. Zhong, W., Lu, H., Yang, M.H.: Robust object tracking via sparsity-based collaborative model. In: CVPR, pp. 1838–1845, Providence, June 2012