Neurocomputing 257 (2017) 115–127

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Online object tracking based on CNN with spatial-temporal saliency guided sampling

Peng Zhang[a,*], Tao Zhuo[b], Wei Huang[c], Kangli Chen[a], Mohan Kankanhalli[d]

[a] School of Computer Science, Northwestern Polytechnical University, Xi'an, China
[b] Sensor-enhanced Social Media (SeSaMe) Centre, National University of Singapore, Singapore
[c] School of Information Engineering, Nanchang University, China
[d] School of Computing, National University of Singapore, Singapore

ARTICLE INFO

Article history:
Received 27 June 2016
Revised 8 October 2016
Accepted 10 October 2016
Available online 6 February 2017

Keywords:
Tracking
CNN
Spatial-temporal
Saliency
Sampling

ABSTRACT

Arbitrary tracking is hard due to nonstop intrinsic and extrinsic variations in realistic scenarios. Even for the popular tracking-by-learning strategies, effective appearance modeling of the non-rigid objects is still challenging because of the targets' articulatory deformations on-the-fly, which may heavily degrade the discriminative capability of the online generated visual features. With widely emerged deep learning showing its success for feature extraction in different recognition tasks, more and more deep models such as CNN have been demonstrated contributive to improving the performance of online tracking. However, only depending on the outputs from last layer of CNN is not an optimum representation since the coarse spatial resolution cannot guarantee an accurate localization for a qualified sampling process, especially when objects have severe deformations, sampling from the region with a pre-defined scale would inevitably guide a poor online learning. To overcome such a limitation of CNN based tracking, in this work, we incorporated spatial-temporal saliency detection to guide a more accurate target localization for qualified sampling within an inter-frame motion flow map. With an optional strategy for the output combination of intra-frame appearance correlations and inter-frame motion saliency based on a compositional energy optimization, the proposed tracking has shown a superior performance in comparison to the other state-of-art trackers on both challenging non-rigid and generic tracking benchmark datasets.

1. Introduction

Long term tracking is a key technique demanded by multimedia analysis such as video understanding or object based encoding in the coming age of big data [41]. In realistic scenarios, tracking an arbitrary object is still challenging because of its drastic and unpredictable appearance changes. The initially proposed tracking approaches assumed that most possible object status can be known in advance for appearance model pre-training [1–3,56], unfortunately, tracking failure easily happened to those off-line trackers because of the poor adaptiveness to dynamic variations. Thus, to fulfil the runtime specification requirement of arbitrary tracking, online tracking-by-learning strategy has gradually become popular and achieved impressive performance in recently proposed tracking work [4].

The most popular way to conduct online tracking-by-learning is to continually update the target appearance model by supervised or semi-supervised learning [19], which has shown a promising performance in articulated object tracking such as structured SVMs [6–8]. Hare et al. proposed a structured output tracking (Struck) strategy [7] to avoid the intermediate classification step by explicitly allowing to express different trackers' demands in their output space, but Struck tracker was robust in challenge scenarios [4] owing to its kernelized structured online SVM learning [29]. Consequently, an advanced online version of structured output SVM tracker 'KCF' was proposed by Henriques et al. [8] by the motivation of Hare's Struck. Based on well-established theory of circulant matrices, Henriques built up the possibility of extremely efficient learning and detection in frequency domain with Fast Fourier Transform (FFT) and the biggest bright spot of KCF is the 300 fps running speed, which can be easily implemented on normal PC platform, and the related extended work of KCF can be found in [9,10]. In most cases, the sample quality is an essential factor for the online learning based tracking, even the previously discussed tracking approaches have tried different semi-supervised or supervised ways to guarantee the learning sample accuracy, but the information known to the learner is usually deficient before learning process starts, together with impossibility of incorporating manual interaction into the process of sample generation

* Corresponding author.
E-mail addresses: zh0036ng@nwpu.edu.cn (P. Zhang), zhuotao@nus.edu.sg (T. Zhuo), huangwei@ncu.edu.cn (W. Huang), chenkangli@mail.nwpu.edu.cn (K. Chen), mohan@comp.nus.edu.sg (M. Kankanhalli).

http://dx.doi.org/10.1016/j.neucom.2016.10.073
0925-2312/

and labeling while tracking on-the-fly. Therefore, such cursory connection among samples would lead to performance degeneration when model is updated, which means the descriptive capability of those online generated samples were still insufficient in representing different objects' characteristics.

To obtain more descriptive features, the quickly emerged deep learning technology [5,11] has demonstrated its outstanding capabilities in a variety of computer vision and pattern recognition tasks such as image/video classification [42] and object detection [12,40]. However, the extensive computational cost and the requirement of large quantity of data became the main obstacles that limited the deep learning to be applied in different applications with online learning [13,14,16,43]. Hong et al. utilized a pre-trained model to carry out the learning of a discriminative saliency detection using CNN [16] and Wang et al. proposed an online tracker based on an off-line trained CNN [13], which requires a compact image representation learnt from a large quantity of auxiliary images. Differing from [13] and [16], Li et al. proposed an online tracking by extending CNN by formulating a truncated structural loss functional model [14]. Li's work has shown a promising tracking performance in some scenarios, but this work mainly focused on changing CNN from off-line mode to online mode instead of on the generation of training samples. For sample generation online, some other tracking based on deep learning perform the feature extraction by CNN with an incremental scheme as [26,27]. Wang et al. proposed to learn the generic features from auxiliary video sequences by using a two-layer convolutional neural network with an adaptation module which was introduced to adapt the pre-learned features according to specific target objects [26], but its embedded temporal slowness constraints are also an inevitably impose for the feature learning process. Such a double-layer structure also appeared in Li's DeepTrack [27]. Without a pre-trained procedure, Li's DeepTrack constructed multiple CNN classifiers on different objects' instances to rule out noise samples during model update, and the model update is performed by fine-tuning its deep models online which means that only the outputs from the last layer are used to represent targets as noted in [15,17,44,45]. One noticeable schema shared by those discussed CNN trackers is the intensive dependence of the last layer's output features because they are closely related to category-level semantics and most invariant to challenging intra-class variations or precise location. But considering the most important purpose of tracking is to continually localize target position/trajectory rather than offering its semantic understanding, only depending on the information of last layer in CNN model would not be an optimal way to achieve long-term tracking due to the helpless representation from coarse spatial resolution.

After investigating the lower convolutional layers of a CNN model being able to help localize the target position more accurately even if they have limitations in handling the object appearance changes, Ma et al. explained the multi-level of CNN structure as a hierarchical convolutional layers of an image pyramid representation [17], which hierarchically search the maximum response of each layer to localize target region for online sample generation by adaptively learning correlation filters on each convolutional layer as an appearance encoder. Unfortunately, the extracted target region based on maximum spatial correlation response in each layer in [17] cannot always guarantee to guide a qualified sampling process because its inherent scaling invariance definition, especially when tracking a deformable non-rigid object, the predefined fixed-scale has high risk to lead an inaccurate target region for sampling, which also means that only depending on the spatial correlation is far from sufficiency. Therefore, how to effectively incorporate the spatial-temporal motion with spatial information for more accurate target region localization.

Motivated by the recent achievements of video saliency extraction [18] of adaptive scaled target region, in this study, we proposed a deep learning based online object tracking by incorporating the intra-frame appearance correlations and inter-frame motion saliency into an compositional energy optimization process for qualified sampling region localization. The final output target localization with optional strategy has demonstrate effective in guiding a qualified sampling process for tracking-by-learning with CNN, which improve both the accuracy and robustness of online performance.

The organization of the paper is as follows: Section 2 briefly introduces the related work as the preliminary knowledge for understanding the proposed work. Section 3 gives the symbols and abbreviations definitions utilized in presenting our approach of spatial-temporal saliency guided target localization and tracking framework in Sections 4 and5. In Section 6, the tracking performance comparison is shown and discussed in both qualitative and qualitative analysis on different benchmark datasets. Finally, we conclude this paper in Section 7.

2. Related work

In this section, we briefly introduce circulant matrix based kernelized correlations and video saliency detection /segmentation as the important preliminary knowledge to help understanding the technical descriptions presented in the proposed work.

2.1. Circulant matrix based kernelized correlations

Correlation filters by FFT have demonstrated a high computational efficiency for different recognition tasks by regressing all the circular-shifted input features to a target function, e.g. Gaussian, to remove the dependence of hard-thresholded sample of object appearance. Let an object region of interest (ROI) in a video frame be represented as a basic sample $\mathbf{x}$, which is denoted as an $n \times 1$ vector. According to the circulant theory in [8], arbitrary translation of $\mathbf{x}$ can be calculated by a cyclic shift operation with the permutation matrix $\mathbf{A}$, and a small translation of $\mathbf{x}$ can be modeled by using the product $\mathbf{Ax} = [x_n, x_1, x_2, \ldots, x_{n-1}]$ to shift $\mathbf{x}$ by one element, where the permutation matrix $\mathbf{A}$ is defined as Eq. (1):

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \tag{1}$$

Then, a big step translation can be formulated by moving u shifts with the matrix power $\mathbf{A}^u\mathbf{x}$, which also means the usage of negative u would result in an opposite direction. Based on this cyclic property, an identical $\mathbf{x}$ can be periodically obtained after each n shifts and thus the full set of shifted samples is denoted as $\chi = \{\mathbf{A}^u\mathbf{x} | u = 0, \ldots, n-1\}$. Furthermore, the first half of this set as shifts is in the positive direction and the second half is the negative. For extension, a circulant matrix can be composed by using the set of χ as the rows of data, which is defined as:

$$\mathbf{C}(\mathbf{x}) = \begin{pmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{pmatrix} \tag{2}$$

Although the obtained vector $\mathbf{x}$ can fully specify the pattern deterministically, all of those circulant matrices can be made diagonal

by the discrete Fourier transform without taking **x** into account, which is very useful in realistic implementations, which can refer [8] for more elaborative details.

2.2. Video saliency detection/segmentation

Although extraction of target objects in a scene is related to accurate object recognition and classification, interestingly, a reliable saliency based moving object detection is often feasible without any actual scenario understanding, which is suitable for the motion region estimation between consecutive frames in online tracking task. Recently, different saliency based object detection methods have been proposed by referring the achievements in cognitive psychology and neurobiology [57]. Cheng et al. proposed a regional contrast based salient object detection algorithm [28] by simultaneously evaluating the global contrast differences and spatial weighted coherence scores. Wang et al. integrated spatiotemporal edge map and geodesic distance to obtain accurate spatiotemporal saliency results as a prior to object segmentation [30]. An efficient salient object detection was proposed by Zhang et al. based on Minimum Barrier Distance Transform [20]. Driven by such transform, Zhang's method could outperform on 4 large evaluation benchmark database and obtain comparable even better accuracy than state-of-the-art methods at nearly 80 fps. Via demonstrating that the visual saliency can be modeled by a quantum mechanics owing to its rich mathematical structure and unique properties, Aytekin et al. proposed a quantum-mechanical approach named Quantum-Cut for saliency detection [21]. Aytekin's Quantum-Cut successfully built a fully automatic, parameter free and a solid link between between graph-cut algorithms and ground-state wave functions obtained using a derived energy function, which came from a fundamental theorem of quantum mechanics named Schödinger's equation as: $-\frac{\hbar^2}{2\tilde{m}}\nabla^2\mathbf{\Psi} = -V(\mathbf{r})\mathbf{\Psi} + E\mathbf{\Psi}$ where $V(\mathbf{r})$ is a potential distribution and $\hbar$ is a Planck constant, $\mathbf{\Psi}$ represents a wave-function, which is explained as a eigenfunction of self-consistently and well-defined observable called $\mathcal{O}$ in physics such as energy E or mass $\tilde{m}$ [23]. With its corresponding operator $\hat{\mathcal{O}}$ as a measurement of $\mathcal{O}$, the solution to salient object detection/segmentation is to minimize the target function of $cut(\mathcal{O}, \hat{\mathcal{O}})/Area(\hat{\mathcal{O}})$.

3. Symbols and abbreviations definitions

The list below shows the symbols and abbreviations used in the following presentation.

Ω_m - inter-frame motion response maps	Θ - inter-frame flow field
$\rho_{i,j}$ - weights of boundary penalties	Ω_i^{obj} - the regional penalties of target object
Ω_i^{bkg} - regional penalties of background	$\mathcal{N}$ - a neighborhood of a node
$\mathbf{y}$ - label vector	$\mathcal{H}_m$ -Hamiltonian operator in the Schödinger's equation
$\mathbf{\Psi}^*\mathbf{\Psi}$ - the PDF of the quantum particles' occurrence	$\psi^*(i)\psi(i)$ - soft labeling function
s_t, s_{t+1} - two consecutive SIFT frames	$\mathbf{p}(m,n)$ - each pixel in frame s_t
$\mathbf{w}$ - SIFT motion flow vector at $\mathbf{p}$	ϵ - spatial neighborhoods
$\mathbb{D}$ - data term of SIFT features	$\mathbb{S}$ - smoothness term of flow neighborhood
Δ - fine-tuning term of flow vectors range	α, β - regularizer parameters
τ, d - pre-defined thresholds	$\mathbf{x}$ -a feature vector of length $M \times N$
$\mathbf{x}_{m,n}$ - circularly shifted samples	ω - a correlation filter
λ - a regularization parameter	κ - a kernel in the Hilbert space
$\langle\cdot\rangle$ - dot product operation	$\mathbb{K}$ - be a kernel matrix
$\mathbf{P}$ - a pre-defined permutation matrix	Φ -an $n \times n$ circulant matrix
υ - cyclic shifts of an $N \times 1$ vector	$*$ - element-wise complex-conjugation operator
$\mathbf{a}$ - a single test sample	$\hat{\mathbf{k}}$ - the vector with elements $\hat{k}_i$
$\hat{\mathbf{y}}$ - responses for all the estimations	$\mathbf{X}, \mathbf{Y}, \mathbf{W}$ - the Fourier transforms of $\mathbf{x}, \mathbf{y}, \omega$
$(\hat{m}, \hat{n})$ - final output optimal spatial localization	Ω - the final optimum localized target region
Ω_a - intra-frame appearance correlation response maps	θ - a pre-defined threshold for optional strategy
$\Omega_a(\bar{m}, \bar{n})$ - optimal appearance correlation response	$\Omega_m(\bar{m}, \bar{n})$ - optimal motion response
$S[\cap]/S[\cup]$ - region overlap function	$\mathcal{C}$ - an empirical threshold
b_t - the bounding box of the tracking result	b_g - the bounding box of ground truth
VOR - PASCAL VOC Overlap Ratio	CLE - Center Location Error

4. Proposed work

To effectively achieve deep learning based tracking through accurately localizing the sampling region, the optimal response by spatial target appearance correlation and spatial-temporal salient motion extraction need to be calculated in both intra-frame and inter-frame domain. In the following discussion, we start to present the proposed work with elaborative theoretical details.

4.1. Temporal target localization by salient motion detection

For temporal region localization between the consecutive frames, motion-saliency extraction is utilized to obtain the inter-frame motion response maps Ω_m from a flow field Θ, which is estimated along a pair of SIFT frames. Let each pixel of Θ be regarded as a node that has connection to both terminals and weights of boundary penalties $\rho_{i,j}$, the regional penalties of target object and background inside Θ can be denoted as Ω_i^{obj} and Ω_i^{bkg}, which represent the edge weights from a node to source and sink terminals [23]. Corresponding to the edges whose labels of nodes are different in a neighborhood $\mathcal{N}$, we use the label vector $\mathbf{y}$ to indicate the object ($y = 1$) or background ($y = 0$), the object region extraction can be achieved by minimizing:

$$\arg\min_{\mathbf{y}} \gamma \cdot \left(\sum_{i\in\Theta} \Omega_i^{obj}(1 - y_i) + \Omega_i^{bkg} y_i \right) + \left(\sum_{i\in\Theta} \sum_{j\in\mathcal{N}_i} y_i - y_i y_j \cdot \rho_{i,j} \right) \tag{3}$$

By exploring the link between interactive graph-cut [22] and quantum mechanics [23], in which the Schödinger's equation with a modified Hamiltonian operator $\mathcal{H}_m$ can then be defined as $\mathbf{\Omega} = \mathcal{H}_m \cdot \mathbf{\Psi}$. Referring the edge weight $\rho_{i,j}$ between nodes i, j, we can have:

$$\begin{aligned}\Omega_i &= \psi(i)\left(\Omega_i^{bkg} - \Omega_i^{obj} + \frac{\hbar^2 \sum_{j\in\mathcal{N}} \rho_{i,j}}{2\tilde{m}}\right) - \sum_{j\in\mathcal{N}} \psi(j) \frac{\hbar^2}{2\tilde{m}} \rho_{i,j} \\ &= \psi(i)\left(\Omega_i^{bkg} - \Omega_i^{obj}\right) + \frac{\hbar^2}{2\tilde{m}} \sum_{j\in\mathcal{N}} (\psi(i) - \psi(j)) \cdot \rho_{i,j}\end{aligned} \tag{4}$$

According to Schödinger's equation and simply following the properties of a PDF, $\mathbf{\Psi}^*\mathbf{\Omega} = \mathbf{\Psi}^* E_{\tilde{m}} \mathbf{\Psi}$, where $\mathbf{\Psi}^*\mathbf{\Psi} = 1$ is the PDF of the quantum particles' occurrence and can be analogous to a soft

labeling function $y_i = \psi^*(i)\psi(i)$. Therefore, the extraction of foreground object from background using Eq. (4) can be achieved by an energy minimization process similar as calculation of Hamiltonian eigenvalues in quantum mechanics as:

$$\begin{aligned} E_{\tilde{m}} &= \sum_{i\in\Theta}(\psi^*(i)\psi(i))\left(\Omega_i^{bkg}-\Omega_i^{obj}\right) \\ &\quad + \frac{\hbar^2}{2\tilde{m}}\sum_{i\in\Theta}\psi^*(i)\sum_{j\in\mathcal{N}}(\psi(i)-\psi(j))\cdot\rho_{i,j} \\ &= \sum_{i\in\Theta} y_i\left(\Omega_i^{bkg}-\Omega_i^{obj}\right) + \frac{\hbar^2}{2\tilde{m}}\sum_{i\in\Theta}\sum_{j\in\mathcal{N}}(y_i - y_i y_j)\cdot\rho_{i,j} \\ &\quad + \frac{\hbar^2}{2\tilde{m}}\sum_{i\in\Theta}\sum_{j\in\mathcal{N}}(\psi^*(i)\psi(j))(\psi(i)\psi^*(j)-1)\cdot\rho_{i,j} \end{aligned} \tag{5}$$

The Eq. (5) shows that the term $(\Omega_i^{bkg}-\Omega_i^{obj})$ of potential filed in quantum mechanics is able to guide an object detection/segmentation by minimizing $cut(\mathcal{O},\bar{\mathcal{O}})/Area(\bar{\mathcal{O}})$ inside a motion flow map, which is estimated from two consecutive SIFT frames s_t and s_{t+1}. Let each pixel in frame s_t denoted by $\mathbf{p}(m, n)$, $\mathbf{w} = [u(\mathbf{p}), v(\mathbf{p})]$ be its corresponding SIFT motion flow vector at $\mathbf{p}$ and ϵ denote the all the spatial neighborhoods, the motion flow map is generated by also using an energy based function:

$$\begin{cases} E(\mathbf{w}) = \mathbb{D}(s,\mathbf{p},\tau) + \mathbb{S}(u,v,d,\alpha) + \Delta(u,v,\beta) \\ \mathbb{D}(s,\mathbf{p},t) = \sum_{\mathbf{p}} \min\left(\|s_t(\mathbf{p}) - s_{t+1}(\mathbf{p}+\mathbf{w}(\mathbf{p}))\|_{L1}, \tau\right) \\ \mathbb{S}(u,v,d,\alpha) = \sum_{(\mathbf{p},\mathbf{q})\in\epsilon} \min\left(\alpha\cdot|u(\mathbf{p})-u(\mathbf{q})|, d\right) \\ \qquad\qquad + \min\left(\alpha\cdot|v(\mathbf{p})-v(\mathbf{q})|, d\right) \\ \Delta(u,v,\beta) = \sum_{\mathbf{p}} \beta\cdot(|u(\mathbf{p})| + |v(\mathbf{p})|) \end{cases}$$

where function $\mathbb{D}$ is the data term restricting the matching of SIFT features along the motion flow vector $\mathbf{w}$ at pixel $\mathbf{p}$ and function $\mathbb{S}$ is the smoothness term restricting the flow vectors of neighborhood pixels to be similar. The third term Δ is fine-tuning term to restrict the flow vectors within an acceptable small range when sufficient information can be guaranteed. Variables α, β are the regularizer parameters, and τ, d are the pre-defined thresholds. More details about motion flow generation can refer [24]

4.2. Spatial region localization by kernelized correlation filters

Suppose a feature vector $\mathbf{x}$ of length $M \times N$, and each circularly shifted sample $\mathbf{x}_{m,n}(m,n) \in \{0,1,\ldots,M-1\}\times\{0,1,\ldots,N-1\}$ is assumed to have a Gaussian function label $y(m, n)$ [17], then a correlation filter ω of identical size as $\mathbf{x}$ can be learned by minimizing the following function:

$$\omega^* = \arg\min_{\omega}\sum_{m,n}\|\omega\cdot\mathbf{x}_{m,n} - y(m,n)\|^2 + \lambda\|\omega\|_2^2 \tag{6}$$

where $\lambda(\lambda \geq 0)$ is defined as a regularization parameter and the inner product is performed in the Hilbert space with a kernel defined as $\kappa(\mathbf{x},\mathbf{x}') = \langle\varphi(\mathbf{x}),\varphi(\mathbf{x}')\rangle$, where $\langle\cdot\rangle$ denotes the dot product. Let $\mathbb{K}$ be a kernel matrix composed of elements $\mathbb{K}_{ij} = \kappa(\mathbf{x}_i,\mathbf{x}_j)$, with a pre-defined permutation matrix $\mathbf{P}$ that cyclically shifts vectors by only one element, all the possible translated versions of $\mathbf{x}$ can be defined as $\mathbf{x}_i = \mathbf{P}^i\mathbf{x}$ for row i, $\forall i = 0,\ldots,M-1$. Since the permutation matrix $\mathbf{P}$ is unitary, if κ is a unitary invariant kernel which fulfills $\kappa(\mathbf{x},\mathbf{x}') = \kappa(\mathbf{P}\mathbf{x},\mathbf{P}\mathbf{x}')$ for any unitary matrix $\mathbf{P}$, we can have $\mathbb{K}_{ij} = \kappa(\mathbf{P}^i\mathbf{x},\mathbf{P}^j\mathbf{x}) = \kappa(\mathbf{P}^{-i}\mathbf{P}^i\mathbf{x},\mathbf{P}^{-i}\mathbf{P}^j\mathbf{x}) = \kappa(\mathbf{x},\mathbf{P}^{j-i}\mathbf{x})$, which suggests that $\mathbb{K}_{ij}$ depends only on $(j-i)$ mod M.

The Gaussian kernels have the form $\kappa(\mathbf{x},\mathbf{x}') = \phi(\|\mathbf{x}-\mathbf{x}'\|^2)$, let vector $\mathbf{k}$ be the reduced form of kernel matrix $\mathbb{K}$, and the dimension of $\mathbb{K}$ is $M \times N$ and $\mathbf{k}$ is $N \times 1$. Then, each element k_i of $\mathbf{k}$ is given by:

$$k_i = \kappa(\mathbf{x},\mathbf{P}^i\mathbf{x}') = \phi(\|\mathbf{x}-\mathbf{P}^i\mathbf{x}'\|^2) \tag{7}$$

We can expand the norm, obtaining:

$$k_i = \phi(\|\mathbf{x}\|^2 + \|\mathbf{x}'\|^2 - 2\mathbf{x}^T\mathbf{P}^i\mathbf{x}') \tag{8}$$

Let $\Phi(\upsilon)$ denote an $n \times n$ circulant matrix obtained by concatenating all the possible cyclic shifts of an $N \times 1$ vector υ [31], and we assume the function ϕ can be performed by element-wise to any input vector, with the properties in Fourier transform domain that

$$\Phi(\|\mathbf{x}-\mathbf{x}'\|^2) = \mathcal{F}^{-1}(\mathcal{F}(\mathbf{x})\odot\mathcal{F}^*(\mathbf{x}')) \tag{9}$$

then the vector $\mathbf{k}$ can be calculated as:

$$\mathbf{k} = exp\left(-\frac{1}{\sigma^2}(\|\mathbf{x}\|^2 + \|\mathbf{x}'\|^2 - \mathcal{F}^{-1}(\mathcal{F}(\mathbf{x})\odot\mathcal{F}^*(\mathbf{x}')))\right) \tag{10}$$

where the operators $\odot$ and * denote the element-wise product and complex-conjugation, respectively. and function $\mathcal{F}$ is the Fourier transform operation. Eq. (10) guarantees that a dot-product kernel function can be efficiently performed for all estimated samples element-wise and by using only a few FFT operations. For a single test sample $\mathbf{a}$, $\vec{\mathbf{k}}$ is the vector with elements $\vec{k}_i = \kappa(\mathbf{a},\mathbf{P}^i\mathbf{x})$, and the responses $\hat{\mathbf{y}}$ for all the estimations can be efficiently obtained as:

$$\hat{\mathbf{y}} = \mathcal{F}^{-1}\left(\mathcal{F}(\vec{\mathbf{k}})\odot\mathcal{F}\left(\mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathbf{y})}{\mathcal{F}(\mathbf{k})+\lambda}\right)\right)\right) \tag{11}$$

Then, the process by Eq. (6) can be solved by FFT in each individual feature channel $d \in \{1,\ldots,D\}$ as $\mathbf{W}^d = (\mathbf{Y}\odot\bar{\mathbf{X}}^d)/(\sum_{i=1}^{D}\mathbf{X}^i\odot\bar{\mathbf{X}}^i + \lambda)$. Here, $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{W}$ are the Fourier transforms of $\mathbf{x}$, $\mathbf{y}$ and ω. The operator $\bar{}$ denotes the operation of complex conjugation. For another feature vector $\mathbf{x}'$ of size of $M \times N \times D$ coming in the next frame, the correlation response map is then computed by the inverse FFT transform as: $\mathcal{F}^{-1}(\sum_{D}^{d=1}\mathbf{W}^d\odot\mathbf{X}'^d)$. The final output optimal spatial localization $(\hat{m},\hat{n})$ can be found by scanning the maximum value of appearance correlation response $\arg\max_{m,n}\Omega_a(m,n)$ from all the possible shift translations. The theoretical analysis for kernel selection is beyond the scope of this study, and more details can be found in [31].

4.3. Spatial-temporal target region localization for sampling

In the proposed tracking, we perform the target region localization in both spatial and temporal domains by formulating an optional function, which is based effective composition of different localization strategies using an energy optimization process. Let Ω denote the final optimum localized target region that is calculated from the intra-frame appearance correlation response maps Ω_a and inter-frame motion response maps Ω_m of size $M \times N$.

$$\Omega = \begin{cases} \Omega_a(\bar{m},\bar{n}) & S[\Omega_a(\bar{m},\bar{n})\cap\Omega_m(\bar{m},\bar{n})]/ \\ & \qquad S[\Omega_a(\bar{m},\bar{n})\cup\Omega_m(\bar{m},\bar{n})] \geq \theta \\ \Omega_m(\bar{m},\bar{n}) & S[\Omega_a(\bar{m},\bar{n})\cap\Omega_m(\bar{m},\bar{n})]/ \\ & \qquad S[\Omega_a(\bar{m},\bar{n})\cup\Omega_m(\bar{m},\bar{n})] < \theta \end{cases}$$

where θ is a pre-defined threshold to balance the optimized output from the optimal appearance correlation response $\Omega_a(\bar{m},\bar{n})$ or motion response $\Omega_m(\bar{m},\bar{n})$ with the region overlap function $S[\cap]/S[\cup]$. This can also be regarded as an optional strategy for fusing the outputs of different models.

4.4. Convolutional neuron networks

Visual features play an essential role in the whole online learning based tracking. Especially in most recent years, the deep learning based feature extraction such as CNN has helped different

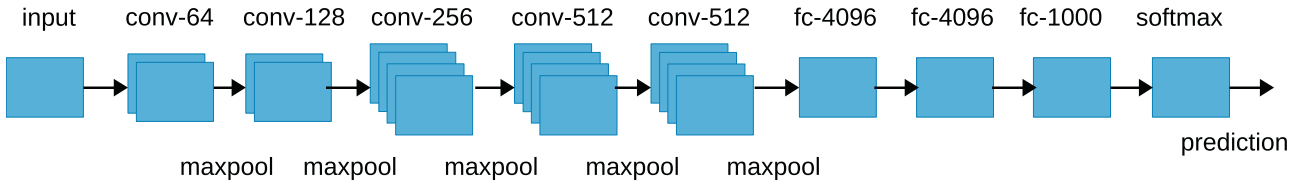


Fig. 1. CNN architecture of VGG-Net-19, conv denotes the convolutional layer and fc denotes full connected layer.

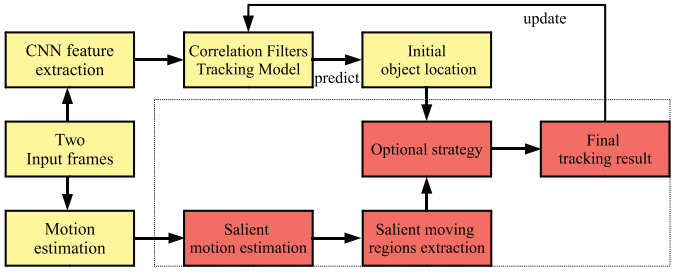


Fig. 2. Overview of the proposed tracking framework.

tracking approaches achieve a great performance improvement in comparison to other state-of-the-art trackers [17]. Among all the published CNN based deep learning models, VGG-Net [25] is a popular deep convolutional networks, which is implemented with large scale dataset ImageNet [39] aiming at a variety of recognition/classification challenges. In the VGG-Net, the features in the earlier layers contain higher spatial resolution for precise localization, while the features in latter layers indicate more semantic information and less spatial accuracy. In the proposed tracking framework, we adopt the VGG-Net-19 as robust feature extraction similar as [17]. In addition, the semantic information of latter layers is also exploited by us to adapt drastic target appearance changes in supplement to the earlier layers for precise localization.

In our tracking framework, the CNN model is initialized in the startup frame and continually updated in a lazy manner only when the confidence of estimated object is lower than an empirical threshold, e.g. $\mathcal{C} = 0.9$ in our case. During updating, the confidence of each positive and negative sample is strengthen over $\mathcal{C}$ since the CNN model can accept weighted training samples as the new incoming data. Similar as the boosting-cascade mechanism, the weight of each sample is then estimated according to the objective loss in former iteration [29], which is beneficial to enhance the speed of model update. The parameter configuration of CNN is the same as the image classification model of VGG-19-layer, which is trained using a custom Caffe-based framework, and its specific network structure can also be found in the following link for reference: https://gist.github.com/ksimonyan/3785162f95cd2d5fee77#file-readme-md

5. CNN based tracking framework

As shown in Fig. 2, we give an overview of the proposed tracking framework. Our main contributive components are marked with red color. For more discriminative object representation, we employ the convolutional features trained from CNN as extracted feature. In order to retain more spatial resolution on each convolutional layer, we use only the outputs of the conv3-4, conv4-4, conv5-4 convolutional layers for features representation. For more accurate object prediction purpose, we use the multiple correlation filters on multiple layers to compute the response maps, which can take the advantages of both robustness and accuracy. Details of the initial object location prediction are illustrated in [17]. Although the CNN based feature has demonstrated its efficiency in online tracking, however, due to unpredictable target appearance changes and complex scene variances, the initial object still need to be refined to alleviate the drift problem.

Accurate object regions prediction is very important for online tracking, however, it is difficult to estimate exact object scale changes during online tracking process because of complex appearance changes. For more accurate object regions prediction, we incorporate robust motion estimation into the tracking framework to extract salient moving regions, which could generate qualified samples for online learning and alleviate the drift problem. If the target moves sufficient distance, the moving regions indicate the location of the target. Due to the pixel displacements within the target regions are unpredictable, it is difficult to set a fixed threshold to extract the moving target regions. We use the saliency object detection on the motion map (Fig. 3 (c)) for salient motion estimation, and the adaptive threshold is used to extract the moving regions on the salient motion map (Fig. 3 (d)). To adapt the target scale variation, we extend the initial object location with a small margin (10 pixels in our experiment) to a larger region as the candidate target regions. When only partial of the object regions are moving sufficient distance, the extracted moving regions would be incomplete compared to the whole target regions. Therefore, we incorporate an optional strategy for more accurate object regions prediction purpose. If the overlap ratio of the initial object location ((Fig. 3 (e)), green bounding box) and salient moving regions ((Fig. 3 (d)), red bounding box) is larger than a given threshold, the salient moving regions will be regarded as the final tracking result. Based on the refined object regions prediction, qualified samples are generated for online model up date.

6. Experiments and discussions

The proposed tracking is implemented by referring state-of-the-art tracking framework [17]. In this framework, the convolutional features are extracted based on VGG-Net-19 deep model structure which is trained on ImageNet dataset [39]. To guarantee a robust motion estimation between two consecutive frames, we employed motion flow based on scale invariant feature matching calculation [24] to generate the motion response maps. As an essential guidance for target localization, we incorporate a quantum mechanics based saliency detection with the motion model to achieve more accurate sampling region for online learning. In this process, target areas inside the obtained color motion maps are segmented by calculating the average displacement differences between the candidate regions, and the whole image is measured by using a pre-defined threshold, which is based on the assumption that the movement of the target object is usually sufficient.

The proposed tracking is mainly implemented with MATLAB, and all of the experiments are tested on a workstation with 3.2 GHz Intel i7 processor and 8 GB RAM. For testing video frame of size 640 × 360 without any coding optimization, the average speed of the proposed tracking is 4–5 fps. The highest computational cost part of the proposed tracking is the motion flow computation in [24], which has occupied the nearly 70–80 percent computational cost of the whole algorithm. Since it has been claimed in [24] that its speed can be intensively enhanced more

Fig. 3. Salient moving regions extraction. (a) frame k (b) frame $k+1$ (c) dense optical flow (d) salient motion extraction (e) refined tracking result (original object location is marked in green, final tracking result is marked in red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Average VOR comparison results of 10 different trackers on **NON-RIGID** objects dataset [32]. Symbol '–' indicates that the corresponding tracker can not finish the tracking task completely.

Video	Ours	Struck	HT	MEEM	STC	LSST	MUSTer	DAT	KCF	CFT
Cliff-Dive1	**0.70**	**0.65**	0.62	0.36	0.53	0.57	0.64	0.61	0.50	0.61
Cliff-Dive2	**0.41**	0.34	**0.43**	0.28	0.18	0.27	0.16	0.36	0.13	0.34
Diving	**0.40**	**0.36**	0.13	0.14	0.17	0.15	0.22	0.20	0.23	**0.40**
Gymnastics	0.50	**0.56**	0.14	0.19	0.49	0.49	0.50	0.07	**0.59**	**0.59**
High_Jump	0.14	0.17	**0.50**	0.24	0.12	0.06	0.10	0.05	0.21	**0.31**
Motocross1	**0.65**	0.32	0.54	0.10	0.09	0.10	0.24	0.10	0.12	**0.56**
Motocross2	0.68	**0.74**	0.58	0.64	**0.76**	0.57	0.71	0.40	0.70	0.70
Mountain-Bike	**0.83**	0.69	0.54	0.67	0.59	0.35	**0.72**	0.42	0.65	0.67
Skiing	**0.33**	0.05	**0.30**	**0.33**	-	0.06	0.05	0.28	0.11	**0.33**
Transformer	**0.67**	0.60	**0.65**	0.54	0.36	0.45	0.62	0.45	0.48	0.61
Volleyball	**0.71**	**0.38**	0.31	0.33	0.22	**0.38**	0.20	0.37	0.37	0.37
Average	**0.55**	0.44	0.43	0.35	0.35	0.32	0.38	0.30	0.37	**0.50**

than 10–20 times faster if implemented using CUDA based parallel programming, therefore, the speed of the proposed tracking could be further enhanced for the realistic applications.

For the parameter configuration of the proposed tracking, the hierarchical CNN based tracking model utilized the parameter setting as what is in [17]. For motion flow maps generation, the optimized parameters recommended by Wu et al. [24] is also employed into our work. To accurately estimate the salient motion regions from motion flow maps, we utilized the default parameters in [21] with fine-tuning adjustment on the whole video frame as discussed in proposed work section. In final process of spatial-temporal localization with optional strategy, the balance threshold θ is set to 0.5 in most testing videos of our experiments.

6.1. Tracking evaluation metrics

In order to demonstrate the performance of the proposed tracking, we adopt the PASCAL VOC Overlap Ratio (VOR) and Center Location Error (CLE) as the quantitative evaluation metrics. Let b_t denote the bounding box of the tracking result, b_g denote the bounding box of ground truth, the VOR value can be computed as $vor = \frac{b_t \bigcap b_g}{b_t \bigcup b_g}$. In our experiments, we use the average VOR and CLE values for fair performance evaluation, and the overlap thresholds are specified from 0 to 1 with step 0.05, and the center location thresholds are specified from 0 to 50 with step 2.5.

6.2. Tracking evaluation for non-rigid objects

We firstly evaluate the proposed tracking on the non-rigid objects benchmark dataset [32] which contains 11 challenging videos with drastic non-rigid objects's deformation, rotation, fast motion, scale variation and background clutter. Different from the most used bounding box based annotation, the ground truth of the dataset [32] includes two aspects. The first one is a binary segmentation mask based annotation, and the second one is a tight bounding box from segmentation mask. Due to complex target appearance changes and complex scene, most of the existing state-of-art trackers [4] usually drifts and performs poor on this dataset.

6.3. Quantitative evaluation

To quantitatively demonstrate the performance improvement, We also compared the proposed tracking with other 9 state-of-art trackers including Struck [7], HT [33], MEEM [34], STC [35], LSST [36], MUSTer [37], DAT [38], KCF [8] and CFT [17] by using VOR and CLE as the measurements. Another reason is, all of the STC, MUSTer, DAT, KCF and CFT are kernelized correlation filter based tracking strategies, which are more suitable for the comparison with the proposed tracking.

Tables 1and2 demonstrate the average VOR and CLE of the compared algorithms on 11 video sequences respectively. Overall, the proposed tracking outperforms favorably against the compared state-of-art trackers in both VOR and CLE metrics. Besides, we present the success plots and precision plots of the compared algorithms to demonstrate the overall performance of these trackers with different thresholds, as shown in Fig. 4. Robust feature extraction is very important in online tracking, since the high dimension HOG feature is used in KCF and MUSTer, they perform better than the raw pixel feature based tracker STC and color based tracker DAT. Based on the more robust and accurate CNN feature, CFT achieves better tracking result compared to other state-of-art trackers. In order to adapt the scale changes during online tracking, STC, MUSTer and DAT adopt the scale adaption scheme based on confidence map to adapt the target scale variation. However, due to complex deformation and background clutter, such sale estimation methods often fail to adapt the target appearance. By incorporating the salient moving regions extraction and robust CNN feature into tracking framework, the proposed tracking achieves the best average performance on the evaluated dataset. However, due to partial movement or inaccurate optical flow computation, which may degrade the accuracy of the tracking outputs, such as the tracking results of video *gymnastics, high_jump* and *motocross2*.

Table 2
Average CLE comparison results of 10 different trackers on **NON-RIGID** objects dataset [32]. Symbol '–' indicates that the corresponding tracker can not finish the tracking task completely.

Video	Ours	Struck	HT	MEEM	STC	LSST	MUSTer	DAT	KCF	CFT
Cliff-Dive1	**13.9**	18.7	**14.2**	56.0	18.1	32.6	21.4	18.1	45.6	21.2
Cliff-Dive2	**16.5**	17.8	**11.8**	35.9	104.8	34.1	75.7	18.7	93.2	22.8
Diving	**23.0**	30.9	65.1	92.9	82.7	87.8	73.6	69.9	67.9	**22.9**
Gymnastics	14.9	17.7	103.9	66.1	17.9	18.5	22.9	137.8	**14.5**	**14.3**
High_Jump	73.8	104.1	**13.0**	48.5	62.5	141.8	89.6	133.5	63.1	**26.3**
Motocross1	**12.2**	64.1	25.8	170.9	221.1	142.4	139.1	152.2	191.5	**11.6**
Motocross2	13.9	**9.7**	18.5	18.0	**8.3**	25.6	10.7	55.7	13.6	13.1
Mountain-Bike	**3.0**	9.2	9.5	10.5	**7.1**	134.6	8.0	67.6	8.1	9.1
Skiing	**9.0**	272.0	10.7	9.8	-	260.9	249.5	39.7	195.3	**8.9**
Transformer	**35.8**	37.1	**33.0**	67.5	47.5	96.2	47.2	66.2	81.2	38.4
Volleyball	**7.7**	88.9	105.6	95.1	105.0	91.2	114.1	**88.2**	90.2	88.5
Average	**20.3**	60.9	37.4	61.0	67.5	96.9	77.4	77.1	78.6	**25.2**

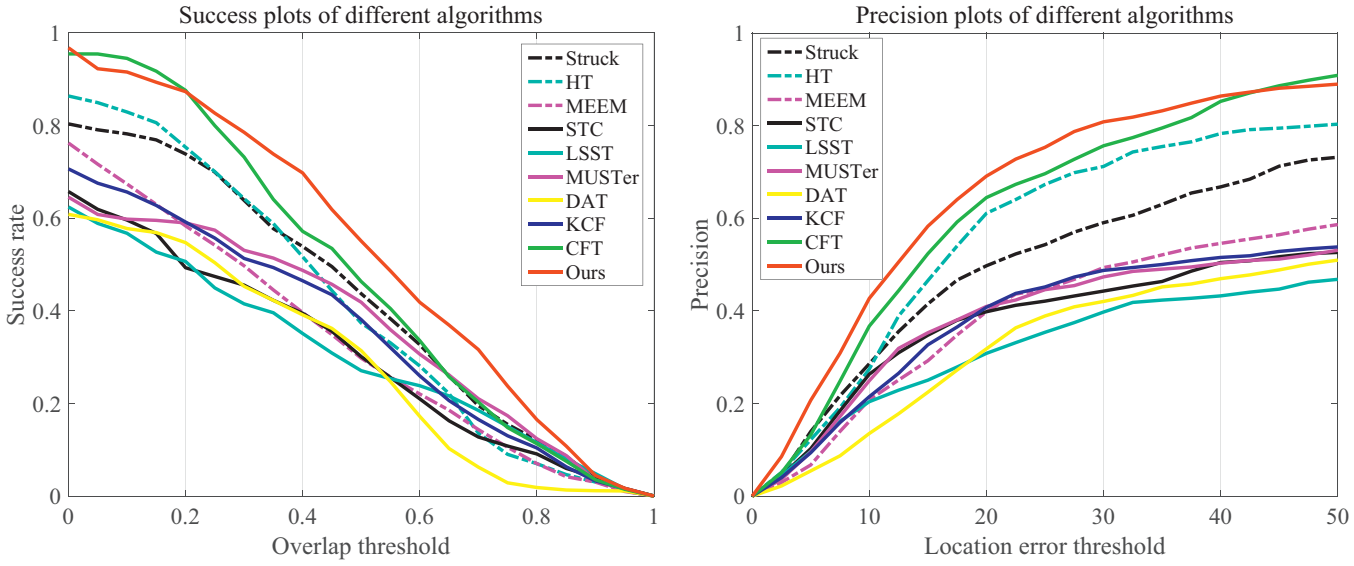


Fig. 4. Success plots and precision plots of 10 different trackers on **NON-RIGID** objects dataset [32].

Table 3
Average VOR comparison results of 16 different trackers on general benchmark dataset containing **ARBITRARY** objects [4].

Sequence	Ours	CT	CSK	LOT	DFT	VTS	Struck	LSK	CXT	VTD	TLD	MIL	STC	LSST	DAT	CFT
Boy	**0.84**	0.42	0.72	0.47	0.72	0.42	0.70	0.66	0.35	0.67	0.66	0.65	0.58	0.67	0.73	**0.83**
Car4	**0.84**	0.14	0.32	0.15	0.19	0.36	0.39	0.28	0.16	0.36	0.19	0.21	0.40	0.24	0.01	**0.54**
CarScale	**0.76**	0.48	**0.52**	0.26	0.48	0.48	0.50	**0.52**	0.45	0.47	0.47	**0.52**	0.50	0.44	0.46	0.47
Couple	**0.72**	0.05	0.31	0.49	0.08	0.35	0.55	0.07	0.06	0.08	0.32	0.52	0.08	0.10	0.45	**0.62**
Crossing	**0.79**	0.57	0.56	0.28	0.45	0.34	0.28	0.33	0.44	0.34	0.45	0.62	0.27	0.45	0.71	**0.76**
David	**0.72**	0.31	0.39	0.31	0.43	0.47	0.43	0.59	0.57	0.46	**0.64**	0.30	0.57	0.17	0.52	0.59
David3	**0.85**	0.23	0.48	0.50	0.59	0.25	0.71	0.46	0.09	0.18	0.05	0.29	0.49	0.60	0.74	**0.83**
Deer	**0.85**	0.03	0.59	0.17	0.58	0.05	0.61	0.35	0.39	0.06	0.47	0.45	0.04	0.05	0.13	**0.79**
Dog1	**0.83**	0.65	0.65	0.41	0.57	0.62	0.66	0.65	**0.69**	0.60	0.54	0.59	0.54	0.60	0.14	0.60
Faceocc2	**0.83**	0.57	0.63	0.52	0.62	0.74	0.63	0.56	0.71	0.74	0.62	0.56	0.74	0.75	0.32	**0.81**
Fish	**0.89**	0.68	0.77	0.37	0.74	0.64	0.56	0.54	0.69	0.60	0.68	0.54	0.56	**0.90**	0.10	0.88
Football	**0.77**	0.46	0.49	0.47	0.54	0.60	0.45	0.51	0.50	**0.61**	0.44	0.49	0.55	0.40	0.05	**0.77**
Mhyang	**0.88**	0.48	0.80	0.45	0.34	0.84	0.79	0.70	0.81	0.80	0.82	0.64	0.74	**0.86**	0.64	**0.86**
MountainBike	**0.80**	0.12	**0.78**	0.54	**0.78**	0.36	0.69	0.10	0.10	0.51	0.12	0.15	0.64	0.50	0.11	0.75
Singer1	**0.89**	0.31	0.31	0.13	0.04	0.43	0.31	0.08	0.56	0.43	**0.57**	0.21	0.58	0.19	0.39	0.40
Singer2	0.05	0.14	0.05	0.08	0.26	0.04	0.05	0.04	0.06	0.04	0.03	0.05	**0.45**	**0.56**	0.02	0.05
Subway	**0.79**	0.03	0.16	0.19	0.35	0.17	0.68	0.44	0.15	0.16	0.65	0.15	0.19	0.17	0.77	**0.83**
Sylvester	**0.68**	0.53	0.62	0.33	0.18	0.57	0.66	0.15	0.64	0.57	0.49	0.55	0.57	0.33	0.25	**0.71**
Trellis	**0.80**	0.35	0.29	0.33	0.29	0.41	0.48	0.22	0.56	0.45	0.52	0.35	0.52	0.24	0.62	**0.67**
Woman	**0.75**	0.14	0.18	0.10	0.57	0.06	0.72	0.14	0.11	0.06	0.32	0.14	0.40	0.16	0.14	**0.76**
Average	**0.77**	0.33	0.48	0.33	0.44	0.41	0.54	0.37	0.40	0.41	0.45	0.40	0.47	0.42	0.37	**0.68**

6.3.1. Qualitative evaluation

For qualitative evaluation, we present some comparison results of the proposed tracking and some state-of-art trackers, as shown in Fig. 7. To show the comparison results clearly, we mainly choose the kernelized correlation filter based trackers, which include STC, MUSTer, DAT, KCF and CFT. Besides, another scale adaption tracker LSST is also included to demonstrate the complexity of the scale estimation. By incorporating the salient moving regions extraction and robust tracking output refined strategy, the proposed tracking algorithm is able to adapt complex deformation, rotation and scale variation. Such as in video *cliff-dive1, motocross1, mountain-bike* and *transformer*, the proposed algorithm performs better than CFT and other compared trackers. Especially for the target in video *volleyball*, all of the compared trackers are drift except the

Table 4
Average CLE comparison results of 16 different trackers on general benchmark dataset containing **ARBITRARY** objects [4].

Sequence	Ours	CT	CSK	LOT	DFT	VTS	Struck	LSK	CXT	VTD	TLD	MIL	STC	LSST	DAT	CFT
Boy	2.7	34.1	3.2	49.5	**2.4**	46.4	3.5	**2.2**	28.1	7.8	3.6	6.3	25.9	17.4	6.1	2.8
Car4	**5.4**	92.0	41.1	114.9	108.4	37.4	**11.8**	42.5	72.4	33.2	84.4	59.5	10.7	120.8	155.5	6.9
CarScale	15.4	73.8	27.8	97.4	74.3	37.9	37.0	**12.2**	34.8	34.3	14.2	**11.5**	55.8	69.5	20.1	14.7
Couple	**6.8**	101.7	73.3	39.3	112.4	42.7	32.3	112.2	102.0	104.0	35.6	37.0	123.1	100.8	34.9	**7.7**
Crossing	**2.5**	7.3	8.9	91.5	23.8	42.5	72.3	45.8	18.7	42.9	22.4	7.0	34.1	45.1	4.9	**2.7**
David	8.4	31.5	16.9	30.8	14.3	16.0	7.3	**5.5**	**4.2**	15.9	12.5	39.8	11.8	128.6	13.7	7.7
David3	**3.8**	85.5	58.9	8.7	56.4	103.7	8.8	72.3	224.7	180.6	208.8	102.3	6.3	54.4	14.2	**3.6**
Deer	**5.1**	233.9	6.8	80.9	9.7	222.9	7.7	90.9	25.3	219.2	20.3	30.3	402.0	201.3	143.9	**5.5**
Dog1	5.0	8.4	**3.6**	36.2	19.9	16.6	4.9	6.7	**2.8**	17.5	4.7	11.8	-	8.2	65.4	4.7
Faceocc2	**6.3**	20.4	7.5	19.5	10.3	7.5	**6.3**	15.8	7.3	7.5	18.2	21.9	10.1	11.0	46.7	**7.1**
Fish	4.3	15.2	8.4	38.6	6.4	12.5	22.2	20.1	8.8	15.7	12.7	19.9	**4.0**	**3.3**	88.6	4.3
Football	**4.3**	15.7	15.5	12.2	9.0	5.9	19.5	14.3	12.5	**4.7**	12.6	11.8	16.1	63.6	189.3	**4.7**
Mhyang	3.8	27.2	3.4	36.2	46.9	3.5	2.8	9.6	2.9	4.1	5.8	14.3	4.5	**2.7**	14.8	**2.6**
MountainBike	7.6	215.7	**6.8**	29.1	**6.5**	115.8	10.2	221.7	209.6	42.0	128.9	201.7	7.0	83.1	223.8	7.8
Singer1	**3.1**	14.3	23.5	164.3	212.6	12.8	11.0	114.7	26.9	9.1	26.6	44.1	**5.8**	132.3	35.5	8.7
Singer2	177.0	116.7	170.1	155.7	75.9	178.7	173.4	173.9	194.8	178.7	189.1	161.9	**52.8**	**42.9**	246.5	183.9
Subway	**2.7**	151.1	172.9	147.6	58.9	143.5	5.1	28.0	141.3	143.9	6.5	149.8	-	135.3	4.9	**2.9**
Sylvester	12.5	14.9	**8.4**	29.2	87.4	19.0	**5.5**	78.4	13.4	19.2	15.5	15.8	9.45	64.08	96.10	12.95
Trellis	**4.7**	40.1	65.6	41.6	43.5	35.0	12.4	72.6	20.2	31.5	15.2	38.7	32.0	103.3	13.3	**6.4**
Woman	9.7	121.1	215.4	141.7	**7.8**	152.6	**2.2**	85.0	131.0	116.5	150.0	121.7	21.5	119.8	131.0	8.6
Average	**14.6**	71.0	46.9	68.2	49.3	62.6	22.8	61.2	64.0	61.4	49.4	55.4	46.3	75.4	77.5	**15.3**

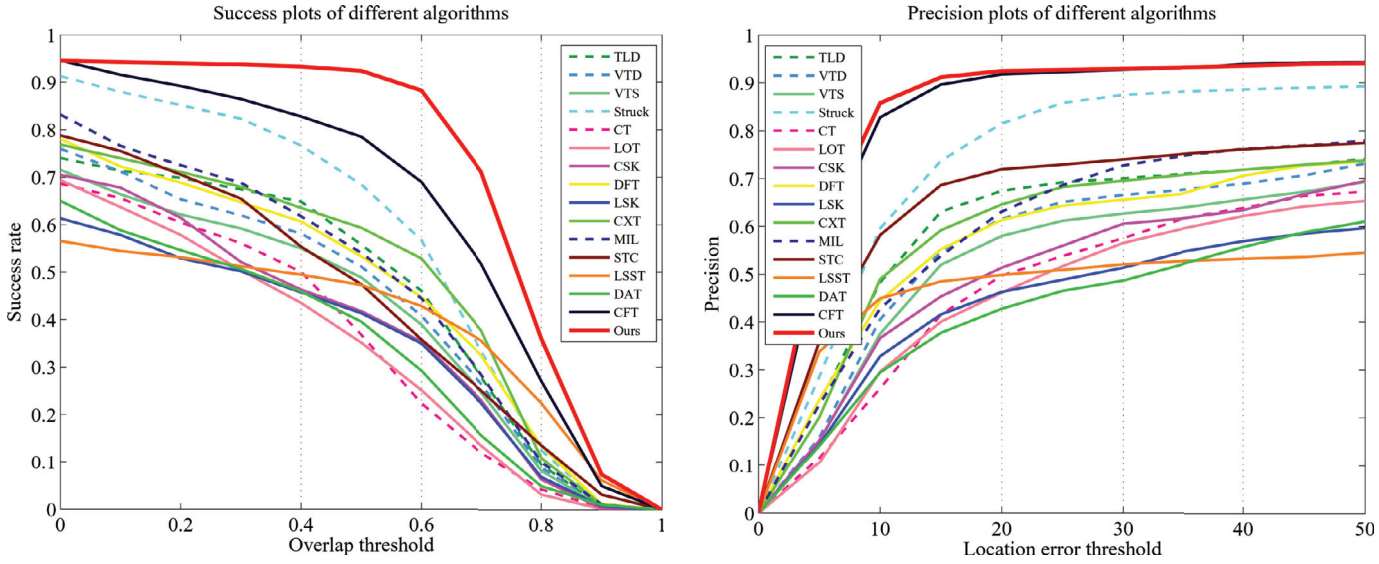


Fig. 5. Success plots and precision plots of 12 different trackers on general dataset containing **ARBITRARY** objects [4].

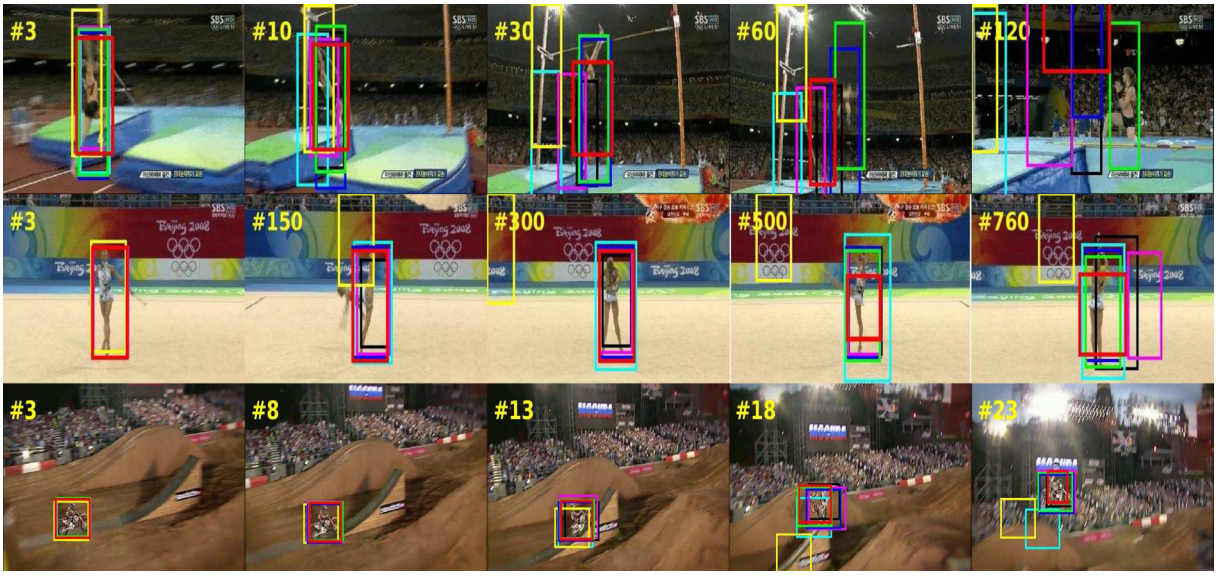


Fig. 6. Some failure cases of the proposed algorithm. STC, LSST, MUSTer, DAT, KCF, CFT and Ours. From top to bottom, the video sequences are *high_jump*, *gymnastics* and *motocross2*.

Fig. 7. Representative tracking results on **NON-RIGID** objects dataset [32]. The comparison trackers contain STC, LSST, MUSTer, DAT, KCF, CFT and Ours. From top to bottom, the video sequences are *cliff-dive1, cliff-dive2, diving, motocross1, mountain-bike, skiing, transformer* and *volleyball*.

proposed algorithm. Due to partial movement and complex scene, it is difficult to predict accurate target regions by salient moving regions extraction, but the proposed tracker still achieves good performance on videos *gymnastics* and *skiing* as CFT.

6.4. Tracking evaluation for arbitrary objects

To verify the tracking performance for arbitrary objects, we need to evaluate the proposed tracking on more general benchmark dataset [4], which contains a variety of objects (*e.g.*, face, pedestrian, sporter, car, toy) and challenging scenarios including scale and illumination change, occlusion, fast movement, out of plane rotation and background clutters. For comparison in both qualitative and quantitative aspects, we employed 15 representative state-of-the-art tracking approaches containing CT [46], CSK [47], LOT [48], DFT [49], VTS [50], Struck [7], LSK [51], CXT [52], VTD [53], TLD [54], MIL [55], STC [35],LSST [36], DAT [38] and CFT [17]

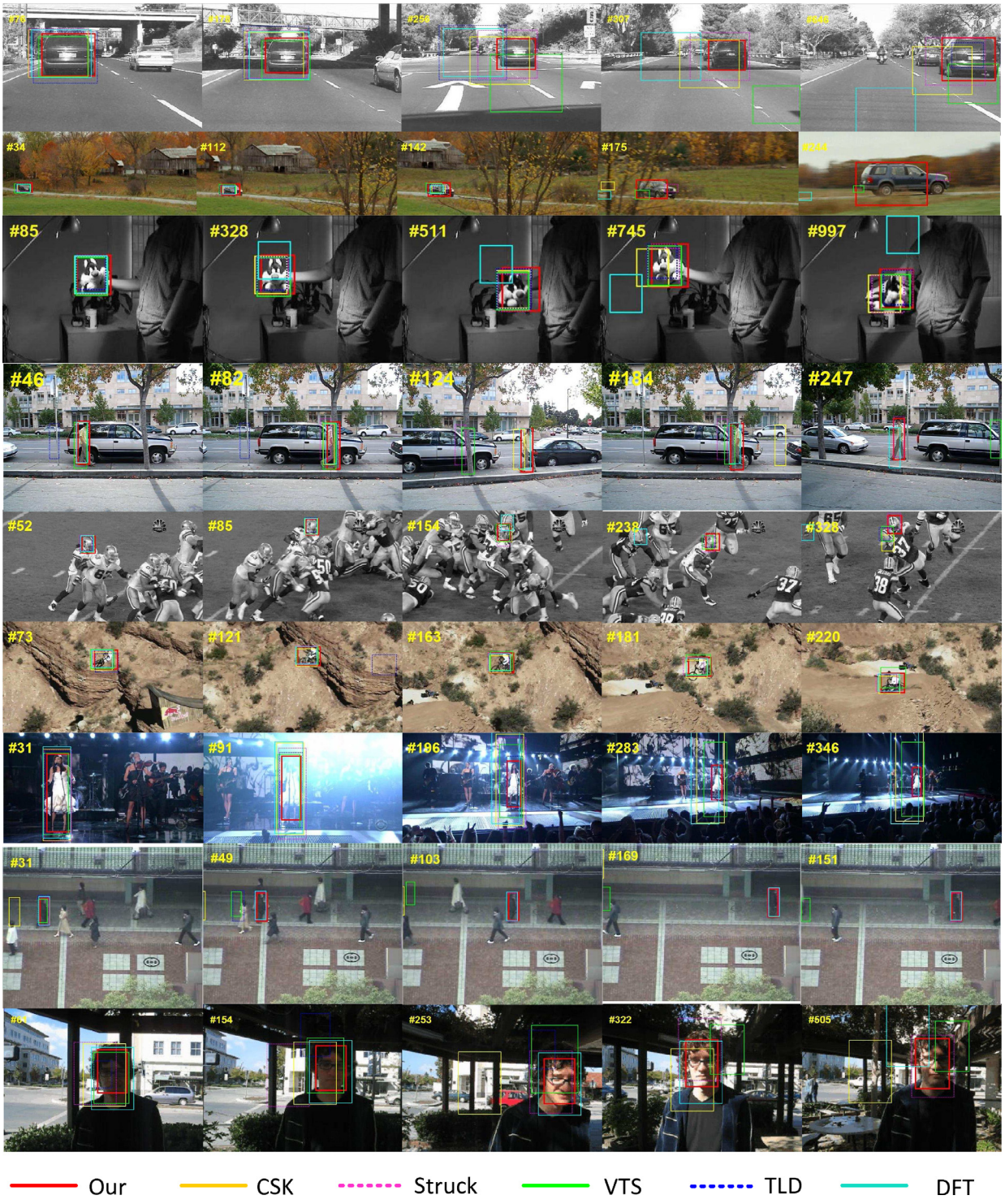


Fig. 8. Representative tracking results on **ARBITRARY** objects dataset [4]. The comparison trackers contain CSK, Struck, VTS, TLD, DFT and Ours. From top to bottom, the video sequences are *Car4, CarScale, Sylvester, David3, Football, MountainBike, Singer1, Subway* and *Trellis*.

6.4.1. Quantitative evaluation

The overall satisfactory performance shown in Tables 3 and 4 have verified that, the proposed tracking can achieve more robust and accurate performance a variety of scenarios including illumination change, scale variation, fast movement, deformation, background clutters, heavy occlusion, out of plane rotation etc. Furthermore, the success plots and precision plots in Fig. 5) are also shown to demonstrate the average performance of all the compared trackers. The thresholds of success plot are specified from 0 to 0.9 by a fixed step 0.1. For each threshold, the higher the plot, the more robust the tracking becomes. The thresholds of precision plot are specified from 10 to 50 by a fixed step 10, and the higher plot means higher accuracy as well.

6.4.2. Qualitative evaluation

We also present representative tracking results on more general benchmark dataset in Fig. 8. The qualitative comparison with the other state-of-the-art trackers still shows that the proposed

tracking is able to perform more accurately as well as robustly in different challenging scenarios:

- *Scale variation* as sequence '*CarScale*' which is shown in row #2 of Fig. 8, when the size of target gradually changes, the proposed tracking achieves a best performance among all the compared trackers. Although the happened partial occlusion gives trouble to the other trackers accordingly, with the help of more robust scale estimation, the proposed tracking is still able to obtain more accurate results.
- *Illumination variation* as sequences '*Car4*' and '*Trellis*', which are shown in row #1 and row #9 of Fig. 8, is another type of challenge which needs to be evaluated during experiments. The proposed tracking is not sensitive to illumination changes and outperforms other compared trackers. Especially for sequence '*Trellis*', the illumination variation on the target is very heavy, which means hard to obtain the motion information accurately. Thus in this testing, most of the compared trackers have gradually drifted from the target (such as CSK, DFT and VTS). Comparatively, even the color or intensity of the target is similar towards the background, the proposed tracking can also achieve a satisfactory performance in benefiting from the robust visual features learned with CNN.
- *Pose variation* as sequence '*Singer1*' which is shown in row #7 of Fig. 8. The target in sequence *Singer1* appears in rapid changing poses and scales. Moreover, there are drastic illumination changes. Most of the evaluated trackers perform poor on this sequence in scaling estimation, while the proposed tracking easily outperforms all the compared trackers because of the effective hierarchical feature output and learning with CNN.
- *Occlusion* as sequences '*David3*' and '*Football*' which are shown in row #4 and #5 of Fig. 8. There is heavy occlusion and out of plane rotation, the predicted target displacement by motion is hard to be guaranteed, which becomes the main reason of tracking drift such as VTS and CSK. With optional strategy for online sampling based on spatial-temporal localization, the proposed tracking is able to track the target more accurately in comparing with the other tracker which shared a similar online learning strategy such as DFT.
- *Clutter background* as sequence '*Mountainbike*' which is shown in row #6 of Fig. 8. The texture of the target gradually becomes difficult to be distinguished from the background during the movement with an out-of-plane rotation on-the-fly. Different trackers perform well on this sequence in the beginning but many failed in the end. Depending on a robust motion flow region estimated with scale and rotation invariant feature matching, the proposed tracking still achieve the best performance among all the tested trackers.

6.5. Limitation discussion

Due to unpredictable target appearance changes and complex scenes, it is hard to accurately segment the foreground from background during online tracking, which would inevitably weak the performance of the proposed tracking in some cases as shown in Fig. 6. For the target in video *high_jump*, the bar nearby the target is also moving, therefore worse tracking results are obtained by the salient moving regions extraction, and the proposed algorithm drifts in this video. The target appearances in video *gymnastics* are drastically changes, and partial movement of the target often happens in this video, salient moving regions may lead to inaccurate scale variation sometimes. In video *motocross2*, the shadow surrounding the target is wrongly regarded as the foreground, the performance of the proposed algorithm degrades a little in this condition.

7. Conclusion

In this paper, a deep learning based tracking strategy with CNN is proposed by incorporating the target's intra-frame appearance correlations and inter-frame motion saliency into a compositional learning framework. Compared with other existing CNN trackers only depending on the last layer output for target localization, the proposed approach is able to achieve more accurate localization for qualified sampling through formulating a couple energy optimization process of quantum mechanics based saliency detection and motion flow map generation. In addition, a kernelized appearance correlation is performed in frequency domain to enhance the efficiency of the whole methodology. With an optional functionality to define the final output, the proposed tracking has demonstrated its outperforming robustness and accuracy on the challenging tracking benchmark datasets.

Acknowledgment

This work is supported by the grants 61571362, 61363046, 61403182 approved by the National Natural Science Foundation, China, and the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centre in Singapore Funding Initiative, and the Young Talented Scientist Grant 20153BCB23029 approved by the Jiangxi Provincial Department of Science and Technology.

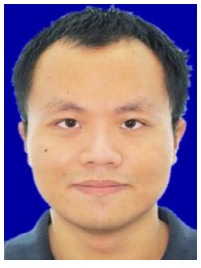
References

[1] D. Tao, X. Li, X. Wu, S.J. Maybank, General tensor discriminant analysis and gabor features for gait recognition, IEEE Trans. Pattern Anal. Mach. Intell. 29 (10) (2007) 1700–1715.
[2] S. Gu, Y. Zheng, C. Tomasi, Linear time offline tracking and lower envelope algorithms, Proceedings of the International Conference on Computer Vision, 2011.
[3] C. Xu, D. Tao, C. Xu, Multi-view intact space learning, IEEE Trans. Pattern Anal. Mach. Intell. 37 (12) (2015) 2531–2544.
[4] Y. Wu, J. Lim, M.-H. Yang, Object tracking benchmark, Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015.
[5] T. Liu, D. Tao, M. Song, S.J. Maybank, Algorithm-dependent generalization bounds for multi-task learning, Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016.
[6] P. Zhang, T. Zhuo, Y. Zhang, D. Tao, J. Cheng, Online tracking based on efficient transductive learning with sample matching costs, in: Neurocomputing, vol. 175, Elsevier, 2016, pp. 166–176. Part A, 29
[7] S. Hare, A. Saffari, P.H.S. Torr, Struck: Structured output tracking with kernels, Proceedings of the IEEE International Conference on Computer Vision, 2011.
[8] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 583–596.
[9] H. Jiang, J. Li, D. Wang, H. Lu, Multi-feature tracking via adaptive weights, in: Neurocomputing, Elsevier, 2016 http://dx.doi.org/10.1016/j.neucom.2016.03.074.
[10] L. Zhang, D. Bi, Y. Zha, S. Gao, H. Wang, T. Ku, Robust and fast visual tracking via spatial kernel phase correlation filter, in: Neurocomputing, Elsevier, 2016 http://dx.doi.org/10.1016/j.neucom.2015.10.131.
[11] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[12] Y. Guo, Y. Liu, R. Oerlemans, S. Lao, S. Wu, M.S. Lew, Deep learning for visual understanding: A review, in: Neurocomputing, Elesiver, 2016 http://dx.doi.org/10.1016/j.neucom.2015.09.116.
[13] N. Wang, D.-Y. Yeung, Learning a deep compact image representation for visual tracking, Proceedings of the Neural Information Processing Systems, 2013.
[14] H. Li, Y. Li, F. Porikli, Robust online visual tracking with a single convolutional neural network, Proceedings of the Asian Conference on Computer Vision, 2014.
[15] G. Wu, W. Lu, G. Gao, C. Zhao, J. Liu, Regional deep learning model for visual tracking, in: Neurocomputing, Elesiver, 2016 http://dx.doi.org/10.1016/j.neucom.2015.10.064.
[16] S. Hong, T. You, S. Kwak, B. Han, Online tracking by learning discriminative saliency map with convolutional neural network, Proceedings of the International Conference on Machine Learning, 2015.
[17] C. Ma, J.B. Huang, X. Yang, M.-H. Yang, Hierarchical convolutional features for visual tracking, Proceedings of the IEEE International Conference on Computer Vision, 2015.
[18] Y. Zhang, F. Zhang, L. Guo, Saliency detection by selective color features, in: Neurocomputing, Elesiver, 2016 http://dx.doi.org/10.1016/j.neucom.2016.04.005.

[19] D. Tao, X. Tang, X. Li, X. Wu, Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 28 (7) (2006) 1088–1099.
[20] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, Mech, Minimum barrier salient object detection at 80 FPS, Proceedings of the IEEE International Conference on Computer Vision, 2015.
[21] C. Aytekin, S. Kiranyaz, M. Gabbouj, Automatic object segmentation by quantum cuts, Proceedings of the IEEE International Conference on Pattern Recognition, 2015.
[22] Y.Y. Boykov, Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images, Proceedings of the IEEE International Conference on Computer Vision, 2015.
[23] R.L. Liboff, Introductory Quantum Mechanics, San Francisco: Addison Wesley, 2003.
[24] C. Liu, Beyond Pixels: Exploring New Representations and Applications for Motion Analysis, Ph.D. Thesis of Massachusetts Institute of Technology, 2009 Ph.D. thesis.
[25] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Proceedings of the International Conference on Learning Representations, 2015.
[26] L. Wang, T. Liu, G. Wang, K.L. Chan, Q. Yang, Video tracking using learned hierarchical features, IEEE Trans. Image Process. 24 (4) (2015) 1424–1435.
[27] H. Li, Y. Li, F. Porikli, Deeptrack: Learning discriminative feature representations by convolutional neural networks for visual tracking, British Machine Vision Conference (BMVC) (2014).
[28] M.-M. Cheng, N.J. Mitra, X. Huang, P.H.S. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 569–582.
[29] T. Liu, D. Tao, Classification with noisy labels by importance reweighting, IEEE Trans. Pattern Anal. Mach. Intell. 38 (3) (2016) 447–461.
[30] W. Wang, J. Shen, F. Porikli, Saliency-aware geodesic video object segmentation, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2015.
[31] R.M. Gray, Toeplitz and Circulant Matrices: A Review, Now Publishers, 2006.
[32] J. Son, I. Jung, K. Park, B. Han, Tracking-by-segmentation with online gradient boosting decision tree, Proceedings of the IEEE International Conference on Computer Vision, 2015.
[33] M. Godec, P.M. Roth, H. Bischof, Hough-based tracking of non-rigid objects, Proceedings of the IEEE International Conference on Computer Vision, 2011.
[34] J. Zhang, S. Ma, S. Sclaroff, Meem: robust tracking via multiple experts using entropy minimization, Proceedings of the European Conference on Computer Vision, 2014.
[35] K. Zhang, L. Zhang, Q. Liu, D. Zhang, M.-H. Yang, Fast visual tracking via dense spatio-temporal context learning, Proceedings of the European Conference on Computer Vision, 2014.
[36] D. Wang, H. Lu, M.-H. Yang, Least soft-threshold squares tracking, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2013.
[37] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, D. Tao, Multi-store tracker (MUSTer): a cognitive psychology inspired approach to object tracking, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2015.
[38] H. Possegger, T. Mauthner, H. Bischof, In defense of color-based model-free tracking, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2015.
[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, F.-F. L., Imagenet: a large-scale hierarchical image database, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2009.
[40] D. Tao, C. Hong, J. Yu, M. Wang, Multimodal deep autoencoder for human pose recovery, Proceedings of the IEEE Transactions on Image Processing, 2015.
[41] J. Xua, X. Luoa, G. Wanga, H. Gilmoreb, A. Madabhushi, A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images, in: Neurocomputing, vol. 191, Elesiver, 2016, pp. 214–223.
[42] S.S. Liewa, M. Khalil-Hania, R. Bakhterib, Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems, in: Neurocomputing, Elesiver, 2016 http://dx.doi.org/10.1016/j.neucom.2016.08.037.
[43] C. Hong, J.K. Zhu, J. Yu, J. Cheng, X. Chen, Realtime and robust object matching with a large number of templates, in: Multimedia Tools and Applications, vol. 75(3), Springer, 2016, pp. 1459–1480.
[44] J. Yu, X. Yang, F. Gao, D. Tao, Deep multimodal distance metric learning using click constraints for image ranking, IEEE Trans. Cybern. (2016), doi:10.1109/TCYB.2016.2591583.
[45] J. Yu, D. Tao, Y. Rui, Learning to rank using user clicks and visual features for image retrieval, IEEE Trans. Cybern. 45 (4) (2015) 767–779.
[46] K. Zhang, L. Zhang, M.-H. Yang, Real-time compressive tracking, Proceedings of the European Conference on Computer Vision, 2012.
[47] F. Henriques, R. Caseiro, P. Martins, J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, Proceedings of the European Conference on Computer Vision, 2012.
[48] S. Oron, A. Bar-Hillel, D. Levi, S. Avidan, Locally orderless tracking, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2012.
[49] L. Sevilla-Lara, E. Learned-Miller, Distribution fields for tracking, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2012.
[50] J. Kwon, K.M. Lee, Tracking by sampling trackers, Proceedings of the IEEE International Conference on Computer Vision, 2011.
[51] B. Liu, J. Huang, L. Yang, C. Kulikowsk, Robust tracking using local sparse appearance model and k-selection, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2011.
[52] T.B. Dinh, N. Vo, G. Medioni, Context tracker: Exploring supporters and distracters in unconstrained environments, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2011.
[53] J. Kwon, K.M. Lee, Visual tracking decomposition, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2010.
[54] Z. Kalal, J. Matas, K. Mikolajczyk, Tracking-learning-detection, Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, 6, 2010, pp. 1409–1422.
[55] B. Babenko, M.-H. Yang, S. Belongie, Visual tracking with online multiple instance learning, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2009.
[56] J. Yu, Z. Kuang, B. Zhang, D. Lin, J. Fan, Image Privacy Protection by Identifying Sensitive Objects via Deep Multi-Task Learning, IEEE Transactions on Information Forensics and Security(T-IFS), 2016 doi:10.1109/TIFS.2016.2636090. doi:10.1109/TCYB.2016.2591583.
[57] J. Yu, Z. Kuang, F. Gao, D. Tao, Deep Multimodal Distance Metric Learning Using Click Constraints for Image Ranking, IEEE Transactions on Cybernetics (T-CYB), 2016.

Dr. Zhang received the B.E. degree from the Xian Jiaotong University, China in 2001. He received his PhD from Nanyang Technological University, Singapore in 2011. Currently, he is an associate professor in School of Computer Science, Northwestern Polytechnical University, China. His current research interests include object detection and tracking, computer vision and pattern recognition. He has published more than 40 high ranked international conference and journal papers. He is a member of ACM.

Dr. Tao Zhuo received the B.S. degree in Computer Science and Technology from the Xi'an Shiyou University, Xi'an, China, in 2009, and the master's degree and Ph.D. degree in Computer Science and Technology from Northwestern Polytechnical University, Xi'an, China, in 2012 and 2016 respectively. Currently, he is a research fellow at the Sensor-enhanced Social Media (SeSaMe) Centre in the Interactive and Digital Media Institute, National University of Singapore. His research interests include visual object tracking, machine learning and computer vision.



Dr. Wei Huang obtained his B.Eng and M.Eng degrees from Harbin Institute of Technology in 2004 and 2006, respectively. He obtained his Ph.D. degree from Nanyang Technological University, Singapore, in 2011. Before joining Nanchang University as an Associate Professor, he worked in University of California San Diego as well as Agency for Science Technology and Research as Research Associate and Research Fellow, respectively. Dr Huang's recent research interests mainly include but not limited to computer vision, medical image computing, information security, and signal processing.

Kangli Chen received the B.S.degree in Information and Computing Science from the Taiyuan University of Technology, China,in 2013. Currently, she is a postgraduate student in School of Computer Science, Northwestern Polytechincal University, China. She has experienced in NUS from 2015–2016 as an international intern student program in the Centre in the Interactive and Digital Media Institute (SeSaMe), National University of Singapore. Her current research interests include object tracking and detection, machine learning and computer vision.

Prof. Mohan Kankanhalli is the Provost's Chair Professor at the Department of Computer Science of the National University of Singapore. He is also the Vice Provost for Graduate Education at NUS. Mohan obtained his BTech from IIT Kharagpur and MS & PhD from the Rensselaer Polytechnic Institute. His current research interests are in Multimedia Computing, Multimedia Security, Image and Video Processing and Social Media Analysis. He was awarded a large grant by Singapore's National Research Foundation to set up the SeSaMe Centre. Mohan is very active in the Multimedia research community. He was the ACM SIGMM Director of Conferences from 2009 to 2013. He is on the editorial boards of several journals, including ACM Transactions on Multimedia Computing, Communications, and Applications; Springer Multimedia Systems Journal; and Multimedia Tools and Applications Journal. Mohan is a Fellow of IEEE.