Spiral visual and motional tracking

Xiao Yun[a,*], Gang Xiao[b]

[a] *China University of Mining and Technology, School of Information and Electrical Engineering, 1 Daxue Road, Xuzhou, Jiangsu 221116, China*
[b] *Shanghai Jiao Tong University, School of Aeronautics and Astronautics, 800 Dongchuan Street, Shanghai 200240, China*

Abstract

Constructing a visual appearance model is essential for visual tracking. However, relying only on the visual model during appearance changes is insufficient and may even interfere with achieving good results. Although several visual tracking algorithms emphasize motional tracking that estimates the motion state of the object center between consecutive frames, they suffer from accumulated error during runtime. As neither visual nor motional trackers are capable of performing well separately, several groups have recently proposed simultaneous visual and motional tracking algorithms. However, because tracking problems are often NP-hard, these algorithms cannot provide good solutions for the reason that they are driven top-down with low flexibility and often encounter drift problems. This paper proposes a spiral visual and motional tracking (SVMT) algorithm which, unlike existing algorithms, builds a strong tracker by cyclically combining weak trackers from both the visual and motional layers. In the spiral-like framework, an iteration model is used to search for the optimum until convergence, with the potential for achieving optimization. Three learned procedures including visual classification, motional estimation, and risk analysis are integrated into the generalized framework and implement corresponding modifications with regard to their performances. The experimental results demonstrate that SVMT performs well in terms of accuracy and robustness.

Keywords: visual tracking; motional tracking; simultaneous visual and

[☆] Fully documented templates are available in the elsarticle package on CTAN.

[*] Corresponding author
Email address: d.xyun@outlook.com (Xiao Yun)

motional tracking; iteration model; risk function

1. Introduction

Visual tracking has received widespread attention for its extensive applications in computer vision such as intelligent video surveillance, human-machine interfaces, robotics, and motion analysis [1, 2]. The construction of a visual appearance model is essential for visual tracking. Depending on the appearance model, existing tracking algorithms can be categorized into two categories: generative [3, 4] and discriminative [5, 6]. Generative tracking algorithms build a target model and then search for the candidate image patch with maximal similarity; for example, the l_1-tracker [3], which uses sparse representation to model the target, and which was subsequently extended by Li et al. [4] through an orthogonal matching pursuit algorithm. Zhang et al. [7] proposed a generic formulation of the l_1-tracker named multi-task sparse learning method, in which tracking is formulated in a particle filter framework and the global and local structural appearance correlations between particles is exploited in [8]. In [9], a structural sparse tracking algorithm not only exploits the intrinsic relationship among target candidates and their local patches to learn their sparse representations jointly, but also preserves the spatial layout structure among the local patches inside each target candidate. The part matching tracker [10] with the spatial-temporal locality-constrained property achieves robust visual tracking by considering both local (i.e., the low-rank and sparse structure information) and global (i.e., multi-mode template updating) matchings. A consistent low-rank sparse tracker [11] was proposed in which the low-rank nature underlying the image observation is exploited and the temporal consistency between the representation of the selected candidate particles is taken into account. On the contrary, discriminative algorithms cast tracking as a classification task which separates the target foreground from the background. Babenko et al. [5] introduced multiple instance learning (MIL) into online tracking where samples are considered as positive and negative bags or sets. In the online discriminative feature selection (ODFS) tracking method [6], the classifier score is explicitly coupled with sample importance and its objective function is optimized by feature selection. Besides these efforts, other researchers have worked on tracking methods that are both generative and discriminative. For instance, in the fast compressive tracking (FCT) algorithm [12], the object

is represented by features extracted in the compressive domain, and these features are used to distinguish the foreground from the background. It is a simple yet efficient algorithm that combines the merits of both generative and discriminative algorithms. Motivated by FCT, Song et al. [13] took into account both appearance and spatial layout information in the projections and further proposed an online informative feature selection approach via maximizing entropy energy, which can select the most informative features from the pool. However, when the object changes its appearance (i.e., because of background clutter, illumination changes, occlusion, etc), depending only on visual appearance is insufficient or may even interfere with achieving good tracking.

Meanwhile, several previous works solved the problem of visual tracking from the perspective of motional tracking [14], in which the object is represented as a point and its motion state is estimated between consecutive frames. Motional trackers range from the Kalman filtering (KF) [15] technique to the Meanshift algorithm [16], particle filtering method [17], Markov chain Monte Carlo schemes [18], and more. The Kalman filter, which has been extensively used in dynamic systems, aims to estimate the optimal state of the tracked target from a series of measurements in an efficient computational way. Several research efforts [19–22] have applied the Kalman filter to visual tracking and have shown good improvement.

Visual and motional tracking are closely interrelated and there is no clear boundary between them. By operating simultaneously, these two methods can benefit from one another. An adaptive tracking framework [23] was presented to track non-rigid objects by fusing visual and motional feature descriptors. Features are extracted from different points of view and are used to update the object model for achieving tracking robustness. Cehovin et al. [24] developed a coupled-layer model that combines the local visual structure and the global motion of the target in a probabilistic model. Hua et al. [14] combined occlusion and motion reasoning methods via a tracking-by-detection approach which handles occlusion by integrating detection and motion models. Kalal et al. [25] presented a visual tracking framework consisting of tracking, learning, and detection (TLD) steps to achieve a long-term tracking task. An adaptive compressive tracking method [26] was proposed in which the most discriminative features are selected via an online vector boosting method and an effective trajectory rectification approach is adopted which can make the estimated location more accurate. Yang et al. [27] developed an online Fisher discrimination boosting feature selection mech-

anism which can enhance the discriminative capability between target and background and utilized a weighted particle filtering framework for visual tracking. In [28], the proposed convolutional networks based on the convolutional neural network have a lightweight structure and exploit local structural and inner geometric layout information from data without manual tweaking. Tracking can be formulated as the task of risk minimization, which is NP-hard, and may therefore cause numerical instability. However, the above algorithms are not able to provide good solutions because they are driven top-down which are limited by low flexibility and often encounter error accumulation and drift problems. In this paper, we propose a cyclical framework and use a risk modification model to address these problems.

A good tracker should be robust against challenges such as pose variation, illumination change, occlusion, etc. Although a large number of visual trackers have been reported in the literature, they still do not have sufficient capabilities to handle the above situations for the reason that the visual trackers independently locate the object in every frame and thereby are sensitive to a change in the visual appearance. To solve this problem, an effective method to estimate the motion state of the object from consecutive frames needs to be further explored. However, motional tracker follows the target frame to frame, thus suffering from accumulated error during runtime. Neither visual nor motional trackers can solve the tracking problem independently, but if they operate simultaneously, there is potential to improve the performance [25]. Motivated by it, this paper unifies visual and motional trackers into a generalized framework in which the visual part describes the target's visual properties and locates it in each frame, the motional part provides training data to update the visual results by the frame-to-frame information, and then the visual part re-initializes the motional results to prevent its tracking failure.

The main contributions of our work are summarized as follows:

1. In this paper, the spiral visual and motional tracking (SVMT) algorithm is proposed by using a cyclical process to build a strong tracker from initial weak trackers iteration by iteration. This iteration framework keeps searching for the optimum until it reaches convergence, and thereby has the potential to achieve optimization.

2. Because visual and motional trackers each have their individual advantages, SVMT combines them into a generalized tracking framework to make best use of strengths and avoid weaknesses.

3. The proposed risk function makes corresponding modifications with

respect to the processing errors from visual and motional layers. Therefore, three components including visual classification, motional estimation, and risk analysis are integrated into each iteration step, and keep learning for approaching optimization throughout the tracking process.

The remainder of this paper is organized as follows. In Section 2, we describe the proposed SVMT algorithm in detail. The experimental results are presented in Section 3. Section 4 concludes with a general discussion.

2. Problem Formulation

The proposed SVMT algorithm is described in detail in this section.

In this paper, the goal of tracking is to evaluate the risk function $R(\boldsymbol{x})$ by considering $\boldsymbol{x}$ as the tracking result, and to find the optimal $\boldsymbol{x}$ that minimizes $R(\boldsymbol{x})$. However, this problem is NP-hard and may cause numerical unstability. Therefore, we propose a spiral visual and motional tracking (SVMT) algorithm using an iteration process to solve this problem. In SVMT, each iteration step repeats the same procedures, and keeps searching for the optimum until convergence is reached. If we describe a single iteration step as a circle, and a later iteration step as a smaller circle, SVMT can be represented as a spiral-like framework (see Fig. 1(a)) in which connected circles are arranged in descending order of size, and the point (red dot in Fig. 1(a)) represents the final result. The kth iteration step in Fig. 1(a) ($k = 1, \cdots, N_k$, where N_k denotes the total number of iteration steps) consists of three components: visual classification, motional estimation, and risk analysis, which are illustrated in Fig. 1(b). The visual and motional trackers are integrated into a generalized framework, and the risk function implements corresponding modifications with respect to the processing errors. The SVMT algorithm is, in fact, a process designed to build a strong tracker from initial weak trackers iteration by iteration. We discuss the detailed procedure of each of the three components in the following sections.

2.1. Visual classification

At each frame, each of the extracted test samples [12] is convolved with a set of Haar-like feature filters at multiple scales [5]. Then, each of them is vectorized into a very high-dimensional image feature $\boldsymbol{h} \in \mathbb{R}^n$ that can be embedded into an extremely compressive feature vector $\boldsymbol{v} \in \mathbb{R}^m$ ($m \ll n$). This linear transformation is expressed as

$$\boldsymbol{v} = \Phi \boldsymbol{h}, \tag{1}$$

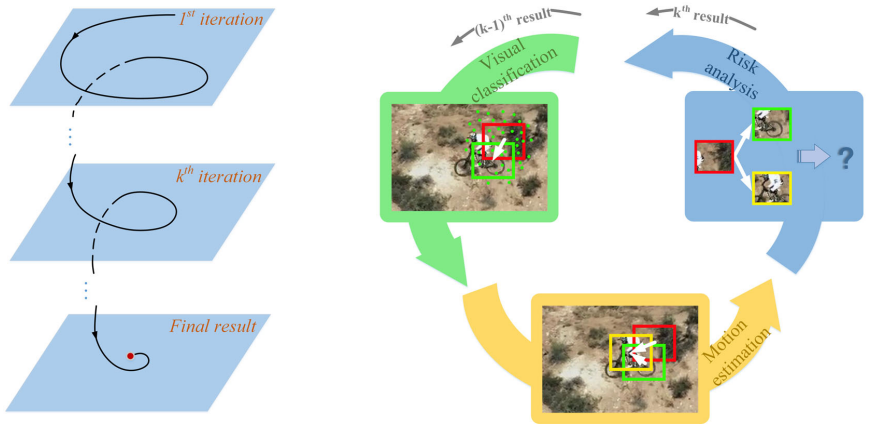

Figure 1: General framework of SVMT: (a) spiral-like representation of SVMT and (b) the three components in the kth iteration step: 1. visual classification: select some samples (centers of which are marked with green dots) around the $(k-1)$th iteration result (red box), and obtain the visual result (green box) through visual classification; 2. motional estimation: compute the motional result (yellow box) based on the $(k-1)$th iteration result (red box) and the visual result (green box) via motional estimation; 3. risk analysis: make risk decision with the $(k-1)$th iteration result (red box), the visual result (green box), and the motional result (yellow box), and output the kth iteration result.

where the random projection matrix $\boldsymbol{\Phi} \in \mathbb{R}^{m \times n}$ is data-independent of any training samples, and has to satisfy the Johnson–Lindenstrauss lemma [29] to restructure $\boldsymbol{h}$ from $\boldsymbol{v}$ with minimum error. In this paper, $\boldsymbol{\Phi}$ is computed according to [30].

The task of visual classification is completed by using a naive Bayesian classifier [31]. We represent the compressive vector of the ith test sample as $\boldsymbol{v}(i) = \{v_1(i), \cdots, v_m(i)\}$. Then, each element $v_j(i)(j = 1, \cdots, m)$ in $\boldsymbol{v}(i)$ is assumed to be independently distributed and modeled with the naive Bayesian classifier [12]. Thus, the classifier score of $\boldsymbol{v}(i)$ in the kth iteration

step is computed as

$$
\begin{aligned}
S^{(k)}(i) &= \log\left\{\frac{\prod_{j=1}^{m} p\left[v_j^{(k)}(i)\left|y^{(k)}=1\right.\right] p\left(y^{(k)}=1\right)}{\prod_{j=1}^{m} p\left[v_j^{(k)}(i)\left|y^{(k)}=0\right.\right] p\left(y^{(k)}=0\right)}\right\} \\
&= \sum_{j=1}^{m} \log\left\{\frac{p\left[v_j^{(k)}(i)\left|y^{(k)}=1\right.\right]}{p\left[v_j^{(k)}(i)\left|y^{(k)}=0\right.\right]}\right\},
\end{aligned}
\tag{2}
$$

where $p(y^{(k)}=1)=p(y^{(k)}=0)$, and $y^{(k)}\in\{0,1\}$ denotes a binary variable representing the positive and negative label of the sample. The conditional distributions in Eq. (2) are assumed to be Gaussian distributed $p[v_j^{(k)}(i)\left|y^{(k)}=1\right.] \sim N(\mu_j^{1(k)},\sigma_j^{1(k)})$ and $p[v_j^{(k)}(i)\left|y^{(k)}=0\right.] \sim N(\mu_j^{0(k)},\sigma_j^{0(k)})$. The positive Gaussian parameters $(\mu_j^{1(k)},\sigma_j^{1(k)})$ are incrementally updated as $\mu_j^{1(k)} \leftarrow \lambda\mu_j^{1(k)}+(1-\lambda)\mu^{1(k)}$ and $\sigma_j^{1(k)} \leftarrow \sqrt{\lambda(\sigma_j^{1(k)})^2+(1-\lambda)(\sigma^{1(k)})^2+\lambda(1-\lambda)(\mu_j^{1(k)}-\mu^{1(k)})^2}$, where $\lambda>0$ is a learning parameter, and $\mu^{1(k)}$ and $\sigma^{1(k)}$ are the mean and covariance Gaussian parameters computed from the historical frames [12]. The negative Gaussian parameters $(\mu_j^{0(k)},\sigma_j^{0(k)})$ are updated in the same way. Then, we find the sample with the maximal classifier score by

$$
S^{(k)}(i_m) = \underset{i}{\operatorname{argmax}}\, S^{(k)}(i). \tag{3}
$$

The state of the i_mth sample with the maximal classifier score $S^{(k)}(i_m)$ is saved as the visual result $\boldsymbol{x}_{vi}^{(k)}$.

In comparison with $S^{(k)}(i)$ which scores the possibility to be the target, the error of the ith sample is defined to score its possibility to be the background

$$
error^{(k)}(i) = \sum_{j=1}^{m} \log\left\{\frac{p\left[v_j^{(k)}(i)\left|y^{(k)}=0\right.\right]}{p\left[v_j^{(k)}(i)\left|y^{(k)}=1\right.\right]}\right\}. \tag{4}
$$

Then, the visual error rate $E_{vi}^{(k)}$ is defined to compute the normalized error rate of the i_mth sample as

$$
E_{vi}^{(k)} = \frac{error^{(k)}(i_m)-\operatorname{argmin}_i error^{(k)}(i)}{\operatorname{argmax}_i error^{(k)}(i)-\operatorname{argmin}_i error^{(k)}(i)}. \tag{5}
$$

2.2. Motional estimation

The Kalman filter is known as an iteration method for state estimation and optimization problems. The optimal estimation of the current value of the parameter can be obtained from the previous estimation and the latest measurement value. In this paper, the prediction form of the Kalman filter is used to estimate the motion state:

$$\begin{cases} \boldsymbol{x}_{mo}^{(k)} = F \cdot \boldsymbol{x}_{mo}^{(k-1)} + \boldsymbol{w}^{(k-1)} \\ \boldsymbol{z}^{(k)} = H \cdot \boldsymbol{x}^{(k)} + \boldsymbol{u}^{(k)} \end{cases}, \tag{6}$$

where $\boldsymbol{x}_{mo}^{(k)}$ and $\boldsymbol{z}^{(k)}$ represent the estimated and measured values of the motion state in the kth iteration step, respectively, and F and H denote the state and measurement transition matrices, respectively. The system noise $\boldsymbol{w}$ and measurement noise $\boldsymbol{u}$ are mutually independent zero-mean Gaussian noise sequence distributions with covariance Q and R, respectively [32].

In the prediction stage, the state and the error covariance are predicted as

$$\begin{cases} \hat{\boldsymbol{x}}_{mo}^{(k|k-1)} = F\hat{\boldsymbol{x}}_{mo}^{(k-1)} \\ P^{(k|k-1)} = FP^{(k-1)}F^T + Q^{(k)} \end{cases}, \tag{7}$$

and after the measurement is attained, the Kalman filter is updated as

$$\begin{cases} K^{(k)} = P^{(k|k-1)}H^T(HP^{(k|k-1)}H^T + R)^{-1} \\ P^{(k)} = (I - K^{(k)}H)P^{(k|k-1)} \\ \hat{\boldsymbol{x}}_{mo}^{(k)} = \hat{\boldsymbol{x}}_{mo}^{(k|k-1)} + K^{(k)}(\boldsymbol{z}^{(k)} - H\hat{\boldsymbol{x}}_{mo}^{(k|k-1)}) \end{cases}, \tag{8}$$

where K is known as the Kalman gain, P is the covariance matrix of the state estimation error, and I denotes the identity matrix. The goal of the filter is to minimize the error between the true and estimated state vectors [33]. In this paper, the measurement state $\boldsymbol{z}^{(k)}$ is defined as

$$\boldsymbol{z}^{(k)} = \begin{cases} \boldsymbol{x}_{vi}^{(k)} & \text{if } S^{(k)} > 0 \\ \boldsymbol{x}_{vi}^{(k-1)} & \text{else} \end{cases}. \tag{9}$$

The motional error rate is defined by evaluating the performance of the Kalman filter as

$$E_{mo}^{(k)} = \left(\frac{\boldsymbol{x}_{mo}^{(k)} - \hat{\boldsymbol{x}}_{mo}^{(k)}}{\boldsymbol{x}_{mo}^{(k)} + \hat{\boldsymbol{x}}_{mo}^{(k)}}\right)^T \left(\frac{\boldsymbol{x}_{mo}^{(k)} - \hat{\boldsymbol{x}}_{mo}^{(k)}}{\boldsymbol{x}_{mo}^{(k)} + \hat{\boldsymbol{x}}_{mo}^{(k)}}\right), \tag{10}$$

which represents the normalized error rate when choosing the estimated state vector $\hat{\boldsymbol{x}}_{mo}^{(k)}$ as the motional result but $\boldsymbol{x}_{mo}^{(k)}$ is true.

2.3. Risk analysis

After obtaining the visual and motional error rates $E_{vi}^{(k)}$ and $E_{mo}^{(k)}$, the risk function is defined as

$$R^{(k)} = w_{vi}^{(k)} E_{vi}^{(k)} + w_{mo}^{(k)} E_{mo}^{(k)}, \tag{11}$$

where $w_{vi}^{(k)}$ and $w_{mo}^{(k)}$ are the visual and motional weights, respectively. In our tracker, it is imperative to adapt appropriate weight options for flexible circumstances. Thus, the determination of risk weights $(w_{vi}^{(k)}, w_{mo}^{(k)})$ is

$$\begin{cases} w_{vi}^{(k)} = 0, w_{mo}^{(k)} = 1 & \text{if } L_{vi}^{(k)} < T_w \\ w_{vi}^{(k)} = 1, w_{mo}^{(k)} = 0 & \text{else if } L_{mo}^{(k)} < T_w \\ w_{vi}^{(k)} = \dfrac{L_{vi}^{(k)}}{L_{vi}^{(k)} + L_{mo}^{(k)}}, w_{mo}^{(k)} = \dfrac{L_{mo}^{(k)}}{L_{vi}^{(k)} + L_{mo}^{(k)}} & \text{else} \end{cases}, \tag{12}$$

where the visual and motional likelihood functions are defined as

$$\begin{cases} L_{vi}^{(k)} = e^{-\lambda_o (\rho_{vi}^{(k)})^2} \\ L_{mo}^{(k)} = e^{-\lambda_o (\rho_{mo}^{(k)})^2} \end{cases}. \tag{13}$$

In Eq. (13), λ_o denotes a control parameter [24], and $\rho_{vi}^{(k)}$ and $\rho_{mo}^{(k)}$ are the Euclidean distances [34] between the template and $\boldsymbol{x}_{vi}^{(k)}$ or $\boldsymbol{x}_{mo}^{(k)}$, respectively. Considering short-time tracking without great appearance changes, the template is set as the feature in the previous frame [2]. The threshold is computed as $T_w = \lambda_w e^{\lambda_o (\rho_o^{(k)})^2}$, where λ_w is a control parameter, and $\rho_o^{(k)}$ represents the mean value of the Euclidean distances between the negative samples and the template.

The main steps of the proposed SVMT algorithm are summarized in Algorithm 1.

3. Experiments

In this section, the SVMT algorithm is tested on several challenging real-world sequences, and some qualitative and quantitative analyses are performed on the tracking results.

Algorithm 1 Spiral visual and motional tracking

Input: the tth frame

for $k = 1$ to N_k do

1 Visual classification:

1.1 Select test samples by $X^{\gamma} = \{\boldsymbol{x}^{(k)} | \|\boldsymbol{x}^{(k)} - \boldsymbol{x}_{mo}^{(k-1)}\| < \gamma\}$.

1.2 Obtain the compressive feature vector $\boldsymbol{v}$ using Eq. (1), and then apply the naive Bayes classifier to get the classifier score $S^{(k)}(i)$ by Eq. (2).

1.3 Find the i_mth sample with maximal classifier score $S^{(k)}(i_m)$ using Eq. (3) and save its state $\boldsymbol{x}_{vi}^{(k)}$ as visual result.

1.4 Compute the visual error rate $E_{vi}^{(k)}$ by Eq. (5).

2 Motional estimation:

2.1 Use Kalman filter to obtain the motional result $\boldsymbol{x}_{mo}^{(k)}$.

2.2 Compute the motional error rate $E_{mo}^{(k)}$ by Eq. (10).

3 Risk analysis:

3.1 Compute the risk weights $(w_{vi}^{(k)}, w_{mo}^{(k)})$ by Eq. (8).

3.2 Obtain the risk function $R^{(k)}$ using Eq. (11).

3.3 If the reduction in $R^{(k)}$ is smaller than a threshold, break.

4 Updating:

4.1 Extract positive and negative samples via $X^{\alpha} = \{\boldsymbol{x}^{(k)} | \|\boldsymbol{x}^{(k)} - \boldsymbol{x}_{mo}^{(k)}\| < \alpha\}$ and $X^{\varsigma,\beta} = \{\boldsymbol{x}^{(k)} | \varsigma < \|\boldsymbol{x}^{(k)} - \boldsymbol{x}_{mo}^{(k)}\| < \beta\}$.

4.2 Extract the sample features and update the classifier parameters.

end for

Output: tracking result $\boldsymbol{x}_{mo}^{(k)}$.

3.1. Experimental setup and evaluation criteria

The sample parameters were set as $\alpha = 4$, $\beta = 30$, $\varsigma = 8$ and $\gamma = 20$ [30], and the control parameters λ_o and λ_w were set as 1.8 and 1.2, respectively. The state and measurement transition matrices of the Kalman filter were set as [35]

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

and the covariance of the system and measurement noise were set as $Q = 0.1$ and $R = 1.0$ [33]. SVMT was implemented using Visual Studio 2010 on an Intel Dual-Core 1.70GHz CPU with 4 GB RAM.

Two metrics, i.e., location error (pixel) [36] and overlapping rate (%) [6], are used to evaluate the tracking results quantitatively. The location error is computed as $error = \sqrt{(x_G - x_T)^2 + (y_G - y_T)^2}$, where (x_G, y_G) and (x_T, y_T) are the ground truth (either downloaded from a standard database or located manually) and tracking bounding box centers, respectively. The tracking overlapping rate is defined as $overlapping = area(ROI_G \bigcap ROI_T)/area(ROI_G \bigcup ROI_T)$, where ROI_G and ROI_T denote the ground truth and tracking bounding box, respectively, and $area(\cdot)$ is the rectangular area function. A smaller location error and a larger overlapping rate indicate higher accuracy and robustness.

3.2. Experimental results

The performance of SVMT is compared with state-of-the-art visual trackers FCT [12], MIL [5], and ODFS [6], the motional tracker KF [15], and the simultaneous visual and motional tracker TLD [25]. Figs. 2 to 8 and Tables 1 to 2 present the experimental results in twelve challenging sequences named *Basketball*, *Bear*, *Bike*, *Car*, *Deer*, *Doll*, *Faceocc*, *Fox*, *MHuang*, *Shaking*, *Skater*, and *Sylv*, where *Bear* and *Fox* were collected from the Animal World TV show by ourselves, and the others are publicly available [37]. Next, the performance of each sequence is described in detail.

3.2.1. Low resolution and partial occlusion

Sequences *Bear* and *Fox* are low-resolution recordings and present a greater challenge in terms of occlusion. In Sequence *Bear* (131 frames in total), the target polar bear is surrounded and occluded by snow drifts with an appearance similar to that of the target. These background disturbances together with occlusion cause the ODFS, KF, and TLD trackers to drift slightly from the target, as shown in Fig. 2. As seen in Figs. 6 and 7, SVMT performs the best in terms of location error and overlapping rate due to the effectiveness and robustness of combining the features of the visual and motional layers. For convenience of presentation, the ODFS tracking curve is not shown entirely in Fig. 6.

In the first part of Sequence *Fox* (277 frames in total), all six trackers perform well. However, when the target fox walks behind the bushes at around Frame #150, and is occluded by the video subtitle at around Frame #193, ODFS and KF lose track of the target (see Fig. 2). Figs. 6 and 7 also indicate these failures when occlusion occurs. SVMT is able to overcome the partial occlusion and delivers the best performance for this video.

The target in Sequence *Doll* (3870 frames in total) undergoes background clutter and partial occlusion by hands (Frames #2557 and #2637). As can be seen from Fig. 2, only ODFS, KF, and the proposed SVMT algorithms can overcome these problems and achieve good tracking results, whereas SVMT performs the best.

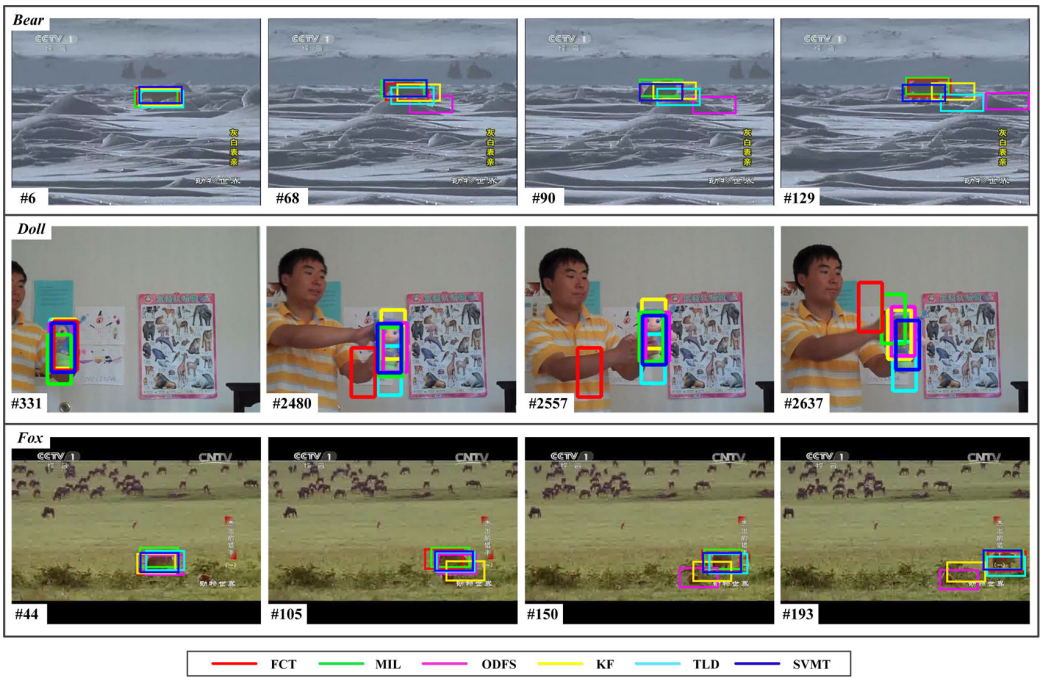

Figure 2: Tracking performances of the test sequences with low resolution and partial occlusion.

3.2.2. Illumination change

The efficiency of SVMT is demonstrated by using Sequence *Car* (351 frames in total) in which a large change in illumination is displayed. This sequence is blurry because it was captured at night and the color of the light is similar to that of the target. As shown in Fig. 3, all the trackers lose the track of the target, except for SVMT which is able to overcome the large illumination change and performs well on this sequence. The location errors and overlapping rates of the other five trackers increased and decreased frame by frame, respectively (see Figs. 6 and 7).

In Sequence *MHuang* (1071 frames in total), not only the illumination of the background (Frames #758 and #1071) but also the facial expression of the target man (Frames #513 and #985) keeps changing throughout the tracking process. Most of the twelve trackers achieve the tracking task successfully, but the proposed SVMT algorithm performs the best.

Sequence *Shaking* (364 frames in total) contains examples of illumination change (Frame #59), pose variation (Frames #127 and #249), and occlusion (Frame #164). When the target undergoes illumination changes at around Frame #59, MIL, KF, and TLD drift toward the background, as shown in Fig. 3. Then, MIL and KF identify a false target because the true target is occluded by the guitar (see Frame #164 in Fig. 3). Besides, the true target keeps on shaking his head during the whole tracking process such that FCT and ODFS are unable to track it accurately (see Frames #127 and #249 in Fig. 3). Once again, SVMT outperforms most of the other methods in most metrics (location accuracy and success rate).

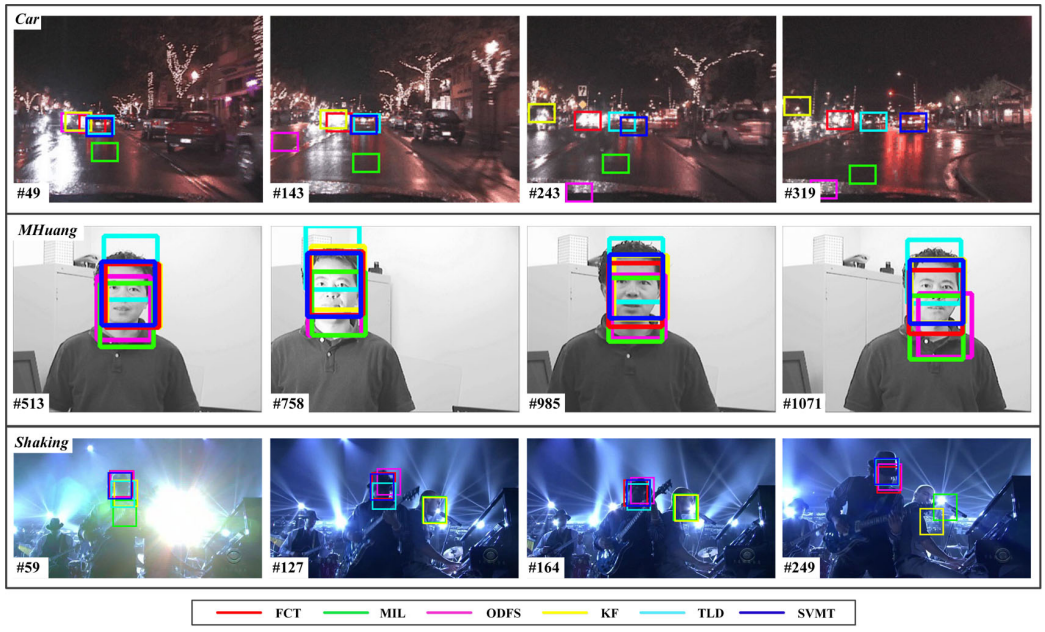

Figure 3: Tracking performances of the test sequences with illumination change.

3.2.3. Pose variation and background clutter

The target mountain bike in Sequence *Bike* (227 frames in total) undergoes background clutter (Frames #45 and #103), pose rotation (Frame #65), and abrupt movement (Frame #210). As can be seen from Fig. 4, MIL and KF lose the target during most of the tracking process. While the target is passing by the mountain at around Frame #103, a background with a similar pattern distracts FCT from the target. In contrast, TLD and SVMT achieve favorable performances in terms of both tracking error and success rate (see Figs. 6 and 7) due to the combined learning of visual and motional information, whereas SVMT performs better with the spiral visual and motional model.

Sequence *Skater* (159 frames in total), in which target is a figure skater, demonstrates the efficiency of SVMT on coping with large-scale pose variation. Sequence *Sylv* (1273 frames in total) also undergoes pose variation and background clutter frequently and heavily. As can be seen from Fig. 4, other five trackers cannot provide accurate tracking results, whereas our tracker shows satisfying performance in terms of both accuracy and robustness.

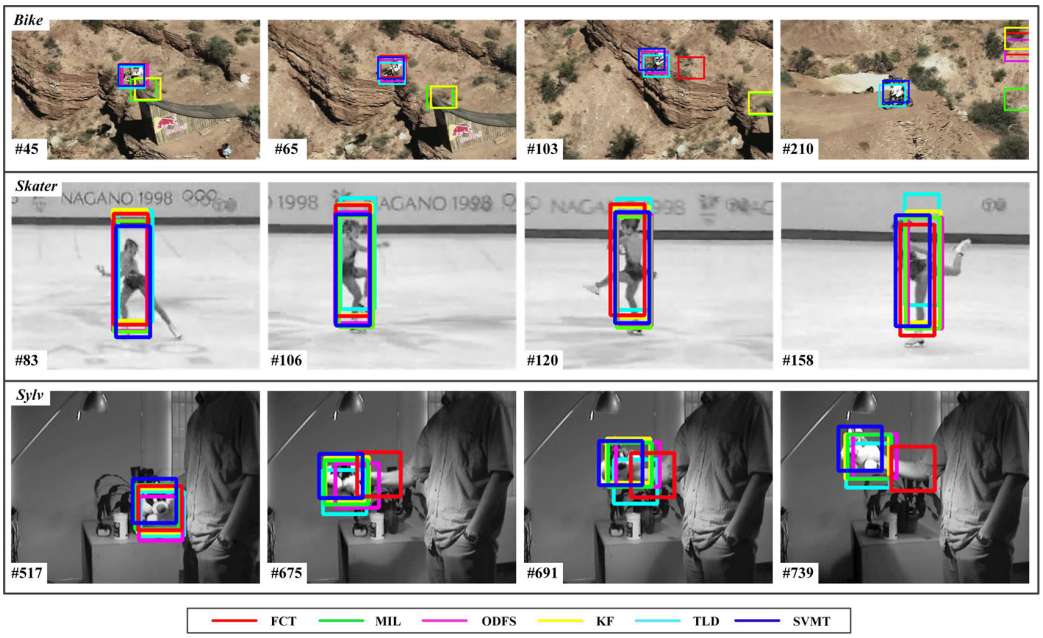

Figure 4: Tracking performances of the test sequences with pose variation and background clutter.

3.2.4. Large-scale occlusion and abrupt movement

Sequence *Basketball* (203 frames in total) shows the performances of these trackers when the target is non-rigid and undergoes heavy occlusion and abrupt movement. As shown in Fig. 5, when the target player is fully occluded by another player at around Frame #20, FCT loses the target, and MIL as well as TLD drifts away from the target. At around Frame #53, TLD mistakes another player for the target when the true target reappears in the camera view. Only KF, ODFS, and SVMT are able to handle these problems, whereas the result of SVMT is the most accurate, as shown in Figs. 6 and 7.

The experiment in Sequence *Deer* (70 frames in total) aims at evaluating the performances on tracking the head of the deer. As can be seen from Fig. 5, this sequence is a low-resolution and low-frame-rate recording, and the target is fully occluded by another deer. Only our tracker keeps a high accuracy for most of the time in the whole tracking process.

Sequence *Faceocc* (884 frames in total) is an occlusion sequence in which the woman's face is occluded by a book in some different directions. Fig. 5 shows the performances of the six tracking algorithms. When the occlusion occurs, some trackers mistake the book for the target (i.e., FCT and TLD), some drift away from the target (i.e., ODFS), and some present low-accurate locating (i.e., MIL and KF). Only the proposed SVMT algorithm performs a robust tracking.

Tables 1 and 2 are included here to demonstrate the performance of the twelve test sequences on average location error (pixel) and success rate (%). The success rate is defined as the number of times success is achieved in the whole tracking process by considering one frame as a success if the overlapping rate exceeds 0.5 [6]. A smaller average location error and a larger success rate indicate increased accuracy and robustness. In Sequence *Car*, most of the trackers do not achieve a large success rate (see Table 2) because the target sizes are relatively small such that a slight drift away from the target may cause a great reduction in the success rates. Tables 1 and 2 show that SVMT outperforms both state-of-the-art separate and simultaneous visual and motional trackers.

The proposed SVMT tracking algorithm is a combination between visual and motional trackers. In this experiment, the importance of these two components to improve the tracking performance is evaluated in Fig. 8. Al-

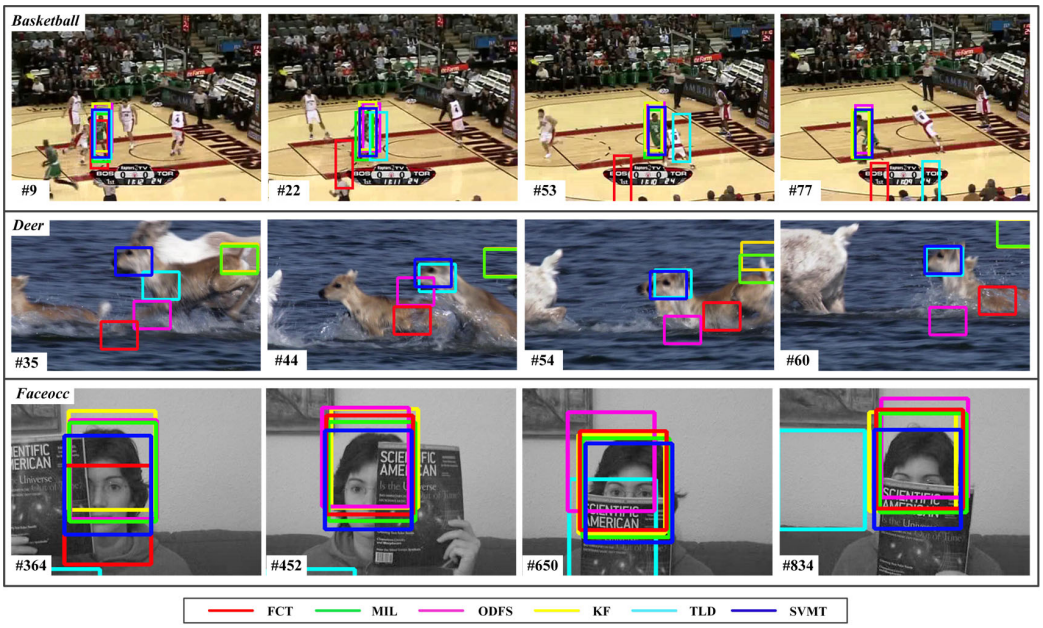

Figure 5: Tracking performances of the test sequences with large-scale occlusion and abrupt movement.

Table 1: Comparisons on average location error (pixel) of the test sequences. Bold fonts indicate the best performance.

Sequences	SVMT	FCT	MIL	ODFS	KF	TLD
Basketball	**15**	106	17	23	23	148
Bear	**6**	8	**6**	108	41	47
Bike	**6**	156	218	126	219	13
Car	**7**	51	58	105	77	30
Deer	**9**	120	220	91	252	13
Doll	**5**	72	24	13	19	**5**
Faceocc	**18**	32	24	18	16	34
Fox	**3**	9	11	48	101	14
MHuang	**5**	47	13	30	14	36
Shaking	**7**	15	155	18	153	36
Skater	**11**	17	15	14	21	26
Sylv	**7**	13	11	14	9	27

though visual part of the proposed SVMT algorithm is essential for tracking, when the visual feature is unreliable, the motional part needs to be more

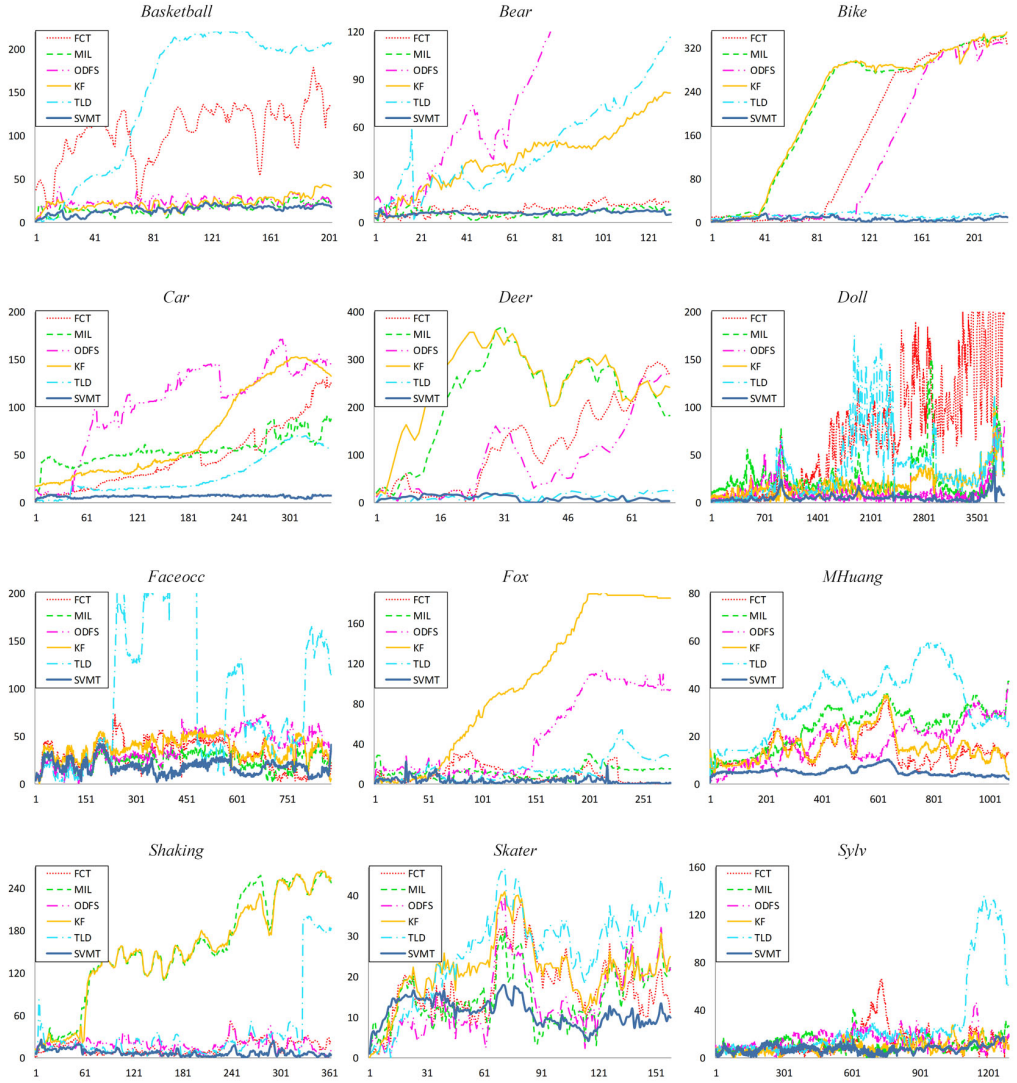

Figure 6: Comparisons on location error (pixel) of the test sequences.

important than the visual one to keep a good tracking. As can be seen from Fig. 8, visual and motional weights compete with each other frequently and fiercely for almost all these test sequences. For example, when the target deer in Sequence *Deer* is occluded by another deer with a similar appear-

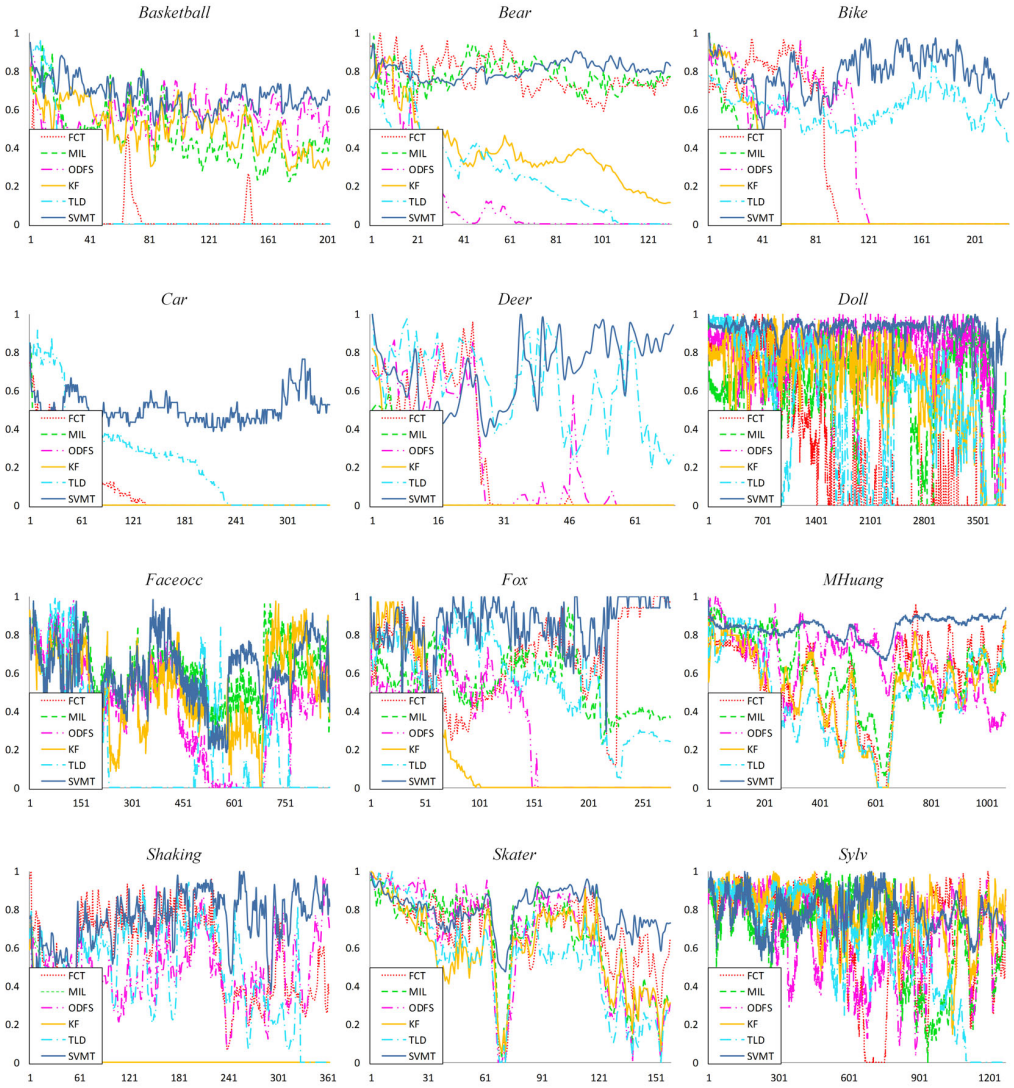

Figure 7: Comparisons on overlapping rate of the test sequences.

ance, relying only on visual feature may cause tracking failure. Therefore, our tracker decreases the visual weight and meanwhile increases the motional one so as to achieve a favorable tracking performance.

Table 2: Comparisons on success rate (%) of the test sequences. Bold fonts indicate the best performance.

Sequences	SVMT	FCT	MIL	ODFS	KF	TLD
Basketball	**100**	0	33	76	57	9
Bear	**100**	**100**	**100**	12	17	1
Bike	**100**	38	12	48	16	86
Car	**41**	3	1	0	0	12
Deer	**80**	30	6	31	4	70
Doll	**100**	29	76	92	81	67
Faceocc	**75**	53	74	40	53	22
Fox	**98**	77	60	31	20	68
MHuang	**100**	67	74	81	62	46
Shaking	**94**	59	4	40	1	45
Skater	**98**	84	74	69	66	67
Sylv	**100**	83	78	75	94	82

4. Conclusion

Visual tracking can be completed through both visual and motional processes. Visual and motional tracking are closely interrelated and there is no clear boundary between them. However, conventional solutions ignore this inter-dependence completely or partially, which has a negative impact on the performance. Unlike existing approaches, this paper proposes an approach in which the spiral vision and motional tracking (SVMT) algorithm unifies visual and motional trackers into a generalized framework, and uses an iteration model to achieve optimization. In SVMT, each iteration step is decomposed into visual classification, motional estimation, and risk analysis steps in which the latter step represents the error of each iteration step and makes corresponding modification to approach optimization. SVMT is, in fact, a process designed to build a strong tracker from initially weak trackers iteration by iteration. Numerous real-world video sequences were used to test SVMT and other state-of-the-art algorithms, and here we only selected representative videos for presentation. Thus, experimental results demonstrated that SVMT is highly accurate and robust.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant No. 61673270) and China National Key Basic Research and Development Program (973 Program, Grant No. 2014CB744903).

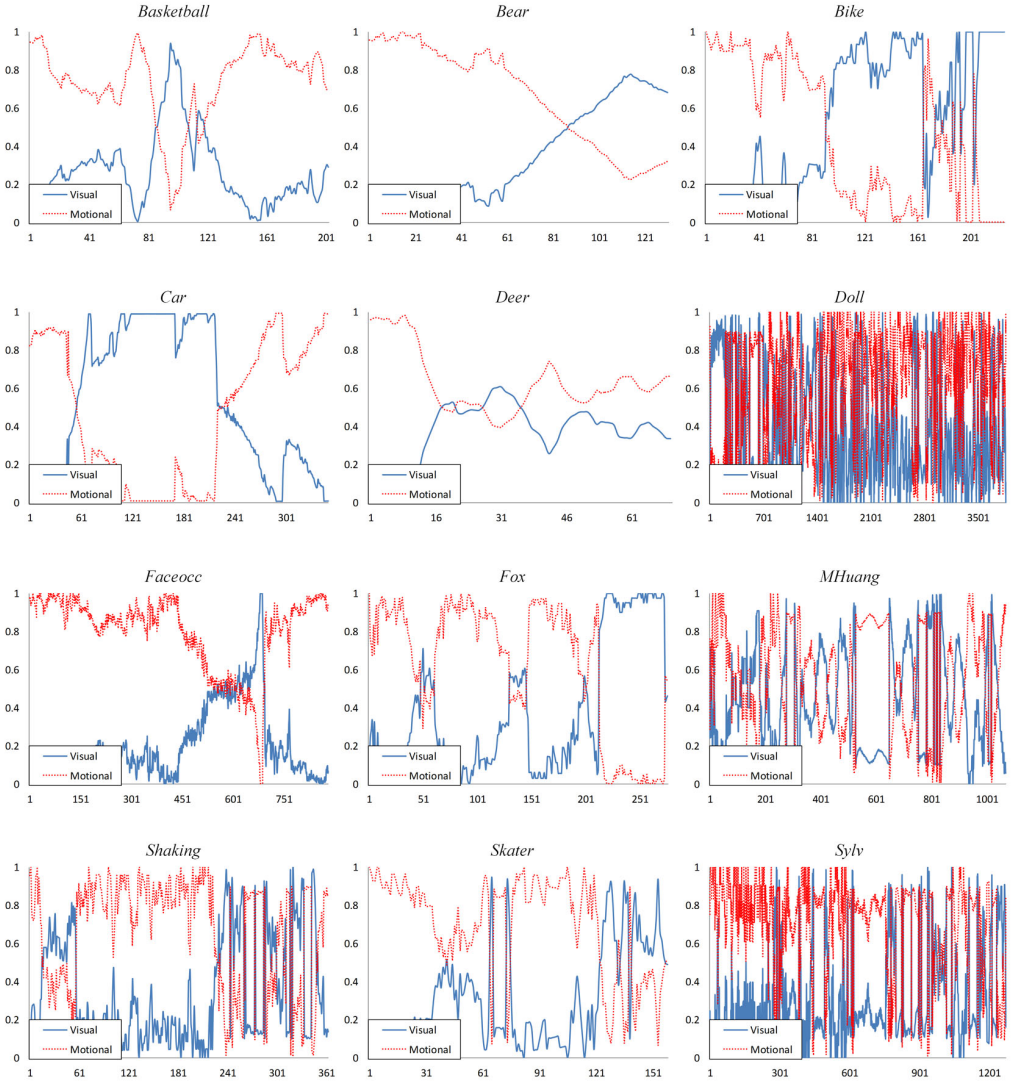

Figure 8: Comparisons on visual and motional weights in SVMT of the test sequences.

References

[1] T. A. Biresaw, A. Cavallaro, C. S. Regazzoni, Correlation-based self-correcting tracking, Neurocomputing 152 (2015) 345–358.

[2] X. Yun, Z. Jing, G. Xiao, B. Jin, C. Zhang, A compressive tracking

based on time-space Kalman fusion model, Science China Information Sciences 59 (1) (2016) 1–15.

[3] X. Mei, H. Ling, Robust visual tracking and vehicle classification via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (11) (2011) 2259–2272.

[4] H. Li, C. Shen, Q. Shi, Real-time visual tracking using compressive sensing, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 1305–1312.

[5] B. Babenko, M.-H. Yang, S. Belongie, Robust object tracking with online multiple instance learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (8) (2011) 1619–1632.

[6] K. Zhang, L. Zhang, M.-H. Yang, Real-time object tracking via online discriminative feature selection, IEEE Transactions on Image Processing 22 (12) (2013) 4664–4677.

[7] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, Robust visual tracking via multi-task sparse learning, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 2042–2049.

[8] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, Robust visual tracking via structured multi-task sparse learning, International Journal of Computer Vision 101 (2) (2013) 367–383.

[9] T. Zhang, S. Liu, C. Xu, S. Yan, B. Ghanem, N. Ahuja, M.-H. Yang, Structural sparse tracking, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 150–158.

[10] T. Zhang, K. Jia, C. Xu, Y. Ma, N. Ahuja, Partial occlusion handling for visual tracking via robust part matching, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1258–1265.

[11] T. Zhang, S. Liu, N. Ahuja, M.-H. Yang, B. Ghanem, Robust visual tracking via consistent low-rank sparse learning, International Journal of Computer Vision 111 (2) (2015) 171–190.

[12] K. Zhang, L. Zhang, M.-H. Yang, Fast compressive tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (10) (2014) 2002–2015.

[13] H. Song, Robust visual tracking via online informative feature selection, Electronics Letters 50 (25) (2014) 1931–1933.

[14] Y. Hua, K. Alahari, C. Schmid, Occlusion and motion reasoning for long-term tracking, in: European Conference on Computer Vision (ECCV), 2014, pp. 172–187.

[15] R. E. Kalman, A new approach to linear filtering and prediction problems, Journal of Fluids Engineering 82 (1) (1960) 35–45.

[16] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5) (2002) 603–619.

[17] M. Isard, A. Blake, Icondensation: Unifying low-level and high-level tracking in a stochastic framework, in: European Conference on Computer Vision (ECCV), 1998, pp. 893–908.

[18] D. W. Park, J. Kwon, K. M. Lee, Robust visual tracking using autoregressive hidden markov model, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 1964–1971.

[19] T. Zhou, Y. Yan, Video target tracking based on mean shift algorithm with Kalman filter, in: 10th International Conference on Natural Computation (ICNC), 2014, pp. 980–984.

[20] A. Ghahremani, A. Mousavinia, Visual object tracking using Kalman filter, mean shift algorithm and spatiotemporal oriented energy features, in: 4th International Conference on Computer and Knowledge Engineering (ICCKE), 2014, pp. 625–629.

[21] G. M. Rao, S. P. Nandyala, C. Satyanarayana, Fast visual object tracking using modified Kalman and particle filtering algorithms in the presence of occlusions, International Journal of Image, Graphics and Signal Processing 6 (10) (2014) 43.

[22] S. Zhang, S. Chan, B. Liao, K. Tsui, A new visual object tracking algorithm using Bayesian Kalman filter, in: IEEE International Symposium on Circuits and Systems (ISCAS), 2014, pp. 522–525.

[23] H. Firouzi, H. Najjaran, Adaptive non-rigid object tracking by fusing visual and motional descriptors, in: International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2010, pp. 58–62.

[24] L. Cehovin, M. Kristan, A. Leonardis, An adaptive coupled-layer visual model for robust visual tracking, in: IEEE International Conference on Computer Vision (ICCV), 2011, pp. 1363–1370.

[25] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-learning-detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (7) (2012) 1409–1422.

[26] Q. Liu, J. Yang, K. Zhang, Y. Wu, Adaptive compressive tracking via online vector boosting feature selection, IEEE Transactions on Cybernetics PP (99) (2016) 1–13.

[27] J. Yang, K. Zhang, Q. Liu, Robust object tracking by online Fisher discrimination boosting feature selection, Computer Vision and Image Understanding 153 (2016) 100–108.

[28] K. Zhang, Q. Liu, Y. Wu, M. H. Yang, Robust visual tracking via convolutional networks without training, IEEE Transactions on Image Processing 25 (4) (2016) 1779–1792.

[29] D. Achlioptas, Database-friendly random projections: Johnson-lindenstrauss with binary coins, Journal of computer and System Sciences 66 (4) (2003) 671–687.

[30] K. Zhang, L. Zhang, M.-H. Yang, Real-time compressive tracking, in: European Conference on Computer Vision (ECCV), 2012, pp. 864–877.

[31] A. Jordan, On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes, Advances in neural information processing systems 14 (2002) 841.

[32] D. Simon, Training radial basis neural networks with the extended Kalman filter, Neurocomputing 48 (1) (2002) 455–475.

[33] C. K. Chui, G. Chen, Kalman filtering with real-time applications, Springer-Verlag New York, Inc., 1987.

[34] P. Pérez, C. Hue, J. Vermaak, M. Gangnet, Color-based probabilistic tracking, in: European Conference on Computer Vision (ECCV), 2002, pp. 661–675.

[35] F. B. Vidal, V. H. C. Alcalde, Window-matching techniques with Kalman filtering for an improved object visual tracking, in: IEEE International Conference on Automation Science and Engineering (CASE), 2007, pp. 829–834.

[36] N. Jiang, W. Liu, Y. Wu, Learning adaptive metric for robust visual tracking, IEEE Transactions on Image Processing 20 (8) (2011) 2288–2300.

[37] Y. Wu, J. Lim, M.-H. Yang, Online object tracking: A benchmark, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2411–2418.