Robust Online Learned Spatio-Temporal Context Model for Visual Tracking

Longyin Wen, Zhaowei Cai, Zhen Lei, *Member, IEEE*, Dong Yi, and Stan Z. Li, *Fellow, IEEE*

***Abstract*— Visual tracking is an important but challenging problem in the computer vision field. In the real world, the appearances of the target and its surroundings change continuously over space and time, which provides effective information to track the target robustly. However, enough attention has not been paid to the spatio-temporal appearance information in previous works. In this paper, a robust spatio-temporal context model based tracker is presented to complete the tracking task in unconstrained environments. The tracker is constructed with temporal and spatial appearance context models. The temporal appearance context model captures the historical appearance of the target to prevent the tracker from drifting to the background in a long-term tracking. The spatial appearance context model integrates contributors to build a supporting field. The contributors are the patches with the same size of the target at the key-points automatically discovered around the target. The constructed supporting field provides much more information than the appearance of the target itself, and thus, ensures the robustness of the tracker in complex environments. Extensive experiments on various challenging databases validate the superiority of our tracker over other state-of-the-art trackers.**

***Index Terms*— Visual tracking, spatio-temporal context, multiple subspaces learning, online boosting.**

I. Introduction

VISUAL tracking has attracted much research due to its importance to practical applications, *e.g.* human-computer interaction, video surveillance, virtual reality, object navigation, etc. Trackers are usually required to work in a long period in unconstrained environments. Challenges arise to the robustness of trackers under various factors, such as pose changes, illumination variation, occlusion. To overcome these difficulties, numerous complex models are designed, but most of them focus on the variations of the target appearance only, *e.g.* by using generative strategy [1], [9], [17], [18], [24], [26], [29], [32], discriminative strategy [2], [3], [5], [12], [13], [23], and other strategies [8], [13], [19], [34], [39], [42], while ignore the relationships between the target and its surroundings.

A typical visual tracking system generally consists of three parts: 1) appearance model, which evaluates the likelihood of the target candidate state; 2) motion model, which is utilized to predict the possible state of the target; and 3) search strategy, which is exploited to find the optimal state of the target. The appearance model, as the most crucial part in the visual tracking system, attracts much more attentions from the researchers. Some work has also been done on the search strategy [22], [43]. In real applications, the motion of target is hard to define, especially when the video is captured by a moving camera, therefore just a few works pay attention to the motion model [20], [28].

This paper focuses on improving the appearance model to build a robust tracker, and doesn't use any assumptions about the targets and scenarios. It exploits the temporal and spatial appearance context information of target to improve the robustness. On the one hand the historical appearances usually influence the subsequent appearances, the temporal context is an important clue to predict the next state of the target. Meanwhile, the appearance of the surrounding background of the target changes gradually during the tracking, so that the spatial context information between the target and the background is also important to differentiate the target from the background. Intuitively, combining these two constraints can achieve better performance.

A. Related Works

In recent decades, numerous tracking methods have been proposed in literatures. However, most of them just perform well in some specific conditions and the performances can't appeal the demands of real applications. Please refer to the survey [41] for more tracking strategies.

Some methods [1], [9], [17], [18], [24], [26], [29], [32] model the appearance of a target in a generative way. In [26], an *Incremental Visual Tracker* (IVT) is proposed, which adaptively updates its subspace-based appearance model with the sequential appearance variations. Based on this fundamental work, many improvements [17], [24], [29], [32] are proposed. Although good tracking results are obtained in some specific environments, their performance will drop in unconstrained environments. Fragment-based tracker [1] describes the target

Manuscript received September 5, 2012; revised May 23, 2013, September 26, 2013, and November 6, 2013; accepted November 16, 2013. Date of publication November 28, 2013; date of current version January 9, 2014. This work was supported in part by the Chinese National Natural Science Foundation under Projects 61070146, 61105023, 61103156, 61105037, 61203267, and 61375037, in part by the National IoT Research and Development under Project 2150510, in part by the National Science and Technology Support Program under Project 2013BAK02B01, in part by the Chinese Academy of Sciences under Project KGZD-EW-102-2, and in part by the AuthenMetric Research and Development Funds. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Hassan Foroosh.

The authors are with the Center for Biometrics and Security Research and National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lywen@cbsr.ia.ac.cn; zwcai@cbsr.ia.ac.cn; zlei@cbsr.ia.ac.cn; dyi@cbsr.ia.ac.cn; szli@cbsr.ia.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2013.2293430

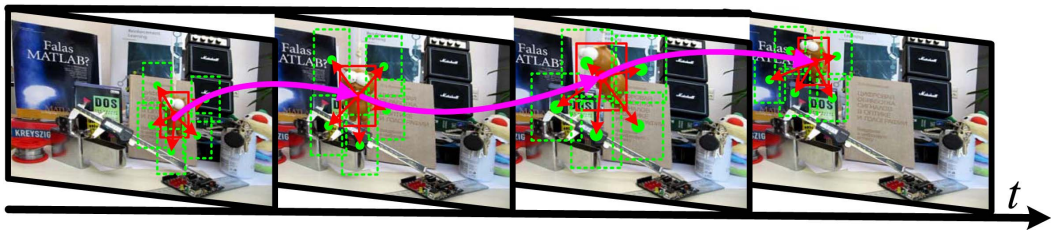


Fig. 1. The use of temporal context constraint and spatial context constraint in the proposed tracking framework. The red solid rectangle represents the target and green dashed rectangles represent the contributors. The magenta curve describes the temporal continuous property of the target while the red arrows describe the spatial supports of the contributors.

with multiple local patch histograms, which integrates the inner structure of the target and handles partial occlusion very well. However, its template is not updated over time, making it difficult to track objects with significant appearance variations.

Compared with the generative models, discriminative trackers [2], [3], [5], [12], [13], [23] regard tracking problem as a classification task, which focuses on differentiating the target from background. Avidan [2] integrates the *Support Vector Machine* (SVM) classifier into the optical flow framework for car tracking. Grabner et al. propose an efficient supervised online boosting tracking method [12]. A semi-supervised version [13] is proposed, in which the labeled data in the first frame is used whereas subsequent training samples are left unlabeled. Bakenko et al. [5] use *Multiple Instance Learning* (MIL) to handle the unreliable labeled positive and negative data obtained online to mitigate the drift problem. In all these trackers, only the appearance of target is considered, but the relationships between target and its background are not fully exploited.

To improve the robustness of the appearance model, Yu et al. [42] combine the generative and discriminative model in a co-training way. Another popular way in surveillance scenarios is introducing detection module into tracking process [13], [19], [34]. The appearance model can be corrected by the detector over time and the target can be recaptured even if it has moved out of view. However, these detection based trackers are easily distracted by other objects with similar appearance.

For long-term tracking tasks in unconstrained environments, some spatial constraints have been introduced to improve the robustness. Yang et al. construct *Context-Aware Tracker* (CAT) [40] to prevent the drifting problem, in which the context are some auxiliary objects that are easy to track and have consistent motion correlations with the target. Similar to CAT, Gu et al. [15] consider the spatial relations between the similar objects and propose to track these similar objects simultaneously. Saffari et al. [33] exploit the multi-class LPBoost algorithm to model the variation of the background to complete the tracking task. Grabner et al. [14] invent a novel concept, supporter, to predict the state of the target. Dinh et al. [10] expand the concept of supporter and develop a new context framework based on distracters and supporters. The distracters are the regions that have similar appearance to the target and the supporters are the local key-points which have motion correlation with the target in a short time span. Li et al. [23] propose a SVM based tracker, by constructing a kernel to effectively capture the contextual information of the samples for better tracking performance. Although these trackers make use of such information, the motion correlation between the target and the context is hard to define. Different from the aforementioned methods, our tracker integrates the spatio-temporal information into appearance modeling to achieve more stable and effective tracking.

B. Outline of Approach

In this paper, we incorporate both the spatial and temporal information into the target appearance modeling and propose a *Spatio-Temporal context model based Tracker* (STT). The STT consists of the temporal and spatial context models. For temporal context model, a novel online subspace learning model is proposed to represent the target with low-dimensional feature vectors. Several sequential positive samples are packed into one subspace to update the model. In this way, the temporal appearance information is efficiently extracted to help predict the next state of the target (see Fig. 1). For the spatial context model, we defined a notion *Contributor*, which is viewed as the local contextual information, *i.e.* the patch of the same size as the target at key points around the target. The key points are generated by SURF [6]. Motivated by Fragment-based tracker [1], we divide the target and the contributors into several small blocks to construct the structure relation features. Both the inter-structure relation features (between target blocks and contributor blocks) and intra-structure relation features (between blocks in target) are extracted (see Fig. 2). In unconstrained environments, it is not easy to dig out the strong contextual supports directly. On the other hand, numerous weak contextual supports around the target can be combined to form a strong supporting field. In this work, the representative relational features are optimally selected by Boosting [5] from the dynamically constructed structural relation feature pool, and these features construct the strong supporting field.

Some preliminary results have been shown in our prior work [38], in this paper, we extend the online subspace learning method to multiple subspaces learning, and provide more experiments and analysis to evaluate the effectiveness of the proposed method. The main contributions of the work are summarized as follows:

1) A novel spatio-temporal context model based tracker is proposed, which integrates the spatio-temporal information into the appearance modeling to improve the discriminative ability of the tracker.

2) A temporal context model is constructed by a novel online subspace learning model, in which positive samples in consecutive frames are combined for the online updating, with consideration on correlation between the samples.
3) A spatial context model is constructed by considering the relationships between the target and its surrounding contributors. Instead of building complex motion models to represent the correlation between the target and the contributors, this work efficiently selects the most representative weak relations to construct a strong supporting field by boosting method.
4) Experiments about the proposed multiple subspaces learning model indicate that the performance of the subspace model is improved remarkably by the sample combination step in updating, rather than the introduction of multiple subspaces.
5) Tracking experiments on various publicly available challenging videos demonstrate that the proposed tracker outperforms other state-of-the-art trackers.

The remainder of the paper is organized as follows. Section II gives the overview of our proposed STT. Section III and section IV describe the temporal context model and spatial context model in detail respectively. Experimental results and discussions are presented in Section V. Finally, we conclude this paper in Section VI.

II. Spatio-Temporal Context Modeling

We assume the target state in tracking task follows the Markovian state transition process, where the current target state is determined by its previous states. Let $O_{1:t} = \{O_1, \ldots, O_t\}$ be the observation data set at time t, and $Z_t = (l_t, \pi_t)$ be the target state, where l_t is the target center position and π_t is the size of the target. Then the posterior probability can be estimated using a recursive process:

$$P(Z_t|O_{1:t}) \propto P(O_t|Z_t) \int P(Z_t|Z_{t-1})P(Z_{t-1}|O_{1:t-1})\mathrm{d}Z_{t-1}, \quad (1)$$

where $P(O_t|Z_t)$ is the likelihood of the candidate state, $P(Z_t|Z_{t-1})$ is the state transition probability for the first-order model we are using, and $P(Z_{t-1}|O_{1:t-1})$ is the state estimation probability given all observations at time $t-1$. We define the optimal solution of Equ. (1) as the *Maximum-a-Posteriori* (MAP) estimation, that is

$$Z_t^* = \arg\max_{Z_t} P(Z_t|O_{1:t}).$$

The above probability terms in Equ. (1) are modeled as follows. Assuming the position and size of the target are independent to each other, the state transition probability $P(Z_t|Z_{t-1})$ is modeled as

$$P(Z_t|Z_{t-1}) = P(l_t|l_{t-1})P(\pi_t|\pi_{t-1}).$$

The position transition term $P(l_t|l_{t-1})$ is specified as

$$P(l_t|l_{t-1}) \propto \begin{cases} 1 & \|l_t - l_{t-1}\|_2 < R \\ 0 & \|l_t - l_{t-1}\|_2 \geq R \end{cases},$$

where R is the search radius. The size transition term $P(\pi_t|\pi_{t-1})$ is similarly defined as [4], which is determined by scaling up and down c scales of the target size π_{t-1} in the previous frame.

The target surroundings are important to help determine the target state. Intuitively, we exploit some contributors around the target to construct the contributor state set $f^r(\cdot)$ and use an algorithm to select some useful relations between the target and the contributors to describe the relationships between the target and its surroundings. Let m_c be the number of contributors and $f^r(\cdot) = \{f_1^r(\cdot), \ldots, f_{m_c}^r(\cdot)\}$. The likelihood of the target candidate state in Equ. (1) is defined as:

$$P(O_t|Z_t) \propto \exp\Big\{ -(1-\kappa_b)U(Z_t|O_t) - \kappa_b \cdot U(Z_t|f^r(\cdot), O_t)\Big\}, \quad (2)$$

where $U(Z_t|O_t)$ is the energy function corresponding to the temporal context model $\mathcal{M}^{(t)}$, $U(Z_t|f^r(\cdot), O_t)$ is the energy function corresponding to the spatial context model $\mathcal{M}^{(s)}$, and $\kappa_b \in (0, 1)$ is the balance parameter between the two energy functions. To avoid unreliable updating, we set the energy thresholds $\theta^{(t)}$ and $\theta^{(s)}$ to control whether the two models will be updated. If both $U(Z_t^*|O_t) < \theta^{(t)}$ and $U(Z_t^*|f^r(\cdot), O_t) < \theta^{(s)}$, $\mathcal{M}^{(t)}$ and $\mathcal{M}^{(s)}$ will be updated with the current optimal target state Z_t^*; otherwise, neither will be updated. In the following two sections, we will discuss the global temporal context model $\mathcal{M}^{(t)}$ and the local spatial context model $\mathcal{M}^{(s)}$ in details.

III. Global Temporal Context Model

Target tracking is a physically consecutive process and there exist strong correlations between the target appearances in consecutive frames. Therefore, it is reasonable to use the correlation information to predict the states of the target. Our global temporal context exploits historical appearance changes as a source of global constraints to estimate the state of the target. Here, we propose a novel online subspace learning method to reduce the high dimensionality of the feature space, so that more historical information will be stored and exploited.

Subspace learning has been used for tracking. Li [25] proposes an incremental algorithm for robust *Principal Component Analysis* (PCA). Skocaj. et al. [35] exploit a weighted incremental PCA algorithm for subspace learning. Hall's subspace learning method [16] updates the sample mean sequentially. Lim et al. [26] extend Hall's method by updating the subspace with multiple samples at one time to improve its efficiency. Yu et al. propose a *Multiple Linear Subspaces* (MLS) model [42] by constructing multiple local subspaces to describe the appearance of the target. However, they ignore the energy dissipation in updating process. Nguyen et al. use the *Incremental Probabilistic Principal Component Analysis* (IPPCA) [30] method to represent the appearance of multiple targets, which ignores the relationships between the target appearances in consecutive frames. Please refer to the survey [27] for more multi-linear subspace learning methods.

Different from the existing online subspace learning methods mentioned above, the proposed method considers the

Algorithm 1 Online Subspace Learning Algorithm

Input: $(\mathcal{I}_t, U, \Omega, \tau)$
$\mathcal{I}_t$: image patches (updating samples) collected at the target optimal state Z_t^* of frame t;
U: the set of unprocessed image patches, and symbol $|U|$ represents the number of unprocessed samples;
$\Omega = \emptyset$: the initial learned subspace set;
τ: the required number of samples used to construct the updating subspace.
Output: Ω: the learned subspace set.
1: **if** $|U| < \tau$ **then**
2: Add $\mathcal{I}_t$ to the unprocessed sample set U.
3: **else**
4: **if** $\Omega = \emptyset$ **then**
5: Construct the updating subspace $\widetilde{\Omega}$ with the samples in the set U, Add $\widetilde{\Omega}$ to Ω and clear the unprocessed sample set U.
6: **else**
7: Construct the updating subspace $\widetilde{\Omega}$ by the samples in unprocessed sample set U, and use the subspace $\widetilde{\Omega}$ to update the learned subspace Ω (merge $\widetilde{\Omega}$ and Ω). Clear the unprocessed sample set U.
8: **end if**
9: **end if**

relationships between consecutive frames and the power dissipation in updating process, to capture the subspace of target appearance more accurately. The framework of the proposed online subspace learning algorithm is detailed in Alg. 1. Let $\mathbf{\Omega} = (\psi, V, \Lambda, \sigma^2, W, d^*, n)$ be the appearance model of the target, where ψ, V, Λ, σ^2, W, d^* and n are the sample mean vector, eigenvectors, eigenvalues, power dissipation, sample weights set, reduced dimension and number of samples used to construct the subspace, respectively. The pixel values are used as the features.

A. Temporal Context Energy

Let O_t be the observation, and $\mathcal{I}_t \in \mathbb{R}^d$ be the reshaped vector consist of the pixel values in the image patch corresponding to the state Z_t, where d is the feature dimension. The temporal context energy $U(Z_t|O_t)$ of the candidate state Z_t is calculated as:

$$U(Z_t|O_t) = \frac{\varepsilon(Z_t, O_t)^2}{2\sigma_t^2} + (d - d^*)\log\sigma_t + \sum_{i=1}^{d^*}\Big(\frac{G_{i,t}(Z_t, O_t)^2}{2\lambda_{i,t}} + \frac{1}{2}\log\lambda_{i,t}\Big), \quad (3)$$

where d^* is the reduced dimension of the subspace. σ_t is the power dissipation in dimension reduction of the subspace. $\lambda_{i,t}$ is the i^{th} eigenvalue of the subspace in descending order at time t. The projection vector is defined as $G_t(Z_t, O_t) = (G_{1,t}(Z_t, O_t), \ldots, G_{d^*,t}(Z_t, O_t)) = V_t^T(\mathcal{I}_t - \psi^{(t)})$, where V_t represents the eigenvectors of the learned subspace at time t. $\varepsilon(Z_t, O_t)$ is the residual of the original samples projected to the subspace, that is $\varepsilon(Z_t, O_t) = \|\mathcal{I}_t - V_t V_t^T \mathcal{I}_t\|_2$.

B. Online Subspace Learning

The temporal part in our tracker models the target appearance as an online subspace model, the core of which is the updating strategy. The updating process is presented as follows.

1) Subspace Construction: The subspace construction can be done by the standard *Eigenvalue Decomposition* (EVD) algorithm and the power dissipation rate is employed to determine the reduced dimension d^* of the process, that is:

$$d^* = \arg\min_k \left\{ k \Big| \frac{\sum_{j=1}^k \lambda_j}{\sum_i \lambda_i} \geq \eta \right\}, \quad (4)$$

where λ_j is the j^{th} eigenvalue of the constructed subspace in descending order.

2) Subspace Merge: In our online subspace learning method, the newly constructed subspace is used to update the existing model. The core problem in updating process is how to merge two subspaces into a new one. Unlike Hall's [16] and its variant [26], the proposed subspace learning strategy updates the energy dissipation, which achieves better description ability to recognize the target. Given two subspaces $\mathbf{\Omega}_k = (\psi^{(k)}, V^{(k)}, \Lambda^{(k)}, \sigma_k^2, W_k, d_k^*, n_k)$ and $\mathbf{\Omega}_l = (\psi^{(l)}, V^{(l)}, \Lambda^{(l)}, \sigma_l^2, W_l, d_l^*, n_l)$, we aim to get the merged subspace $\mathbf{\Omega}_{k+l} = (\psi^{(k+l)}, V^{(k+l)}, \Lambda^{(k+l)}, \sigma_{k+l}^2, W_{k+l}, d_{k+l}^*, n_{k+l})$. We set the eigenvalue matrix $\Lambda^{(k)} = diag(\lambda_{1,k}, \ldots, \lambda_{d_k^*,k})$ and $\Lambda^{(l)} = diag(\lambda_{1,l}, \ldots, \lambda_{d_l^*,l})$, and the eigenvector matrix $V^{(k)} = [v_{1,k}, \ldots, v_{d_k^*,k}]$ and $V^{(l)} = [v_{1,l}, \ldots, v_{d_l^*,l}]$, where $v_{i,k}$ and $v_{i,l}$ are the i^{th} eigenvectors of the subspaces $\mathbf{\Omega}_k$ and $\mathbf{\Omega}_l$ respectively. Obviously, it is easy to get the weight set of the merged subspace $W_{k+l} = W_k \bigcup W_l$ and the number samples $n_{k+l} = n_k + n_l$. In the following, we mainly discuss the way to get $\psi^{(k+l)}$, $V^{(k+l)}$, $\Lambda^{(k+l)}$, σ_{k+l}^2 and d_{k+l}^* of the merged subspace. In updating, we weight the samples to indicate the probability belonging to the positive samples. Intuitively, large weights should be assigned to the samples with less noise and small ones to the samples with more noise. Taking the subspace $\mathbf{\Omega}_k$ as an example, the mean value $\psi^{(k)}$, power dissipation σ_k^2, and covariance matrix $S^{(k)}$ of the updating samples at time k are represented as:

$$\psi^{(k)} = \frac{1}{\sum_{\omega_i \in W_k} \omega_i} \sum_{\omega_i \in W_k} \omega_i \mathcal{I}_i, \quad (5)$$

$$\sigma_k^2 = \frac{1}{d - d_k^*} \sum_{i=d_k^*+1}^{d} \lambda_{i,k}, \quad (6)$$

$$S^{(k)} = \frac{1}{\sum_{\omega_i \in W_k} \omega_i^2} \sum_{\omega_i \in W_k} \omega_i^2 (\mathcal{I}_i - \psi^{(k)})(\mathcal{I}_i - \psi^{(k)})^T, \quad (7)$$

where $\mathcal{I}_i \in \mathbb{R}^d$ is the i^{th} updating sample, generated by vectorizing the image patch corresponding to the optimal target state Z_i^* at time i. ω_i is the weight of the i^{th} updating sample and W_k is the weight set of the samples at time k. The mean value $\psi^{(l)}$, power dissipation σ_l^2, and covariance matrix $S^{(l)}$ of the subspace $\mathbf{\Omega}_l$ are similarly defined.

According to the mean value representation in Equ. (5), it is easy to get the mean value $\psi^{(k+l)}$ of the merged subspace $\mathbf{\Omega}_{k+l}$, that is

$$\psi^{(k+l)} = \gamma\, \psi^{(k)} + (1 - \gamma)\psi^{(l)}, \quad (8)$$

where $\gamma = \frac{\sum_{\omega_i \in W_l} \omega_i}{\sum_{\omega_i \in W_{k+l}} \omega_i}$.

According to the covariance matrix representation in Equ. (7), we have:

$$S^{(k+l)} \approx \rho S^{(k)} + (1-\rho)S^{(l)} + yy^T, \tag{9}$$

where

$$\rho = \frac{\sum_{\omega_i \in W_k} \omega_i^2}{\sum_{\omega_i \in W_{k+l}} \omega_i^2},$$
$$y = \left(\rho(1-\gamma)^2 + (1-\rho)\gamma^2\right)^{\frac{1}{2}}(\psi^{(k)} - \psi^{(l)}).$$

Furthermore, based on the covariance matrix decomposition in [30], the matrix $S^{(k)}$ can be represented as $S^{(k)} = \sigma_k^2 I + \sum_{i=1}^{d_k^*}(\lambda_{i,k} - \sigma_k^2)v_{i,k}v_{i,k}^T$, where d_k^* is the reduced dimension corresponding to $S^{(k)}$. The covariance matrix $S^{(l)}$ is similarly decomposed. Then, by plugging them into Equ. (9), we have:

$$S^{(k+l)} \approx (\rho\sigma_k^2 + (1-\rho)\sigma_l^2)I + \sum_{i=1}^{d_k^*}\rho(\lambda_{i,k} - \sigma_k^2)v_{i,k}v_{i,k}^T + \sum_{i=1}^{d_l^*}(1-\rho)(\lambda_{i,l} - \sigma_l^2)v_{i,l}v_{i,l}^T + yy^T,$$

where $v_{i,k}$, $\lambda_{i,k}$, σ_k are the i^{th} eigenvector, i^{th} eigenvalue, and the power dissipation of the subspace Ω_k, and $v_{i,l}$, $\lambda_{i,l}$, σ_l are the corresponding ones of the subspaces Ω_l. By denoting

$$L = [\sqrt{\rho(\lambda_{1,k} - \sigma_k^2)}v_{1,k}, \ldots, \sqrt{\rho(\lambda_{d_k^*,k} - \sigma_k^2)}v_{d_k^*,k},$$
$$\sqrt{(1-\rho)(\lambda_{1,l} - \sigma_l^2)}v_{1,l}, \ldots, \sqrt{(1-\rho)(\lambda_{d_l^*,l} - \sigma_l^2)}v_{d_l^*,l}, y],$$

it is easy to get the following equation:

$$S^{(k+l)} \approx \left(\rho\sigma_k^2 + (1-\rho)\sigma_l^2\right)I + LL^T.$$

In tracking application, the feature dimension d is always considerably large, which makes it impossible to decompose the matrix LL^T directly due to the computational complexity. Therefore, we decompose the matrix L^TL instead of LL^T to get the equation $L^TL = U\Gamma U^T$, where $\Gamma = diag\{\xi_1, \xi_2, \ldots, \xi_q\}$ is the eigenvalue matrix sorted in descending order, $q = d_k^* + d_l^* + 1$, and $U^TU = I$. The corresponding eigenvectors of the matrix LL^T are represented as $V_q = LU\Gamma^{-\frac{1}{2}}$. Obviously, the eigenvectors of the covariance matrix $S^{(k+l)}$ are the same as the matrix LL^T. By setting $V_q = [v_1, \ldots, v_q]$, the covariance matrix is represented as:

$$S^{(k+l)} = \left(\rho\sigma_k^2 + (1-\rho)\sigma_l^2\right)I + \sum_{i=1}^{q}\xi_i v_i v_i^T.$$

We get the reduction dimension of the merged subspace Ω_{k+l} according to the η-truncation condition

$$d_{k+l}^* = \arg\min_i \left\{ i \Big| \frac{\sum_{j=1}^{i}\left(\rho\sigma_k^2 + (1-\rho)\sigma_l^2 + \xi_j\right)}{\sum_{j=1}^{q}\xi_j} \geq \eta \right\}.$$

The eigenvalues of the merged subspace Ω_{k+l} are calculated as $\lambda_{i,k+l} = \rho\sigma_k^2 + (1-\rho)\sigma_l^2 + \xi_i$, $i = 1, \ldots, d_{k+l}^*$, and the eigenvectors $V^{(k+l)}$ of the merged subspace are obtained by getting the first d_{k+l}^* vectors of the matrix V_q. Finally, the power dissipation is updated

$$\sigma_{k+l}^2 = \frac{1}{d - d_{k+l}^*}\sum_{i=d_{k+l}^*+1}^{q}\xi_i + (\rho\sigma_k^2 + (1-\rho)\sigma_l^2).$$

IV. Local Spatial Context Model

Local context has been proved effective in visual tracking task [10], [14], [23], [31], [36], [40]. Different from previous works, our strategy focuses on the contributors and their weak correlations to the target, and then combines them to construct a strong classifier to locate the target. Multiple instance boosting [5], [37] is used to build the strong supporting field by selecting the most representative contributors, which is easy to complete.

A. Spatial Context Energy

In multiple instance boosting, each selected weak classifier corresponds to each weak correlation. The selected correlations are combined together to evaluate the spatial context energy (namely the spatial energy function in Equ. (2)) of a candidate state. The spatial context energy is expressed as:

$$U(Z_t|f^r(\cdot), O_t) \propto -\sum_j g_j^t(\mathbf{x}^t), \tag{10}$$

where $f^r(\cdot)$ is the contributor state set, Z_t is the target candidate state, O_t is the corresponding observation, $g_j^t(\mathbf{x}^t)$ is the j^{th} selected weak classifier at time t and $\mathbf{x}^t$ is the corresponding relation feature of the candidate state. Without ambiguity, we omit the time index t in the following. Please refer to [5], [37] for more details about the multiple instance boosting.

B. Contributor State Set $f^r(\cdot)$

As defined above, the patch at the key point is called contributor. Here, the SURF descriptor [6] is employed as the key points around the target, which is generated in the rectangle (the green rectangle shown in Fig. 2) centered at the target center with the width $r_e \cdot w$ and height $r_e \cdot h$, where r_e is the enlargement factor. We set the enlargement factor $r_e \in [0.5, 1.6]$ in our experiments. w and h are the width and height of the target in the current frame. If the extracted candidate key points are more than the required ones, we randomly select some of them to be the final key points and use them to generate the contributors; but if they are inadequate, we randomly generate some other points in the rectangle to supplement them.

C. Relation Feature Construction

To incorporate the structure information of the target, we partition the regions of the target and contributors into a predefined number of blocks. Let $N = n_1 \times n_2$ be the predefined number of partitioned blocks, where n_1 and n_2 are the partitioned numbers of blocks in the row and column respectively. We set $n_1 = 5$ and $n_2 = 5$ in our experiments.

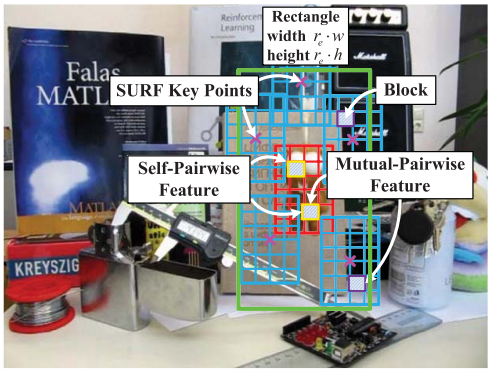


Fig. 2. The illustration of feature construction in the spatial context model of our tracker. The red rectangle represents the target area and the blue rectangles represent the contributors. The purple crosses represent the center of the contributors, which are generated in the green rectangle. The width and height of the green rectangle are $r_e \cdot w$ and $r_e \cdot h$ respectively. The small yellow rectangles represent the target blocks and the purple ones represent the background blocks.

The structure information is integrated by modeling the relationships between blocks. Let $I(x, y)$ be the pixel value of the image at position (x, y), $b_i(Z)$ be the i^{th} block of the target corresponding to target state Z, and $b_i(f_k^r(Z))$ be the i^{th} block of the k^{th} contributor corresponding to the contributor state $f_k^r(Z)$. The weak relation function $d_f\big(b_p(\cdot), b_q(\cdot)\big)$ between two blocks is defined as:

$$d_f\Big(b_p(\cdot), b_q(\cdot)\Big) = \sum_{(i,j)\in b_p(\cdot)} I(i, j) - \sum_{(i,j)\in b_q(\cdot)} I(i, j).$$

Next, we collect all of these weak relations to construct a relation feature pool $\mathcal{F}$.

The structure information comes from two parts: one is the mutual-pairwise features between the corresponding blocks of the target and the contributors, and the other one is the self-pairwise features between the inner blocks of the target itself. Let $\mathcal{F}_s$ be the self-pairwise feature pool and $\mathcal{F}_m$ be the mutual-pairwise feature pool. We get the relation feature pool $\mathcal{F} = \mathcal{F}_s \cup \mathcal{F}_m$. Specifically, the self-pairwise and mutual-pairwise feature pools are constructed as

$$\mathcal{F}_s = \Big\{ d_f\big(b_i(Z), b_j(Z)\big)\big|_{i\neq j}^{i,j=1,\ldots,N} \Big\},$$
$$\mathcal{F}_m = \Big\{ d_f\big(b_i(Z), b_j(f_k^r(Z))\big)\big|_{k=1,\ldots,m_c}^{i,j=1,\ldots,N} \Big\}.$$

Fig. 2 shows the feature construction process of our tracker.

D. Weak Classifier

We use the online updating *Gaussian Mixture Model* (GMM) to estimate the posterior probability of the weak classifier, that is

$$P(\mathbf{x}_j|y) = \sum_{i=1}^{K} \omega_{ij}(y)\eta(\mathbf{x}_j, \mu_{ij}(y), \sigma_{ij}(y)),$$

where K is the number of Gaussian models, $\omega_{ij}(y)$, $\mu_{ij}(y)$, and $\sigma_{ij}(y)$ are the weight, mean and variance of the i^{th} Gaussian model of the sample with label y (positive or negative), respectively. $\mathbf{x}$ is the constructed feature with dimension $(m+1)\times N^2$ described in Sec. IV-C. $\mathbf{x}_j$ is the j^{th} dimension of $\mathbf{x}$ and y is the label of $\mathbf{x}$. Obviously, we have the relation $\sum_{i=1}^{K} \omega_{ij} = 1$. Furthermore, $\eta(x, \mu, \sigma)$ is the probability density function of the Gaussian distribution, that is

$$\eta(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} \right).$$

Assuming that the positive and negative samples have equal prior probability in the task, *i.e.* $P(y = +1) = P(y = -1)$, it is easy to get the continuous Bayesian weak classifier based on the GMM, that is

$$g_j(\mathbf{x}) = \log \frac{P(\mathbf{x}_j|y = +1)}{P(\mathbf{x}_j|y = -1)}.$$

When the weak classifier receives the samples $\big\{(\mathbf{u}^{(1)}, y^{(1)}), (\mathbf{u}^{(2)}, y^{(2)}), \ldots, (\mathbf{u}^{(n)}, y^{(n)})\big\}$, the GMM of both the positive and negative samples will be updated by the following steps. Firstly, we calculate the similarity between the j^{th} dimension of the received positive samples and the k^{th} Gaussian models to get the matching measure criterion $\mathcal{H}_{kj}$, that is:

$$\mathcal{H}_{kj} = \omega_{kj}(+1)\Big(\prod_{i|y^{(i)}=1} \eta(\mathbf{u}_j^{(i)}, \mu_{kj}(+1), \sigma_{kj}(+1)) \Big)^{\frac{1}{n}},$$

where $\mathbf{u}_j^{(i)}$ is the j^{th} dimension of $\mathbf{u}^{(i)}$. Let M_{kj} be the symbol indicating whether the k^{th} Gaussian model matches the j^{th} dimension of the feature. M_{kj} is defined as

$$M_{kj} = \begin{cases} 1 & \mathcal{H}_{kj} > \mathcal{H}_{lj}; \quad l = 1, \ldots, K; \quad l \neq k \\ 0 & \text{Otherwise} \end{cases},$$

where $k = 1, \ldots, K$. Then, if $M_{kj} = 1$, which means successful match, the mean and variance of the matched Gaussian model will be updated as follows:

$$\mu_{kj}(+1) = (1-\tilde{\lambda})\mu_{kj}(+1) + \tilde{\lambda}\frac{1}{n}\sum_{i|y^{(i)}=1} \mathbf{u}_j^{(i)},$$
$$\sigma_{kj}^2(+1) = (1-\tilde{\lambda})\sigma_{kj}^2(+1) + \tilde{\lambda}\Big(\frac{1}{n}\sum_{i|y^{(i)}=1} (\mathbf{x}_j^{(i)} - \mu_{kj}(+1))^2\Big)^{\frac{1}{2}},$$

where $\tilde{\lambda}$ is the updating step. Otherwise, the mean and variance of the unmatched ones will not be updated. Finally, all the weights are updated as $\omega_{kj}(+1) = (1-\tilde{\lambda})\omega_{kj}(+1) + \tilde{\lambda}M_{kj}$, where $k = 1, 2\cdots, K$. Meanwhile, the updating rule of the negative samples is similarly defined. The updating step $\tilde{\lambda}$ is set to 0.4 in our experiments.

V. Experiments

This section consists of two parts. In the first part, we analyze the influence of the parameters in the proposed online subspace learning strategy and present its superiority over other state-of-the-art methods. In the seconde part, we evaluate the effectiveness of the proposed spatio-temporary context model based tracker.

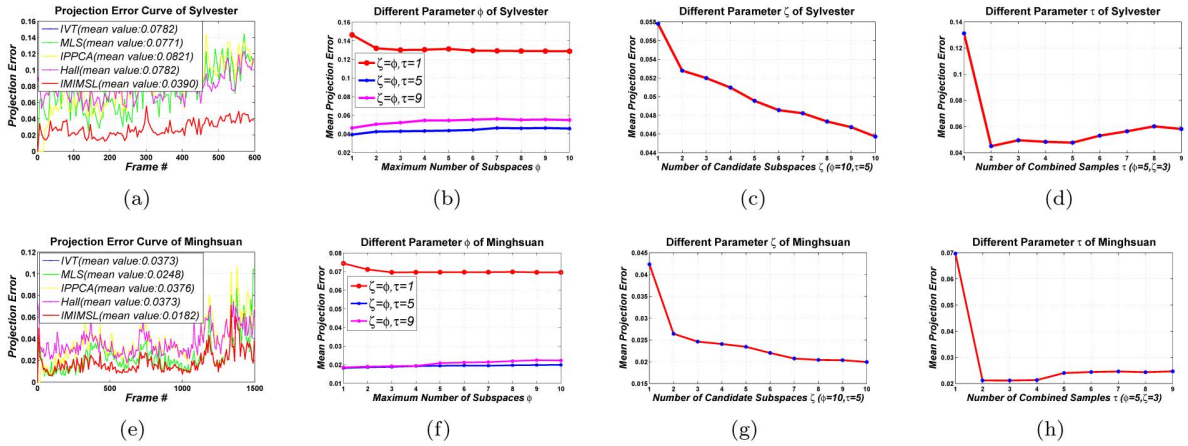


Fig. 3. The comparison curves of the subspace experiments. (a) and (e) are the comparison of our methods with other state-of-the-art methods. (b) and (f) are the mean reconstruction error curves with fixed ζ and τ. (c) and (g) are the mean reconstruction error curves with fixed ϕ and τ. (d) and (h) are the mean reconstruction error curves with fixed ϕ and ζ.

A. Evaluation of Online Subspace Learning

The subspace learning method presented in Sec. III can be naturally extended to the multiple subspaces learning method. We can use several subspaces to describe the target appearance. In multiple subspaces learning, the similarity between two local subspaces is estimated as the weighted combination of the angle measure $Sim_\alpha(\Omega_p, \Omega_q)$ and the compactness measure $Sim_c(\Omega_p, \Omega_q)$, *i.e.* $Sim(\Omega_p, \Omega_q) = \kappa_s Sim_\alpha(\Omega_p, \Omega_q) + (1-\kappa_s) Sim_c(\Omega_p, \Omega_q)$. In our experiments, the balance parameter is set $\kappa_s = 0.15$. Please refer to [7], [42] for more details about the similarity calculation. The multiple subspaces learning algorithm is detailed in Appendix. The temporal context energy of a candidate state given out by the multiple subspaces model is defined as

$$U(Z_t|O_t) = \max_{\ell \in \mathcal{L}} U_\ell(Z_t|O_t),$$

where $\mathcal{L} = \{\ell_1, \ldots, \ell_\zeta\}$ is the index set of the ζ subspaces, which have the smallest mean value distance with the candidate sample. Obviously, $\zeta \in \{1, \ldots, \phi\}$, where ϕ is the total number of subspaces. $U_\ell(Z_t|O_t)$ is the energy given out by the ℓ^{th} subspace, which is calculated in Equ. (3).

To evaluate the performance of the proposed subspace learning method (*Incremental Multiple Instance Multiple Subspace Learning*, abbreviated as IMIMSL), we use the manually labeled target samples from two tracking videos (*Sylvester* and *Minghsuan* [26]) to train the model sequentially and evaluate it on-the-fly. The publicly available sequences *Sylvester* contains severe pose changes while *Minghsuan* presents a challenging lighting condition. The $\ell 2$-norm of the sample reconstruction error is exploited to evaluate the model. As mentioned above, the core parameters in our proposed model include: the total number of subspaces ϕ, the candidate number of nearest subspaces to give out the energy of candidate states ζ and the number of combined samples τ. For simplicity, we use the symbol ϕ-ζ-τ to represent model with these parameters.

1) Influence of the Model Parameters: We choose different parameters for the model, and calculate the mean reconstruction error of these two sequences.

Parameter $\boldsymbol{\phi}$ To exclude the influence of the parameters τ and ζ, we carry out multiple experiments with different values of the parameters ($\tau \in \{1, 5, 9\}$, $\zeta = \phi$). For each experiment, we fix τ and ζ and change ϕ to evaluate the mean reconstruction error in both sequences. According to the results in Fig. 3(b) and (f), we have the following conclusions:

- when $\tau = 1$, the performance of the model will be improved as ϕ increases;
- when $\tau > 1$, the performance of the model will not be improved as ϕ increases, but is comparable with $\phi = 1$;
- The mean reconstruction error of $\tau > 1$ is remarkably lower than $\tau = 1$.

Therefore, it is easy to see that the performance of the model is improved remarkably by the sample combination in updating, rather than the introduction of multiple subspaces, which has not been pointed out in previous literatures.

Parameter $\boldsymbol{\zeta}$ As shown in Fig. 3(c) and (g), we fix $\phi = 10$, and every $\tau = 5$ samples are combined together for updating. The number of candidate subspaces ζ is set in the interval $[1, 10]$, *i.e.* $\zeta \in [1, 10]$, to give out the reconstruction error of the samples. It is clear that the reconstruction error of the samples decreases as ζ increases. The result indicates that when the number of candidate subspaces increases, the performance of the model will be improved. Obviously, the more subspaces are exploited to represent the sample, the smaller reconstruction error will be achieved, but the computational complexity will be increased simultaneously.

Parameter $\boldsymbol{\tau}$ To indicate the influence of τ, we fix $\phi = 5$ and $\zeta = 3$, and change τ. The mean reconstruction error curves for both sequences are shown in Fig. 3(d) and (h). The results indicate that the combined samples for updating can greatly enhance the performance of the subspace model. Meanwhile, the proper selection of τ is very important and the optimal value of τ is determined by the data distribution. If τ is too small, the computational complexity will be high and it will not improve the performance remarkably. Meanwhile, if τ is too large, the local property of the subspaces will be destroyed, and the noise contained in them will not be eliminated effectively, which will decrease the performance of the model.

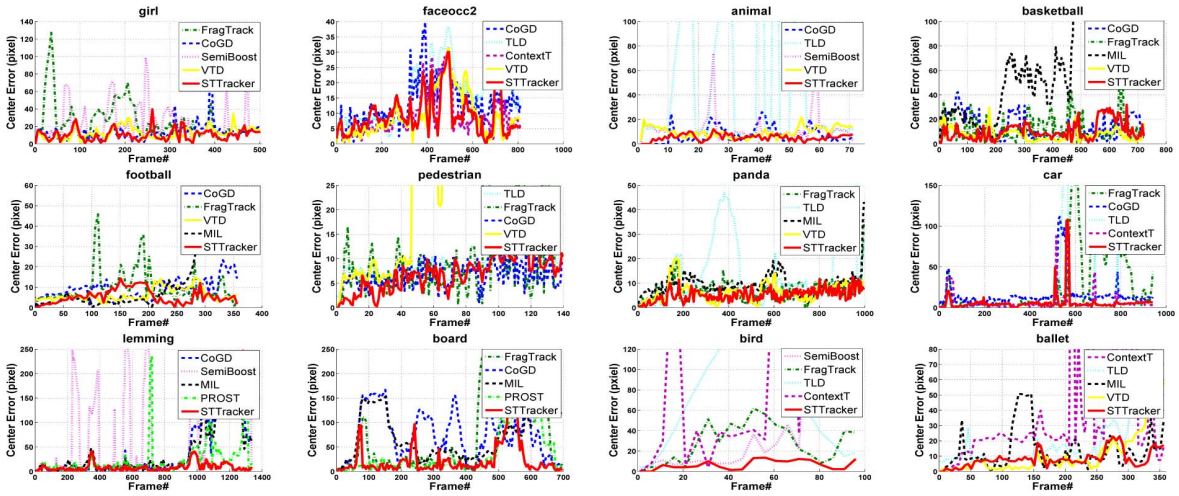


Fig. 4. Target tracking results of our tracker, FragTrack [1], SemiBoost [13], CoGD [42], MIL [5], PROST [34], TLD [19], VTD [21] and ContextT [10]. The results of five trackers with relatively better results in each sequence are displayed.

2) Comparison With Other Learning Methods: In addition, we compare our proposed online subspace learning method with some other state-of-the-art learning methods. Each model is initialized by 5 samples. The remaining samples are used to evaluate the model at first, and then used to update the model online. All the sample weights are set to be 1 because these samples are manually labeled as positive. In our model, we set $\phi = 1$, $\zeta = 1$ and $\tau = 5$. The comparison is conducted with other four state-of-the-art learning strategies: Hall's [16], IVT [26], MLS [42] and IPPCA [30]. The parameters of our model are fixed in both evaluation sequences. The parameters of Hall's, IVT, MLS and IPPCA are set as the default ones in their papers or codes. The comparison curves are shown in Fig. 3(a) and (e). In both sequences, IVT and IPPCA, which only construct a single subspace updated with a single sample, have the worst performances. With the introduction of multiple subspaces updated with multiple samples, MLS outperforms IVT and IPPCA. However, its results are not as accurate as ours since our IMIMSL considers the energy dissipation of dimension reduction during the updating. Although Hall's and MLS combine multiple samples for updating, the ignorance of the energy dissipation causes its unsatisfactory performance in both sequences.

B. Evaluation of Target Tracking

1) Experiment Setup: 12 sequences are used to evaluate the performance of STT. STT is implemented in C++ code and runs on a Intel 2.4GHz PC platform about 0.5 to 4 fps. These 12 sequences contain different challenging conditions, 11 of which are publicly available, and the other one is collected by ourselves. Our tracker is initialized with the first frame and outputs the trajectory of the target. The quantitative comparison results of IVT [26], FragTrack [1], SemiBoost [13], CoGD [42], MIL [5], PROST [34], VTD [21], TLD [19], ContextT [10] and STT are shown in Fig. 4, Table II and III, respectively. Some visual results are shown in Fig. 6. The codes and data used in this paper can be found on our website.[1] Table II is the comparison results of *Average Center Error in Pixels* (ACEP)(↓) criteria and Table III is the comparison results of Successful Frame criteria based on the evaluation metric of PASCAL VOC (↑) object detection [11], in which the overlap ratio between the tracked bounding box and the ground truth bounding box larger than 50% is regarded as the successfully tracked. ↑ means higher scores indicate better performance, and ↓ means lower scores indicate better performance.

TABLE I
VALIDATION EXPERIMENTS OF THE PROPOSED TRACKING METHOD

Validation		ACEP criteria (↓)				Successful Frame criteria (↑)			
Seq.	Frames	STT	TCT	SCT	OMT	STT	TCT	SCT	OMT
pedestrian	**140**	**6.89**	85.1	13.9	11.2	**140**	17	49	111
lemming	**1336**	**8.45**	96.6	161	12.5	**1246**	675	225	1117
bird	**99**	**8.03**	25.9	25.2	14.6	**96**	47	45	72
ballet	**356**	**8.34**	85.5	17.8	9.78	**312**	195	245	297

2) Parameters: The search radius R of the tracker is set in the interval [20], [50], the search scale c is set to 2 and the balance parameter κ_b in Equ. (2) is set to 0.3. For the global temporal context model, only one subspace is used to represent the target for the consideration of computational complexity. Meanwhile, every 5 frames are combined together to update the linear subspace model, *i.e.* $\tau = 5$, and η is set to 0.99 in subspace construction (Equ. (4)). We resize the image patches of the target corresponding to the optimal state in each frame into the standard size 20×20 and vectorize them to get the updating samples with the feature dimension $d = 400$. For the local spatial context model, 12 contributors are generated to construct the supporting field around the target, *i.e.* $m_c = 12$, and each of them is partitioned into 5×5 blocks. 3 Gaussian components are used for positive and negative samples, that is $K = 3$. 350 weak relations are combined together to construct the supporting field. For the positive bags, 45 samples are collected from the circle region with the radius 8. For the

[1] http://www.cbsr.ia.ac.cn/users/lywen/

TABLE II
COMPARISON RESULTS OF ACEP CRITERIA

Seq.	STT	IVT	CoGD	Semi	MIL	Frag	PROST	VTD	TLD	ContextT
girl	**10.4**	40.4	14.1	22.8	31.6	25.4	19.0	12.5	35.7	18.6
faceocc2	9.39	19.7	13.3	25.2	30.7	21.5	17.2	9.40	14.9	**9.25**
animal	**5.20**	226	7.38	12.3	80.3	71.4	-	9.68	50.7	81.2
basketball	**10.5**	95.4	13.8	153	93.3	12.7	-	11	158	159
football	**6.15**	17.2	9.16	102	12.7	9.92	-	6.25	13.0	51.2
pedestrian	6.89	109	**6.75**	30.3	40.3	11.5	-	62.6	8.75	61.5
panda	**5.20**	58.2	64.5	41.7	9.42	6.85	-	6.33	17.7	77.5
car	6.26	56.9	16.6	46.4	80.7	28.6	-	51.8	11.8	**5.47**
lemming	**8.45**	128	39.8	99.8	40.5	82.8	25.1	98	167	182
board	**23.9**	169	74.5	389	69.2	90.1	39.0	70.1	134	103
bird	**8.03**	125	135	68.8	128	29.4	-	148	76.9	105
ballet	8.34	**8.21**	45.8	102	13.8	10.8	-	11.8	33.1	55.0

TABLE III
COMPARISON RESULTS OF SUCCESSFUL FRAME CRITERIA

Seq.	Frames	STT	IVT	CoGD	Semi	MIL	Frag	PROST	VTD	TLD	ContextT
girl	**502**	497	353	482	388	378	378	447	**502**	219	328
faceocc2	**812**	**797**	583	767	548	379	618	665	792	712	687
animal	**71**	**71**	3	62	56	5	13	-	66	43	48
basketball	**725**	**715**	75	335	90	175	630	-	601	15	50
football	**362**	346	246	292	65	272	302	-	**357**	272	55
pedestrian	**140**	**140**	4	135	35	71	92	-	45	80	27
panda	**1000**	**580**	120	175	375	195	465	-	510	315	300
car	**945**	**915**	414	804	504	101	644	-	571	878	896
lemming	**1336**	**1246**	284	907	733	882	733	942	471	234	40
board	**698**	**583**	30	279	105	354	474	524	274	95	60
bird	**99**	**96**	2	47	47	10	33	-	13	10	24
ballet	**356**	**312**	307	203	35	277	287	-	277	232	64

negative bags, 50 samples are collected from the ring region with the inner radius 12 and outer radius 40. The energy thresholds in our experiments are set as $\theta_s \in [-20, -10]$ and $\theta_t \in [10, 20]$. For the other trackers cited here, we use the default parameters provided in the public available codes, and choose the best one in 5 runs, or take the results directly from the published papers. Specifically, we reproduce the CoGD tracker in C++ code and adopt the parameters as described in [42].

3) Efficiency Validation: In this part, we evaluate the effectiveness of the temporal and spatial parts in our tracker. We construct three trackers: the temporal part of our tracker, denoted as *Temporal Context Tracker* (TCT); the spatial part of our tracker, denoted as *Spatial Context Tracker* (SCT); and the *Online subspaces learning combined with Multiple instance learning Tracker* (OMT). The OMT is constructed by replacing the spatial part of our tracker with the Haar feature based online learning classifier [5] and the other parts remain the same. The same parts between these three trackers and our STT adopt the same parameters. We test them in four sequences with both the ACEP criteria and the Successful Frame criteria. The performance comparison is shown in Table I. The results indicate that the combination of spatial and temporal parts can greatly enhance the performance of the tracker. These two parts help each other to estimate the precise state of the target and acquire the accurate updating samples for more robust performance. We notice that OMT works better than SCT and TCT because it considers both the target appearance variation and the background information. STT outperforms OMT in all those four sequences, which indicates the effectiveness of considering the relationships between the target and its surroundings.

4) Comparison With Other Trackers: In this part, we compare and analyze the tracking performance of STT and other state-of-the-art trackers in different challenging situations.

Heavy Occlusion In sequences *car* and *faceocc2*, long-term heavy occlusion occurs several times. IVT, which uses holistic appearances without spatial context information, fails to track the target in this case. Comparatively, TLD and ContextT perform well in these two sequences, because the detection based trackers are able to re-locate the target after the occlusion. With spatio-temporal appearance contextual information, STT also has good performance which indicates the robustness of STT to heavy occlusion. In sequence *girl*,

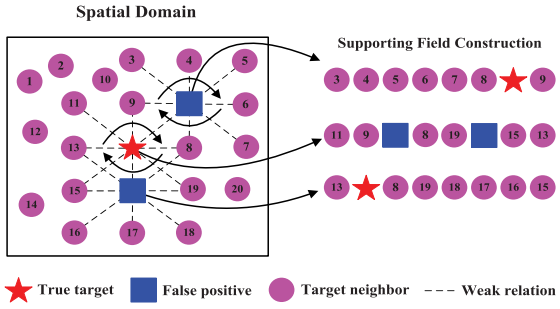


Fig. 5. The red pentagram represents the true target position, the blue triangle represents the false positive in the background and the magenta circle represents other surrounding patches. The dashed line represents the weak correlation between the target and its surroundings. The relation between the target and its surroundings can greatly enhance the discriminability of the tracker.

a similar object occludes the target, as shown in the frame 463 of Fig. 6. In this case, TLD and MIL drift away for the full occlusion of the man's face. For STT, because of the contextual information around the target and the temporal constraint, it exhibits strong discriminative ability and is able to track the target correctly.

Abrupt Motion and Motion Blur In sequence *pedestrian*, there is abrupt motion because of the hand-held camera. In sequence *bird*, the bird changes its moving direction abruptly in the frame 48. Many trackers including IVT, VTD, TLD, fail to track the target in these two cases. When the abrupt motion happens, the temporal information becomes unreliable while the spatial information is still discriminative. Therefore, STT, which combines the temporal and spatial information, is able to predict the position of the target accurately. In sequences *animal* and *lemming*, there exist motion blur, which loses important texture information. Trackers like FragTrack, SemiBoost and TLD fail to track the targets in this case. The proposed STT, with the help of low dimensional 'gist' in temporal model and the contextual information in spatial model, achieves the best performance in these two sequences.

Cluttered Background The cluttered background in sequences *animal*, *football* and *ballet* actually confuses the tracker substantially, as shown in Fig. 6. MIL is easily hijacked by other objects that have similar appearance with the target. Although TLD considers positive and negative constraints and ContextT incorporates semantic context, they

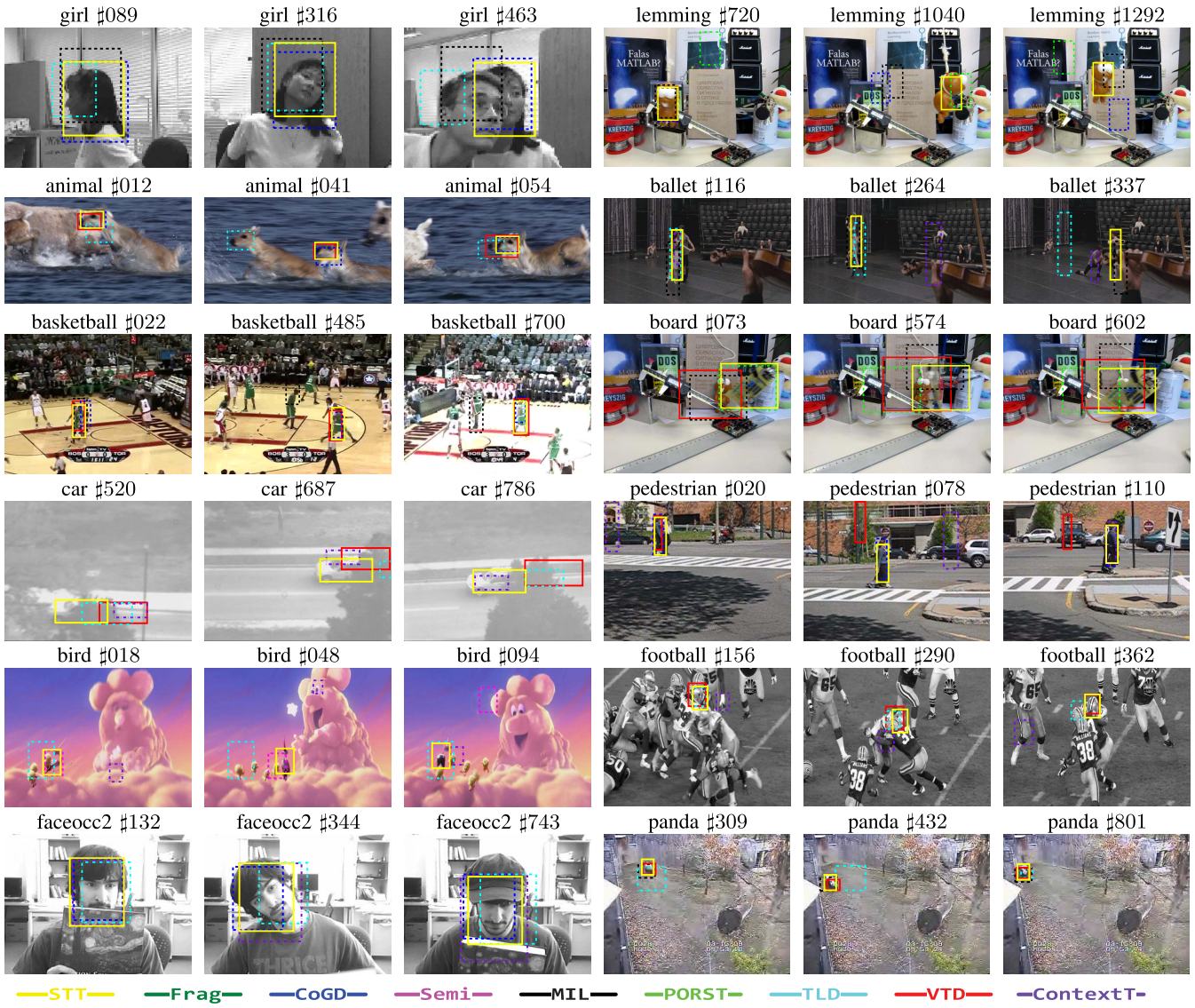


Fig. 6. Tracking results of different trackers. Only the trackers with relatively better performance are displayed.

still frequently skip to other objects because they depend too much on detectors. The complex background in sequences *board* and *lemming* significantly increases the difficulty in tracking task. Many trackers including FragTrack, IVT and VTD, which ignore background information, perform badly in these sequences. Although CoGD, MIL, and PROST take the background into account, their performances are not as accurate as STT. Particularly, we notice that STT is very good at dealing with the distraction by other similar appearance objects due to the consideration of both spatial and temporal appearance constraints. As shown in Fig. 5, when there exists a spatially close object with similar appearance of the target, the surroundings of these two objects are totally different. Once we incorporate the surrounding information around the target to build the supporting field, it is easy to differentiate the target from the the spatially close object with similar appearance. The mutual supervision of spatio-temporal appearance context ensures the stability of STT.

Large Variation of Pose and Scale Some trackers such as FragTrack do not update their model effectively and are easy to lose the target when 3D pose of the target changes dramatically, as seen in sequences *girl*, *board*, *lemming*, *bird*, *ballet*, *panda*, *basketball* and *bird*. IVT, CoGD, and MIL adopt online updating mechanism to learn different appearances of the target, but they may drift away due to the large pose variation and can never recover. TLD and ContextT are good at long term surveillance sequence, but they cannot track the target precisely once large pose variation happens. Since VTD combines multiple basic models with different features of the target, it performs well in the two sequences *panda* and *basketball*. Nevertheless, its tracking performances according to the two evaluation protocols are not as satisfactory as STT, especially in sequence *bird*, as described in Table II and III.

VI. Conclusion

In this paper, a spatio-temporal context model based tracker is proposed. The appearance of the target is described by the global temporal appearance contextual information and the local spatial appearance contextual information. The structured spatial appearance context model discovers the contributors

around the target, and incorporates them to build a supporting field. To prevent the tracker from being drifted away by the surroundings, a strong temporal appearance context model is included, which describes the target with low dimensionality feature vectors. Experimental comparison demonstrates the proposed tracker outperforms the state-of-the-art tracking strategies.

APPENDIX

ONLINE MULTIPLE SUBSPACES LEARNING ALGORITHM

In this section, we present the online multiple subspaces learning algorithm in details, described in Algorithm 2.

Algorithm 2 Online Multiple Subspaces Learning Algorithm

Input: $(\mathcal{I}_t, U, \Omega, \tau, \phi)$
$\mathcal{I}_t$: image patches (updating samples) collected at the target optimal state Z_t^* of frame t;
U: the set of unprocessed image patches, and symbol $|U|$ represents the number of unprocessed samples;
$\Omega = \emptyset$: the initial learned multiple subspaces set, and symbol $|\Omega|$ represents the number of learned subspaces;
τ: the required number of samples utilized to construct the updating subspace;
ϕ: the total number of subspaces.
Output: $\Omega = (\Omega_1, \cdots, \Omega_\phi)$: multi-local subspaces.

```
if |U| < τ then
    Add I_t to the updating pool U.
else
    if |Ω| < φ then
        Construct the updating subspace Ω̃ with the samples in pool U, Add Ω̃ to Ω and clear the updating pool U.
    else
        Construct the updating subspace Ω̃ with the samples in pool U, and calculate the similarity between subspaces, that is {p*, q*} = arg max Sim(Ω_p, Ω_q), where Ω_p, Ω_q ∈ {Ω_1, ··· , Ω_φ} ⋃ {Ω̃}, p ≠ q. Ω_m = Ω_{p*} ⋃ Ω_{q*}, which means the subspace merging process, and replace the subspaces Ω_{p*} and Ω_{q*} with Ω_m. Clear the updating pool U.
    end if
end if
```

REFERENCES

[1] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE CVPR*, Jun. 2006, pp. 798–805.

[2] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1064–1072, Aug. 2004.

[3] S. Avidan, "Ensemble tracking," in *Proc. IEEE CVPR*, Jun. 2005, pp. 494–501.

[4] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, Aug. 2011.

[5] B. Babenko, M.-H. Yang, and S. J. Belongie, "Visual tracking with online multiple instance learning," in *Proc. CVPR*, 2009, pp. 983–990.

[6] H. Bay, A. Ess, T. Tuytelaars, and L. J. V. Gool, "SURF: Speeded-up robust features," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008.

[7] A. Bjoerck and G. H. Golub, "Numerical methods for computing angles between linear subspaces," *Math. Comput.*, vol. 27, no. 123, pp. 579–594, Jul. 1973.

[8] Z. Cai, L. Wen, J. Yang, Z. Lei, and S. Z. Li, "Structured visual tracking with dynamic graph," in *ACCV*, vol. 3. New York, NY, USA: Springer-Verlag, 2012, pp. 86–97.

[9] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. IEEE CVPR*, vol. 2. Jun. 2000, pp. 142–149.

[10] T. B. Dinh, N. Vo, and G. G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE CVPR*, Jun. 2011, pp. 1177–1184.

[11] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[12] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proc. IEEE CVPR*, Jun. 2006, pp. 260–267.

[13] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proc. 10th ECCV*, Oct. 2008, pp. 234–247.

[14] H. Grabner, J. Matas, L. J. V. Gool, and P. C. Cattin, "Tracking the invisible: Learning where the object might be," in *Proc. CVPR*, Jun. 2010, pp. 1285–1292.

[15] S. Gu and C. Tomasi, "Branch and track," in *Proc. IEEE CVPR*, Jun. 2011, pp. 1169–1174.

[16] P. M. Hall, A. D. Marshall, and R. R. Martin, "Adding and subtracting eigenspaces with eigenvalue decomposition and singular value decomposition," *Image Vis. Comput.*, vol. 20, nos. 13–14, pp. 1009–1016, 2002.

[17] W. Hu, X. Li, X. Zhang, X. Shi, S. J. Maybank, and Z. Zhang, "Incremental tensor subspace learning and its applications to foreground segmentation and tracking," *Int. J. Comput. Vis.*, vol. 91, no. 3, pp. 303–327, Feb. 2011.

[18] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.

[19] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *Proc. IEEE CVPR*, Jun. 2010, pp. 49–56.

[20] M. Kristan, J. Pers, S. Kovacic, and A. Leonardis, "A local-motion-based probabilistic model for visual tracking," *Pattern Recognit.*, vol. 42, no. 9, pp. 2160–2168, Sep. 2009.

[21] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proc. CVPR*, 2010, pp. 1269–1276.

[22] M. Li, W. Chen, K. Huang, and T. Tan, "Visual tracking via incremental self-tuning particle filtering on the affine group," in *Proc. IEEE CVPR*, Jun. 2010, pp. 1315–1322.

[23] X. Li, A. R. Dick, H. Wang, C. Shen, and A. van den Hengel, "Graph mode-based contextual kernels for robust SVM tracking," in *Proc. IEEE ICCV*, Nov. 2011, pp. 1156–1163.

[24] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo, "Robust visual tracking based on incremental tensor subspace learning," in *Proc. IEEE ICCV*, Oct. 2007, pp. 1–8.

[25] Y. Li, "On incremental and robust subspace learning," *Pattern Recognit.*, vol. 37, no. 7, pp. 1509–1518, 2004.

[26] J. Lim, D. A. Ross, R.-S. Lin, and M.-H. Yang, "Incremental learning for visual tracking," in *Proc. NIPS*, 2004, pp. 1–8.

[27] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognit.*, vol. 44, no. 7, pp. 1540–1551, Jul. 2011.

[28] J.-Y. Lu, Y.-C. Wei, and A. C.-W. Tang, "Visual tracking using compensated motion model for mobile cameras," in *Proc. IEEE ICIP*, Sep. 2011, pp. 489–492.

[29] X. Mei, S. K. Zhou, and F. Porikli, "Probabilistic visual tracking via robust template matching and incremental subspace update," in *Proc. IEEE ICME*, Jul. 2007, pp. 1818–1821.

[30] H. T. Nguyen, Q. Ji, and A. W. M. Smeulders, "Spatio-temporal context for robust multitarget tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 52–64, Jan. 2007.

[31] S. Palmer, "The effects of contextual scenes on the identification of objects," *Memory Cognit.*, vol. 3, no. 5, pp. 519–526, Sep. 1975.

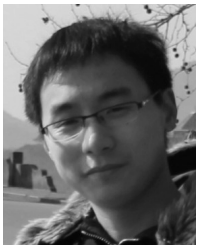
[32] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in *Proc. IEEE Comput. Soc. Conf. CVPR*, Jun. 2006, pp. 728–735.

[33] A. Saffari, M. Godec, T. Pock, C. Leistner, and H. Bischof, "Online multi-class LPBoost," in *Proc. IEEE CVPR*, Jun. 2010, pp. 3570–3577.

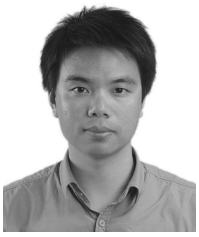
[34] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof, "PROST: Parallel robust online simple tracking," in *Proc. CVPR*, 2010, pp. 723–730.

[35] D. Skocaj and A. Leonardis, "Weighted and robust incremental method for subspace learning," in *Proc. 9th IEEE ICCV*, Oct. 2003, pp. 1494–1501.

[36] A. Torralba and P. Sinha, "Detecting faces in impoverished images," *AI Memo 2001-028, CBCL Memo 208*, 2001, pp. 1–14.
[37] P. A. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Proc. NIPS*, 2005, pp. 1417–1424.
[38] L. Wen, Z. Cai, Z. Lei, D. Yi, and S. Z. Li, "Online spatio-temporal structural context learning for visual tracking," in *Proc. 12th ECCV*, vol. 4. Oct. 2012, pp. 716–729.
[39] L. Wen, Z. Cai, M. Yang, Z. Lei, D. Yi, and S. Z. Li, "Online multiple instance joint model for visual tracking," in *Proc. IEEE 12th Int. Conf. AVSS*, Sep. 2012, pp. 319–324.
[40] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 7, pp. 1195–1209, Jul. 2009.
[41] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, pp. 1–13, 2006.
[42] Q. Yu, T. B. Dinh, and G. G. Medioni, "Online tracking and reacquisition using Co-trained generative and discriminative trackers," in *Proc. 10th ECCV*, Oct. 2008, pp. 678–691.
[43] X. Zhang, W. Hu, S. J. Maybank, X. Li, and M. Zhu, "Sequential particle swarm optimization for visual tracking," in *Proc. IEEE CVPR*, Jun. 2008, pp. 1–8.

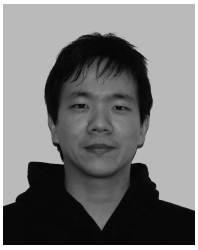


Longyin Wen is currently pursuing the Ph.D. degree with the Institute of Automation, Chinese Academy of Sciences. He received the B.S. degree in automation from the University of Electronic Science and Technology of China in 2010. His research interests are computer vision, pattern recognition, and object tracking in particular.



Zhaowei Cai received the B.S. degree in automation from Dalian Maritime University in 2011. From 2011 to 2013, he was with the Institute of Automation, Chinese Academy of Sciences, as a Research Assistant. He is currently pursuing the Ph.D. degree with the Electrical and Computer Engineering Department, University of California, San Diego. His research interests are computer vision and machine learning.

Zhen Lei received the B.S. degree in automation from the University of Science and Technology of China in 2005 and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA) in 2010. He is currently an Associate Professor with the National Laboratory of Pattern Recognition, CASIA. His research interests are computer vision, pattern recognition, image processing, and face recognition in particular. He has published over 60 papers in international journals and conferences.



Dong Yi is an Assistant Professor with the Institute of Automation, Chinese Academy of Sciences. He received the B.S. degree in electronic engineering in 2003, the M.S. degree in communication and information system in 2006 from Wuhan University, and the Ph.D. degree in pattern recognition and intelligent systems from CASIA. His research areas are unconstrained face recognition, heterogeneous face recognition, and their applications. He has authored and acted as a reviewer for tens of articles in international conferences and journals. He has developed the face biometric modules and systems for the immigration control projects and 2008 Beijing Olympic Games.

Stan Z. Li received the B.Eng. degree from Hunan University, China, the M.Eng. degree from the National University of Defense Technology, China, and the Ph.D. degree from Surrey University, U.K. He is currently a Professor with the National Laboratory of Pattern Recognition and the Director of the Center for Biometrics and Security Research, Institute of Automation, and the Director of the Center for Visual Internet of Things Research, Chinese Academy of Sciences. He was with Microsoft Research Asia as a Researcher from 2000 to 2004. He was an Associate Professor with Nanyang Technological University, Singapore. He was elevated to IEEE Fellow for his contributions to the fields of face recognition, pattern recognition, and computer vision. His research interest includes pattern recognition and machine learning, image and vision processing, face recognition, biometrics, and intelligent video surveillance. He has published over 200 papers in international journals and conferences, and has authored and edited eight books. He was an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and was acting as the Editor-in-Chief for the *Encyclopedia of Biometrics*. He served as a Program Co-Chair for the International Conference on Biometrics 2007 and 2009, the International Joint Conference on Biometrics 2014, the 11th IEEE Conference on Automatic Face and Gesture Recognition, a General Chair for the 9th IEEE Conference on Automatic Face and Gesture Recognition, and has been involved in organizing other international conferences and workshops in the fields of his research interest.