Neurocomputing 74 (2011) 3277–3285

Contents lists available at ScienceDirect

Neurocomputing

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

Visual tracking via dynamic tensor analysis with mean update

Xiaoqin Zhang [a],*, Xingchu Shi [b], Weiming Hu [b], Xi Li [b], Steve Maybank [c]

[a] College of Mathematics & Information Science, Wenzhou University, Zhejiang, China
[b] National Laboratory of Pattern Recognition, CASIA, Beijing, China
[c] Department of Computer Science and Information Systems, Birkbeck College, London, UK

ARTICLE INFO

Article history:
Received 21 November 2010
Received in revised form
9 March 2011
Accepted 8 May 2011
Communicated by Tao Mei
Available online 14 June 2011

Keywords:
Appearance model
Visual tracking
Subspace learning
Incremental updating

ABSTRACT

The appearance model is an important issue in the visual tracking community. Most subspace-based appearance models focus on the time correlation between the image observations of the object, but the spatial layout information of the object is ignored. This paper proposes a robust appearance model for visual tracking which effectively combines the spatial and temporal eigen-spaces of the object in a tensor reconstruction way. In order to capture the variations in object appearance, an incremental updating strategy is developed to both update the eigen-space and mean of the object. Experimental results demonstrate that, compared with the state-of-the-art appearance models in the tracking literature, the proposed appearance model is more robust and effective.

1. Introduction

Visual tracking is an important research topic in the computer vision community, because it is the foundation for high level visual tasks such as motion analysis and behavior understanding. Recent years have witnessed a great advance in the literature, e.g. snakes model [1], condensation [2], mean shift [3], appearance model [4], and the probabilistic data association filter [5].

Generally speaking, most of the tracking algorithms address two major issues: the tracking framework and the appearance model of the object. For visual tracking, handling appearance variations of an object is a fundamental and challenging task. Consequently, effectively modeling such appearance variations plays a critical role in visual tracking.

Image patch [6], which takes the set of pixels in the target region as the model representation, is a direct way to model the target, but it loses the discriminative information that is implicit inside the layout of the target. The color histogram [3,7] provides global statistical information about the target region which is robust to noise, but it has two major problems: (1) the histogram is very sensitive to illumination changes; (2) the relative positions of the pixels in the image are ignored. A consequence of (2) is that trackers based on color histograms are prone to lose track if the object is near to other objects with a similar appearance. In [2], curves or splines are used to represent the apparent boundary of the object, and Condensation algorithm is developed for contour-based tracking. Due to the simplistic representation scheme, which is confined to the apparent boundary, the algorithm is sensitive to image noise, leading to tracking failures in cluttered backgrounds. Stauffer et al. [8] employ a Gaussian mixture model (GMM) to represent and recover the appearance changes in consecutive frames. Jepson et al. [4] develop a more elaborate Gaussian mixture model which consists of three components S,W,L, where S component models temporally stable images, W component models the two-frame variations, and L component models data outliers, for example those caused by occlusion. An online EM algorithm is employed to explicitly model appearance changes during tracking. Later, Zhou et al. [9] replace the component L with a component F, which is a fixed template of the target to prevent the tracker from drifting away from the target. This appearance-based adaptive model is embedded into a particle filter to achieve a robust visual tracking. Wang et al. [10] present an adaptive appearance model based on a mixture of Gaussians model in a joint spatial-color space (referred to as SMOG). SMOG captures rich spatial layout and color information. However, these GMM-based appearance models consider each pixel independently and with the same level of confidence, which is not reasonable in practice.

Recently, the subspace learning-based appearance models have received more and more attention because of the following merits: (1) constant subspace assumption is more reasonable than constant brightness assumption, so it is more robust to model drifting; (2) it is easy to learn the subspace of the object;

* Corresponding author.
E-mail addresses: xqzhang@wzu.edu.cn, xqzhang@nlpr.ia.ac.cn, zhangxiaoqinnan@gmail.com (X. Zhang), xcshi@nlpr.ia.ac.cn (X. Shi), wmhu@nlpr.ia.ac.cn (W. Hu), lixi@nlpr.ia.ac.cn (X. Li), sjmaybank@dcs.bbk.ac.uk (S. Maybank).

0925-2312/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2011.05.006

(3) it possess low computation and storage resources. For example, in [11], a view-based eigenbasis representation of the object is learned off-line, and applied to form a tracking algorithm which matches successive views of the object. However, it is very difficult to collect training samples that cover all possible viewing conditions. Therefore, this algorithm is only feasible under those conditions for which training data has been obtained. Later, some researchers try to update the object subspace in the tracking process to capture the changes of the appearance. The pioneering work on applying the incremental subspace learning to tracking is by Lim et al. [12], where they extend the SKL (sequential Karhunen–Loeve) [13] algorithm to effectively learn the variations of both appearance and illumination in an incremental way. However, their work only focuses on the matching between the object subspace and candidates. The information for classification in the background is discarded. In [14], a two-class FDA (Fisher discriminant analysis) based model is proposed to learn the discriminative subspace to separate the object from the background. It has a more discriminative ability than PCA (principal component analysis) models, since it utilizes the background appearance as negative training data. Zhang et al. [15] propose a graph embedding-based learning algorithm for object tracking, which can simultaneously learn the subspace of the target and its local discriminative structure against the background. Despite the success of the above algorithms in the tracking literature, they still have the following limitation: all the above subspace-based tracking algorithms use a flattened vector to represent a target, so the local spatial information contained in the relative positions of the pixels which form the target is almost lost, making the appearance model not discriminative enough for tracking against cluttered backgrounds. To address this problem, Li et al. [16,17] propose a visual tracking framework based on online tensor decomposition. The framework relies on image-as-matrix techniques for considering the spatial layout information, and adopts the R-SVD (Singular Value Decomposition) technique [19] to incrementally calculate the sample mean and subspace of each tensor mode. However, their calculation of the eigenstructure is not accurate, because, in the tracking process, the eigenvectors with small eigenvalues are discarded in order to fulfill the real-time requirement. The tracking errors accumulate, causing the subspace model to drift away from the target.

Based on the forgoing discussions, we propose a dynamic tensor analysis-based tracking algorithm, which effectively combines the spatial and temporal eigen-space of the object.

The main features of our tracking approach are summarized as follows:

- We propose a dynamic tensor analysis-based tracking algorithm, which effectively captures the spatial and temporal eigen-space of the object.
- We propose an effective strategy for incrementally updating the spatial and temporal eigen-space of the object.
- We conduct a theoretical comparison with the subspace models in [12,16], and show that the proposed incremental updating strategy for the eigen-space of the object is more accurate.

The arrangement of this paper is as follows. Section 2 gives an overview of the tracking algorithm. The details of the proposed appearance model and the incremental updating strategy are introduced in Section 3. The particle filtering-based tracking framework is given in Section 4. Experimental results are presented in Section 5, and Section 6 is devoted to conclusion.

2. Overview of the tracking algorithm

The proposed tracking framework includes three stages: (a) tensor analysis-based appearance model; (b) particle filtering-based tracking framework; (c) dynamic tensor update. In (a), an object region is represented as a tensor that consists of the object's observation matrices obtained from the frames preceding the current frame. Each matrix in the tensor represents an image observation of the object in the tracking process. In (b), the object state in the current frame is obtained by maximum a posteriori (MAP) estimation within the particle filtering framework. (c) During the tracking process, the tensor subspace is needed to update incrementally to accommodate the appearance variations. The aforementioned three stages are executed iteratively as time progresses. The architecture of the framework is shown in Fig. 1.

3. The proposed appearance model

In this section, we first introduce the basic theory of dynamic tensor analysis, and then present the proposed appearance model. Finally, a comparison with two other subspace-based appearance models [12,16] is carried out to show the theoretical advantages of the proposed appearance model.

3.1. Basic theory of dynamic tensor analysis

3.1.1. Tensor decomposition

A tensor can be regarded as a multidimensional generalization of a matrix. We denote an N-order tensor as $\mathcal{A} \in \mathcal{R}^{I_1 \times I_2 \times \cdots \times I_N}$, each element of which is represented as $a_{i_1 \cdots i_n \cdots i_N}$ for $1 \leq i_n \leq I_n$. In the tensor terminology, each dimension of a tensor is associated with a 'mode'. The mode-n unfolding matrix $A_{(n)} \in \mathcal{R}^{I_n \times (\prod_{i \neq n} I_i)}$ of $\mathcal{A}$ consists of the I_n-dimensional mode-n vectors obtained by varying the nth-mode index i_n while keeping the other mode indices

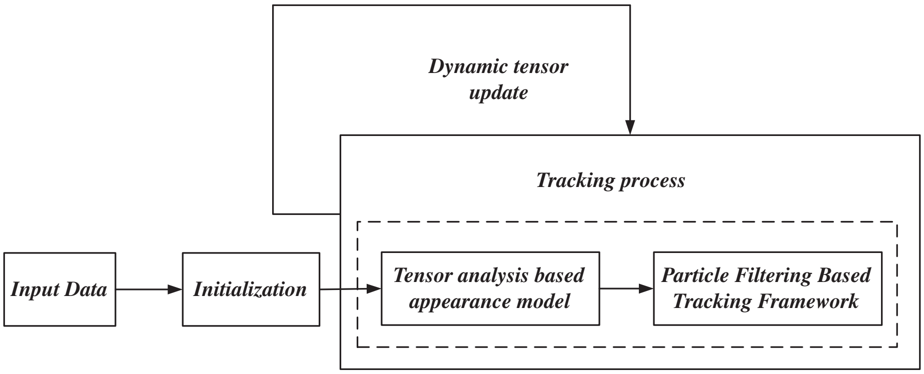

Dynamic tensor update
Tracking process
Input Data
Initialization
Tensor analysis based appearance model
Particle Filtering Based Tracking Framework

Fig. 1. The architecture of the tracking framework.

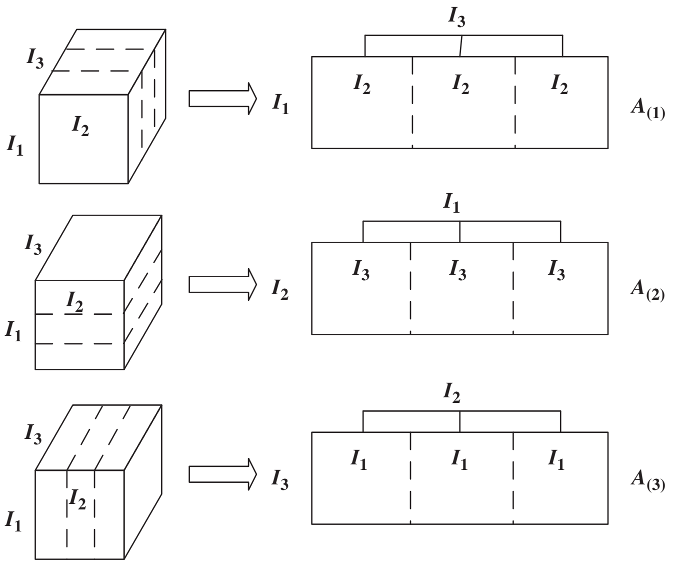


Fig. 2. Illustration of unfolding a 3-order tensor.

fixed. Namely, the column vectors of $A_{(n)}$ are just the mode-n vectors. For a better understanding of the tensor unfolding, we show an example in Fig. 2 of the unfolding of a 3-order tensor.

Based on the mode-n unfolding operation, the tensor $\mathcal{A}$ can be decomposed as follows:

$$\mathcal{A} = \mathcal{B} \times_1 U^{(1)} \times_2 U^{(2)} \cdots \times_N U^{(N)} \tag{1}$$

where $\mathcal{B} = \mathcal{A} \times_1 U^{(1)^T} \times_2 U^{(2)^T} \cdots \times_N U^{(N)^T}$ which denotes the core tensor controlling the interaction among the mode matrices $U^{(1)}, \cdots, U^{(N)}$. The orthonormal column vectors of $U^{(n)}$ span the column space of the mode-n unfolding matrix $A_{(n)} (1 \le n \le N)$.

3.1.2. Dynamic tensor analysis

In some applications, such as visual tracking, the data is typically a time sequence, and hence the tensor data changes over the time, so the off-line tensor decomposition method in Section 3.1.1 is not suitable in this case.

Dynamic tensor analysis (DTA) [18], is an incremental algorithm for tensor decomposition. The basic idea in DTA is to update the covariance matrix along each unfolding mode. In the algorithm in [18], the samples are assumed to have zero mean, so in a incrementally learning process, the covariance matrix of the dth mode is updated as

$$C_d = C_d + X_{(d)} X_{(d)}^T \tag{2}$$

where $X_{(d)}$ is mode-d unfolding matrix of new incoming tensor. Then the mean and eigenstructure of the object region can be easily updated by solving the eigenproblem for the updated C_d.

In order to make the dynamic model depend more heavily on the most recent tensor data, we assume that the previous data are gradually forgotten and new data are gradually added to the dynamic model, so the Eq. (2) is modified as

$$C_d = \lambda C_d + X_{(d)} X_{(d)}^T \tag{3}$$

where $\lambda \in [0, 1]$ is a forgetting factor, and it is used to weight the historical data. If $\lambda = 1$, then the historical data and the new data contribute equally to the construction of the tensor subspace. If $\lambda = 0$, then the historical data are discarded and the tensor subspace only depends on the new data.

3.2. Dynamic tensor analysis-based appearance model

Based on the theory of dynamic tensor analysis, we proposed a dynamic tensor analysis-based appearance model, which consists of the image likelihood for observation evaluation and the incremental updating process of the object subspaces.

3.2.1. Image likelihood based the reconstruction error

Let $\mathcal{O} \in \mathcal{R}^{I_1 \times I_2 \times I_3}$ be a set of image observations of the object, where I_1, I_2 and I_3 represent the width, height and frame index of the image observations. According to the multilinear algebra and tensor theory, the tensor $\mathcal{O}$ can be unfolded in three modes, which is shown in Fig. 2.

Let us focus on the unfolding matrixes $A_{(1)}, A_{(2)}, A_{(3)}$. We can see that the column spaces of $A_{(1)}$ and $A_{(2)}$ capture the spatial layout information of the object, and the row space of $A_{(3)}$ captures the temporal information of the object during tracking process.

In order to obtain the column spaces of $A_{(1)}, A_{(2)}, A'_{(3)}$, the covariance matrixes of them are calculated as follows:

$$C_1 = \sum_i (A_{(1)}^i - \mu_1)(A_{(1)}^i - \mu_1)^T \tag{4}$$

$$C_2 = \sum_j (A_{(2)}^j - \mu_2)(A_{(2)}^j - \mu_2)^T \tag{5}$$

$$C_3 = \sum_k (A\prime_{(3)}^k - \mu_3)(A\prime_{(3)}^k - \mu_3)^T \tag{6}$$

where μ_1, μ_2 are the column mean of unfolding sample matrixes $A_{(1)}, A_{(2)}$, respectively, and μ_3 is the row mean of unfolding sample matrix $A_{(3)}$. Based on the covariance matrixes, the eigen-spaces can be easily obtained by diagonalization: $C_d = U_d S_d U_d^T, d = 1,2,3$.

After the diagonalization process, given a test image candidate $o_t \in R^{I_1 \times I_2}$ and its flattened vector form $v_t \in R^{(I_1 I_2) \times 1}$, the sum of the reconstruction squared error norms on the three modes is calculated as follows:

$$RE_1 = \| (o_t - \mathbb{\mu}_1) - (o_t - \mathbb{\mu}_1) U_1 U_1^T \|^2 \tag{7}$$

$$RE_2 = \| (o'_t - \mathbb{\mu}_2) - (o'_t - \mathbb{\mu}_2) U_2 U_2^T \|^2 \tag{8}$$

$$RE_3 = \| (v_t - \mu_3) - (v_t - \mu_3) U_3 U_3^T \|^2 \tag{9}$$

$$RE = RE_1 + RE_2 + RE_3 \tag{10}$$

where $\mathbb{\mu}_1$ and $\mathbb{\mu}_2$ are defined as follows:

$$\mathbb{\mu}_1 = \left(\overbrace{\mu_1, \ldots, \mu_1}^{I_2} \right) \in R^{I_1 \times I_2}$$

$$\mathbb{\mu}_2 = \left(\overbrace{\mu_2, \ldots, \mu_2}^{I_1} \right) \in R^{I_2 \times I_1}$$

Finally, the image likelihood of the image candidate o_t can be formulated as follows:

$$p(o_t | x_t) \propto \exp(-RE) \tag{11}$$

In this way, the spatial layout information and the temporal information of the object can be effectively combined through the image observation reconstruction process. As a result, the proposed appearance model is robust to image noise, cluttered backgrounds and partial occlusion.

3.3. Dynamic tensor analysis with mean update

In most tracking applications, the tracker must simultaneously deal with the changes in both the target appearance and the illumination. As a result, it is necessary to update the subspaces

$U_d, d=1,2,3$ and the means $\mu_d, d=1,2,3$ incrementally to accommodate these changes. In practice, the zero mean assumption in Eq. (3) for the tensor data is not reasonable. So, the key point of updating the subspaces is how to update both the covariance matrix and the mean.

Denote the current matrix data in mode-d as $A_{(d)} = [I_1, I_2, \ldots, I_m]$, where I_k is the kth sample, and denote the corresponding sample mean as μ_d. The covariance matrix of the sample matrix $A_{(d)}$ is calculated as $C_d = \sum_{k=1}^{m}(I_k-\mu_d)(I_k-\mu_d)^T$. Denote the incoming data matrix in mode-d as $X_{(d)} = [I_{m+1}, I_{m+2}, \ldots, I_{m+n}]$, and the corresponding sample mean as μ'_d. Denote the total data matrix in mode-d as $\tilde{A}_{(d)} = [I_1, I_2, \ldots, I_{(m+n)}]$, with its covariance matrix $\tilde{C}_d$ and sample mean μ''_d.

According the definition of the covariance matrix, $\tilde{C}_d$ can be calculated as

$$\begin{aligned}\tilde{C}_d &= \sum_{k=1}^{m+n}(I_k-\mu''_d)(I_k-\mu''_d)^T \\ &= \sum_{k=1}^{m}(I_k-\mu''_d)(I_k-\mu''_d)^T + \sum_{k=m+1}^{m+n}(I_k-\mu''_d)(I_k-\mu''_d)^T \\ &= \sum_{k=1}^{m}(I_k-\mu_d+\mu_d-\mu''_d)(I_k-\mu_d+\mu_d-\mu''_d)^T \\ &\quad + \sum_{k=m+1}^{m+n}(I_k-\mu'_d+\mu'_d-\mu''_d)(I_k-\mu'_d+\mu'_d-\mu''_d)^T\end{aligned}$$

Based on $\mu''_d = (m/(m+n))\mu_d + (n/(m+n))\mu'_d$, the above formulation can be simplified as

$$\begin{aligned}\tilde{C}_d &= \sum_{k=1}^{m}(I_k-\mu_d)(I_k-\mu_d)^T + \sum_{k=m+1}^{m+n}(I_k-\mu'_d)(I_k-\mu'_d)^T \\ &\quad + \frac{mn}{m+n}(\mu_d-\mu'_d)(\mu_d-\mu'_d)^T \\ &= C_d + (X_{(d)}-\boldsymbol{\mu}'_d)(X_{(d)}-\boldsymbol{\mu}'_d)^T + \frac{mn}{m+n}(\mu_d-\mu'_d)(\mu_d-\mu'_d)^T\end{aligned}$$

where $\boldsymbol{\mu}'_d$ is defined as follows:

$$\boldsymbol{\mu}'_d = \left(\overbrace{\mu'_d, \ldots, \mu'_d}^{n}\right)$$

Similar to Eq. (3), we include the forgetting factor λ to make the covariance matrix more concentrated on the new coming data,

$$C_d \leftarrow \lambda C_d + (X_{(d)}-\boldsymbol{\mu}'_d)(X_{(d)}^T-\boldsymbol{\mu}'_d)^T + \frac{mn}{m+n}(\mu_d-\mu'_d)(\mu_d-\mu'_d)^T \qquad (12)$$

Here, C_d is exponentially forgotten for $\lambda = 1-e^{-1/\tau}$, where τ is a predefined constant.

In summary, the reason why the dynamic tensor analysis-based appearance model with mean update is effective for visual tracking is two-fold: (1) the dynamic analysis of tensor subspace effectively captures the changes of object appearance along the spatial and temporal axes; (2) the calculation of tensor subspace and reconstruction is accurate only when the sample mean is updated.

3.4. Comparison with two other subspace-based appearance models

In this part, we compare the proposed appearance model with other two subspace-based appearance models [12,16].

For the appearance model in [12], we call it ISL (incremental subspace learning) for short. In the implementation of ISL, the image observation in the object region is firstly flatted into a vector and then the R-SVD technique [19] is applied to the vector data to incrementally learn the subspace of the object. ISL only captures the temporal information of the object in the tracking process, while almost loses the spatial information of the object. In contrast, as shown in Fig. 3, in our appearance model, the first two unfolding modes correspond to the vertical and horizontal spatial layout of the object, respectively, and the third unfolding mode corresponds to the temporal evolution of the object. By combining the three modes together, the proposed appearance model is more discriminative than the ISL. Furthermore, let us focus on the bottom row of Fig. 3, if only the third unfolding mode is used, the proposed model degenerates to ISL. As a result, the proposed appearance model is a unified framework for subspace learning and the ISL is a special case of our framework.

For the appearance model in [16], we call it IRSTA (Incremental Rank-R Tensor Subspace Analysis) for short. Although the tensor unfolding process in IRSTA is the same as in our proposed appearance model, the methods for calculating the eigenstructure of the unfolding matrix are different. In the implementation of IRSTA, R-SVD is adopted to incrementally learn the sample mean and subspace of each tensor mode. The method in [16] is efficient, however, the solution of the eigenstructure is not accurate, because in the tracking process, only the first R eigenvectors are retained, and the eigenvectors with small eigenvalues are discarded in order to fulfill the real-time requirement. As a result, the tracking errors accumulate, causing the subspace model to drift away from the target. While in the proposed appearance model, the updating process of the new coming tensor data is performed at the level of the covariance matrix. As a result, all the

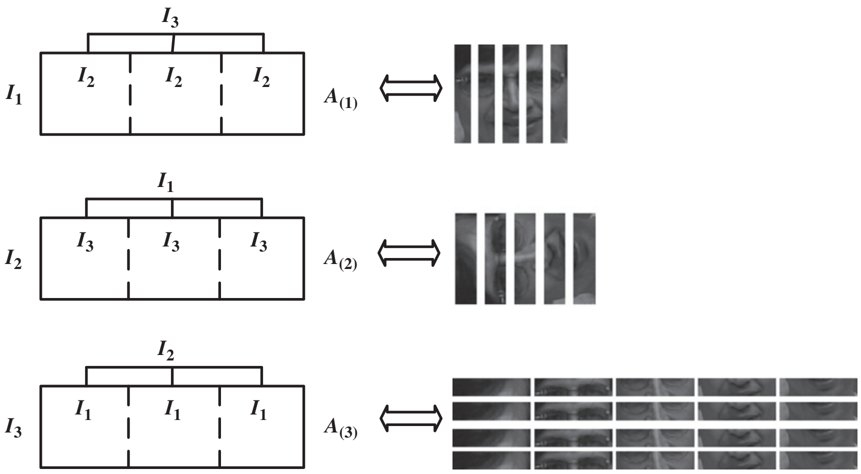


Fig. 3. The corresponding relationship between the unfolding matrices and images.

spatial and temporal information of the object in the tracking process is retained. While maintaining a similar computational complexity to IRSTA, our proposed appearance model is more robust to the model drifting problem.

4. Particle filtering-based tracking framework

Particle filtering provides a flexible and effective tracking framework. Therefore, we embed the above appearance model into a particle filtering framework to form a robust tracking algorithm.

Our algorithm localizes the tracked object in each image frame using a rectangular window. The motion of a tracked object between two consecutive frames is approximated by an affine image warping. Specifically, the motion is characterized by the state of the particle $x_t = (t_x, t_y, \theta, s, \alpha, \beta)$ where $\{t_x, t_y\}$ is the 2-D translation parameters and $\{\theta, s, \alpha, \beta\}$ are deformation parameters. We employ a Gaussian distribution for the state transition distribution $p(x_t|x_{t-1})$,

$$p(x_t|x_{t-1}) = \mathcal{N}(x_t; x_{t-1}, \Sigma) \tag{13}$$

where Σ is a diagonal covariance matrix whose elements are the corresponding variances of affine parameters, i.e., $\sigma_x^2, \sigma_y^2, \sigma_\theta^2, \sigma_s^2, \sigma_\alpha^2, \sigma_\phi^2$.

The observation model $p(o_t|x_t)$ reflects the probability that a sample is generated from the subspace, and it is defined in Eq. (11). Finally, the tracking result is the maximum a posteriori (MAP) estimation given the candidate samples.

5. Experimental results

In order to show the effectiveness of the proposed appearance model (here, called DTAMU: dynamic tensor analysis with mean updating), we first conduct an experimental comparison between DTAMU and DTA (dynamic tensor analysis without mean updating) to investigate the importance of mean updating. Then we

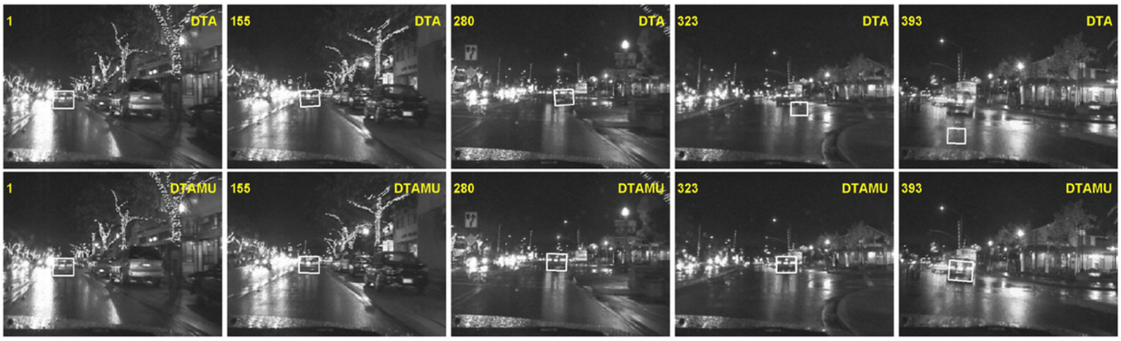


Fig. 4. Tracking results of ‘car’ sequence (top row: DTA, bottom row: DTAMU).

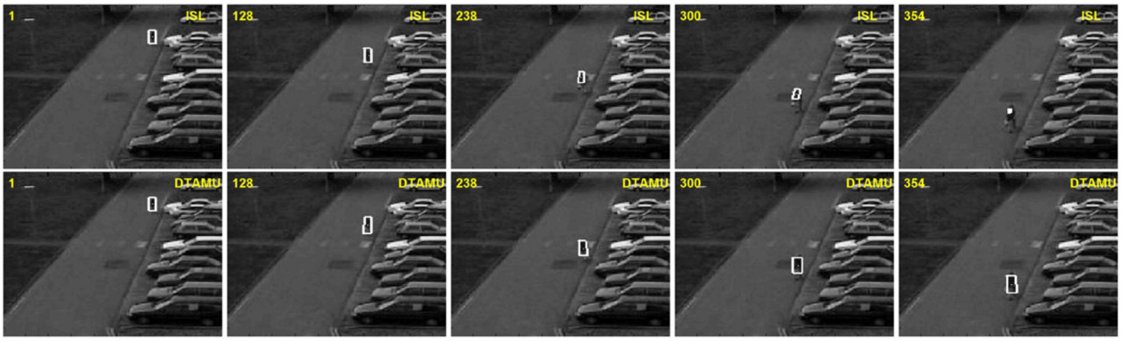


Fig. 5. Tracking results of ‘pedestrian’ sequence (top row: ISL, bottom row: DTAMU).

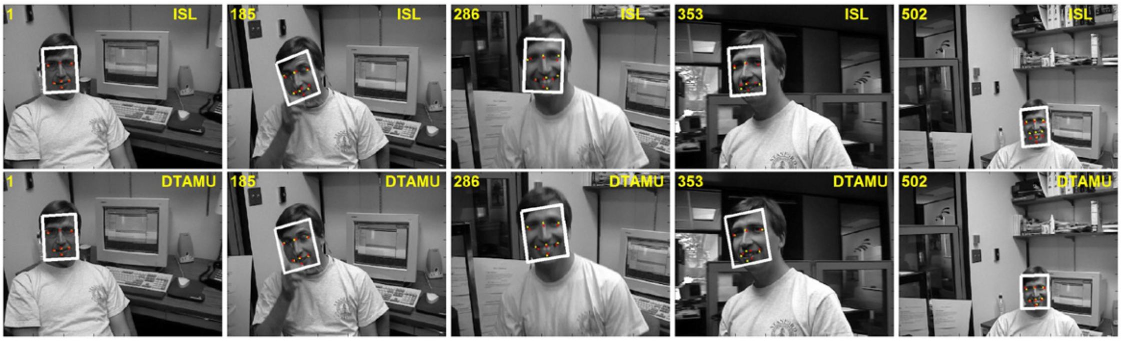


Fig. 6. Tracking results of ‘Dudek’ sequence (top row: ISL, bottom row: DTAMU).

carry out a number of comparison experiments with several state-of-the-art appearance models including: (1) DTAMU vs. ISL; (2) DTAMU vs. IRSTA; (3) DTAMU vs. other state-of-the-art appearance models. All the experiments are conducted with Matlab on a platform with Pentium IV 2.8 GHz CPU and 512 M memory, and the initial object positions are manually labeled.

5.1. DTAMU vs. DTA

To show the importance of the mean updating process in Eq. (12), we conduct an experimental comparison between two subspace updating strategies during tracking: mean updating (DTAMU) and no updating (DTA).

The parameters in this experiment are set to $\{N=300,\Sigma=diag(5^2,5^2,0.01^2,0.01^2,0.001^2,0.001^2)\}$ which are, respectively, the number of particles and the covariance matrix of the transition distribution, and the forgetting factor λ is set to 0.99. As shown in the top row of Fig. 4, we can see that with no updating for the mean of the subspace model, the tracking window gradually deviates from the car when the car turns a corner, and the track is lost completely in the subsequent frames. The reason is that when the car turns a corner, its image intensities undergo large

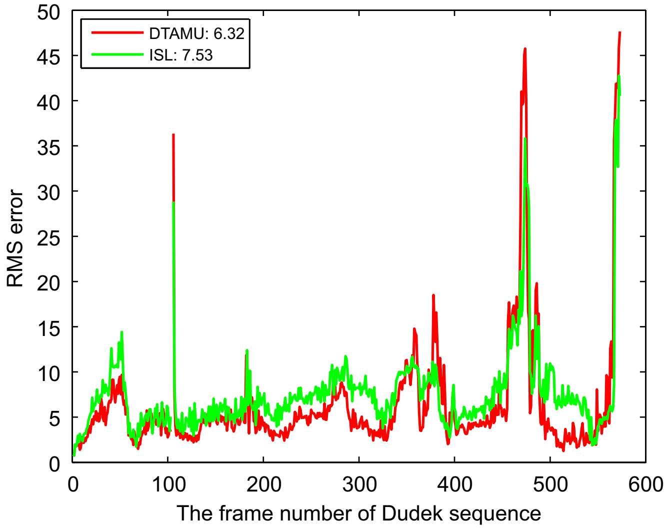


Fig. 7. The RMS error curve of tracking results (red: DTAMU, green: ISL). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

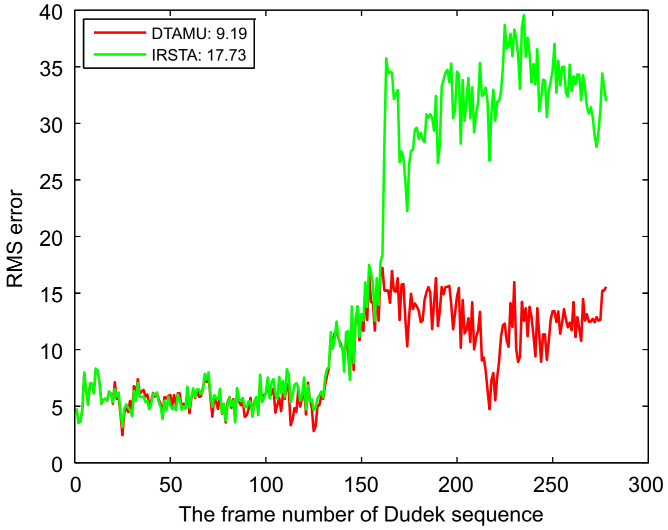


Fig. 10. The RMS error curve of tracking results (red: DTAMU, green: ISL). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

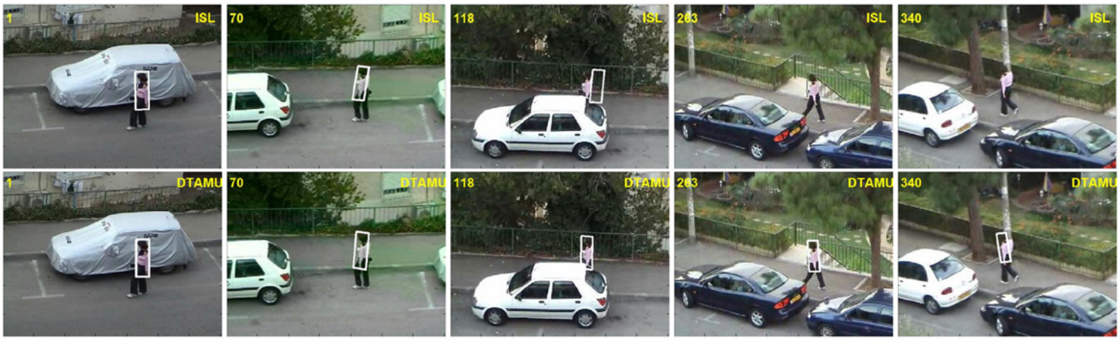


Fig. 8. Tracking results of 'woman' sequence (top row: IRSTA, bottom row: DTAMU).

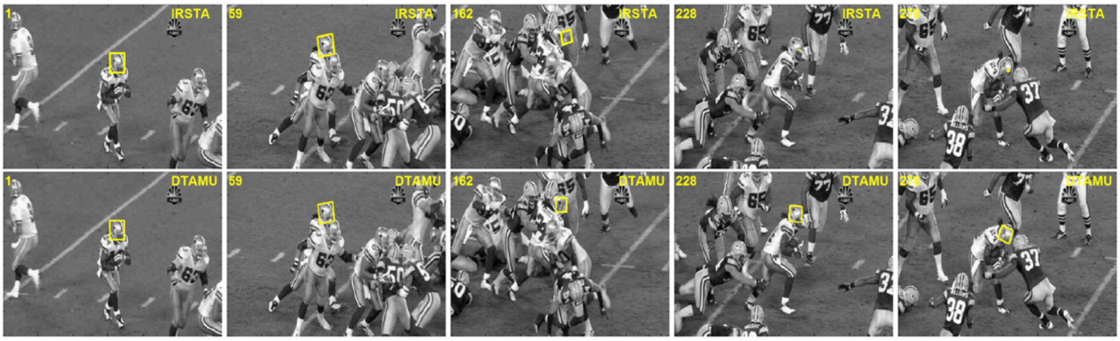


Fig. 9. Tracking results of 'football' sequence (top row: IRSTA, bottom row: DTAMU).

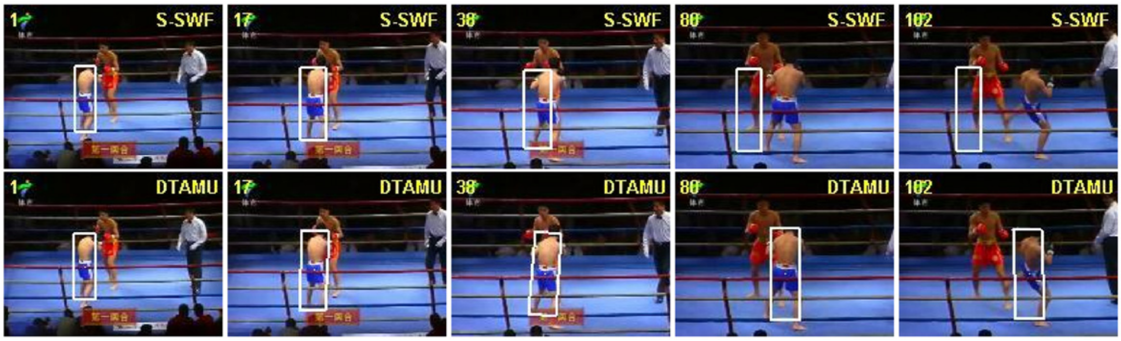


Fig. 11. Tracking results of 'boxing' sequence (top row: S-SWF, bottom row: DTAMU).

changes, so the correspondence of pixels between the car and the subspace is not accurate without mean updating. In contrast, DTAMU can successfully track the car throughout the whole sequence, because the mean and eigenstructure of the car are correctly calculated and updated in the tracking process.

5.2. DTAMU vs. ISL

In this part, an experimental comparison is made between DTAMU and ISL.

The first testing sequence is from the PETS 2001 database which is an open database for research on visual surveillance.[1] This sequence contains a pedestrian of small apparent size, moving down a road in a dark and blurry scene. To make a fair comparison, the parameters adopted in these two algorithms are set to $\{N=500, \Sigma=diag(5^2,5^2,0.03^2,0.03^2, 0.005^2,0.001^2), \lambda=0.99\}$. Fig. 5 illustrates some key frames showing the tracking results for this sequence, in which the top row and the bottom row represent the tracking results of ISL and DTAMU, respectively. From Fig. 5, we can see that the tracking window of ISL drifts from the correct position at frame 238, and in addition its size shrinks gradually, leading to a total loss of track with no recovery. In contrast, DTAMU tracks the object right through the sequence and maintains a suitable window size, which covers the object region. The tracking is successful in spite of the poor lighting. The underlying reason for the above tracking results is as follows: in ISL, most of the spatial layout information in the object region is discarded, and because the apparent size of the pedestrian is small, there is not enough information in the temporal subspace to support the tracker. DTAMU extracts the spatial layout of the object and combines with the temporal subspace, and this additional information makes the tracker more accurate.

To further illustrate the importance of the object's spatial layout information for the localization accuracy, we test these two appearance models on the labeled Dudek sequence.[2] As shown in the Fig. 6, the tracking results are represented by seven key facial points, and the root mean square (RMS) error between the estimated points and the groundtruth is used to evaluate the tracking performance.

For the Dudek sequence, the parameters are set to $\{N=300, \Sigma=diag(11^2,11^2,0.05^2,0.05^2,0.005^2,0.001^2),\lambda=0.98\}$. As illustrated in Fig. 6, the tracking results of DTAMU are more accurate than ISL for most image frames. The RMS error of DTAMU for the whole sequence is 6.32 while the corresponding RMS error of ISL is 7.53. Fig. 7 shows several key frames from the Dudek sequence. We can see that the facial points estimated by DTAMU are consistent with groundtruth, and can be used for facial expression analysis.

5.3. DTAMU vs. IRSTA

In this part, we conduct a quantitative comparison between DTAMU and IRSTA [16]. There are two differences between DTAMU and IRSTA: (1) the calculation of object subspace; (2) the incremental updating process.

For the two testing image sequences, the parameters employed in DTAMU and IRSTA are set as follows: (1) N=400, $\Sigma=diag(5^2,5^2,0.03^2,0.02^2,0.005^2,0.001^2)$, $\lambda=0.97$; (2) N=300, $\Sigma=diag(5^2,5^2,0.03^2,0.02^2,0.003^2,0.001^2)$, $\lambda=0.98$. In addition, the R employed in IRSTA is set to 10. For the first image sequence, the tracking results is shown in Fig. 8, from which we can see that IRSTA fails to track the woman when she is partially occluded by the car, while DTAMU achieves better performance than IRSTA. For the second image sequence, some key frames of the tracking results are shown in Fig. 9. The tracking window of IRSTA drifts off the target at frame 162, while DTAMU can successfully tracks the target until frame 286 when the target is severely occluded. The groundtruth of the tracking window in this sequence is manually labeled for the quantitatively evaluation. As illustrated in Fig. 10, the tracking results of DTAMU is more accurate than IRSTA. The RMS error of DTAMU for the whole sequence is 9.19 while the corresponding RMS error of IRSTA is 17.73. The reason is that only the first R eigenvectors are retained in the incremental updating process, so in each tracking process, the subspace is not accurate enough, causing the model drift problem.

5.4. DTAMU vs. two other appearance models

In this part, we compare DTAMU with two state-of-the-art appearance models [20,10]. Before illustrating the tracking results, we first state the reason of choosing these two appearance models for the comparison. Both of these two appearance models combine the spatial-temporal information of the object. In this way they are similar to DTAMU. The appearance model in [20] is an extension of SWF model [9], which incorporates the spatial constraint to the SWF model, so it is called S-SWF for short in this work. We use the term 'SMOG' (taken from [10]) for the model in [10].

The first testing sequence for DTAMU and S-SWF is a boxing match (see Fig. 11), which contains two boxers and the referee. There are large changes in the appearances of the participants. From experimental results shown in Fig. 11, we can see that S-SWF fails at frame 38 and can not recover again. In contrast, DTAMU can effectively capture the dynamic motion and adapt to the appearance changes. The test sequence for DTAMU and SMOG is shown in Fig. 12, where a man moves in an outdoor environment with large changes in

[1] The source data was obtained from http://homepages.inf.ed.ac.uk/rbf/CAVIAR/.

[2] The source data was obtained from http://www.cs.toronto.edu/vis/projects/dudekfaceSequence.html

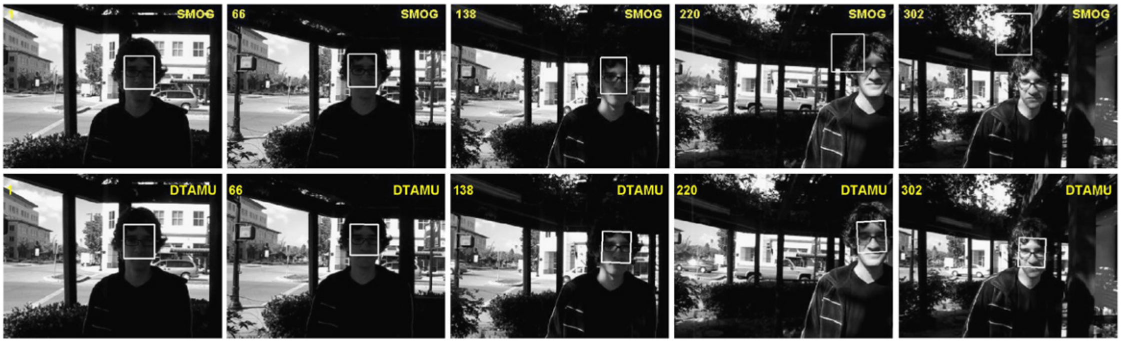


Fig. 12. Tracking results of 'Trellis' sequence (top row: SMOG, bottom row: DTAMU).

illumination and in his appearance. As shown in Fig. 12, SMOG loses track when the man undergoes large illumination changes. However, DTAMU successfully tracks the man through the entire sequence even with large illumination and appearance changes. The reason for these results is that the spatial and temporal subspaces capture more information than simply modeling the object region as a mixture of Gaussians. Thus the proposed appearance model is more robust and discriminative, especially in changing environments.

6. Conclusion

In this paper, we have proposed a dynamic tensor-based appearance model, which effectively combines the spatial and temporal eigen-space of the object using methods from tensor analysis. In order to adapt the changes of object appearance, the eigen-space and mean of the object are incrementally updated on the covariance matrix level, which never loses any correlation information of the object region in both spatial and temporal axes. Several comparison experiment results demonstrate the effectiveness and robustness of the proposed appearance model.

Acknowledgment

This work is supported by NSFC (Grant nos. 60825204, 60935002, 61074123), open project of State Key Laboratory of Industrial Control Technology (ICT1120), and Zhejiang Provincial Top Key Discipline of Information Processing and Automation Technology.

References

[1] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, International Journal of Computer Vision 1 (4) (1988) 321–332.

[2] M. Isard, A. Blake, Condensation: conditional density propagation for visual tracking, International Journal of Computer Vision 29 (1) (1998) 5–28.

[3] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (5) (2003) 234–240.

[4] A.D. Jepson, D.J. Fleet, T.F. El-Maraghi, Robust online appearance models for visual tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (10) (2003) 1296–1311.

[5] C. Rasmussen, G.D. Hager, Probabilistic data association methods for tracking complex visual objects, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 560–576.

[6] G.D. Hager, P.N. Hager, Efficient region tracking with parametric models of geometry and illumination, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (10) (1998) 1025–1039.

[7] K. Nummiaro, E. Meier, L. Gool, An adaptive color-based particle filter, Image and Vision Computing 21 (1) (2003) 99–110.

[8] C. Stauffer, W. Grimson, Adaptive background mixture models for real-time tracking, in: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999.

[9] S. Zhou, R. Chellappa, B. Moghaddam, Visual tracking and recognition using appearance-adaptive models in particle filters, IEEE Transactions on Image Processing 13 (11) (2004) 1491–1506.

[10] H. Wang, D. Suter, K. Schindler, C. Shen, Adaptive object tracking based on an effective appearance filter, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (9) (2007) 1661–1667.

[11] M.J. Black, A.D. Jepson, EigenTracking: robust matching and tracking of articulated objects using a view-based representation, International Journal of Computer Vision 26 (1) (2004) 63–84.

[12] J. Lim, D. Ross, R.S. Lin, M.H. Yang, Incremental learning for visual tracking, in: Advances in Neural Information Processing Systems, 2004, pp. 793–800.

[13] A. Levy, M. Lindenbaum, Sequential Karhunen–Loeve basis extraction and its application to images, IEEE Transactions on Image Processing 9 (8) (2000) 1371–1374.

[14] R. Lin, D. Ross, J. Lim, M. Yang, Adaptive discriminative generative model and its applications, in: Advances in Neural Information Processing Systems, 2004, pp. 801–808.

[15] X. Zhang, W. Hu, S. Maybank, X. Li, Graph based discriminative learning for robust and efficient object tracking, in: Proceedings of International Conference on Computer Vision, 2007, pp. 1–8.

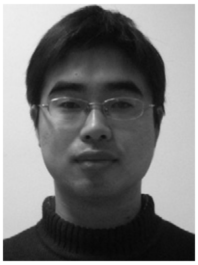
[16] X. Li, W. Hu, Z. Zhang, X. Zhang, G. Luo, Robust visual tracking based on incremental tensor subspace learning, in: Proceedings of International Conference on Computer Vision, 2007.

[17] W. Hu, X. Li, X. Zhang, X. Shi, S. Maybank, Incremental tensor subspace learning and its applications to foreground segmentation and tracking, International Journal of Computer Vision 91 (3) (2011) 303–327.

[18] J. Sun, D. Tao, C. Faloutsos, Beyond streams and graphs: dynamic tensor analysis, in: Proceedings of ACM Conference on Knowledge Discovery and Data Mining, 2006, pp. 374–383.

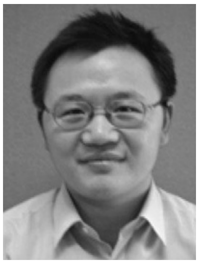
[19] G. Golub, C. Van Loan, Matrix Computations, The Johns Hopkins University Press, 1996.

[20] X. Zhang, W. Hu, S. Maybank, X. Li, M. Zhu, Sequential particle swarm optimization for visual tracking, in: Proceedings of International Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.



Xaoqin Zhang received the B.Sc degree in electronic information science and technology from Central South University, China, in 2005 and Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2010. He is currently a lecture in Wenzhou University, China. His research interests are in visual tracking, motion analysis, and action recognition. He has published more than 30 papers in international and national journals, and international conferences.

Xinchu Shi received the B.Sc degree in Electronics and Information Engineering from Huazhong University of Science and Technology, Wuhan, China, in 2008. Currently, he is a Ph.D. candidate at National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Beijing, China. His research interests include computer vision and pattern recognition.



Weiming Hu received the Ph.D. degree from the Department of Computer Science and Engineering, Zhejiang University. From April 1998 to March 2000, he was a Postdoctoral Research Fellow with the Institute of Computer Science and Technology, Founder Research and Design Center, Peking University. Since April 1998, he has been with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Now he is a Professor and a Ph.D. Student Supervisor in the laboratory. His research interests are in visual motion analysis and recognition of Internet harmful multimedia. He has published more than 100 papers on national and international journals, and international conferences. Now he is an IEEE Senior Member.

Xi Li received the B.Sc degree in communication engineering from Beihang University, Beijing, China, in 2004. In 2009, he got his doctoral degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. He is now a postdoctoral researcher in CNRS, Telecom ParisTech, Paris, France. His research interests include computer vision, pattern recognition, and machine learning.

Steve Maybank received the B.A degree in mathematics from King's College, Cambridge, in 1976 and the Ph.D. degree in computer science from Birkbeck College, University of London, in 1988. He joined the Pattern Recognition Group at Marconi Command and Control Systems, Frimley, in 1980 and moved to the GEC Hirst Research Centre, Wembley, in 1989. From 1993–1995, he was a Royal Society/Engineering and Physical Sciences Research Council (EPSRC) Industrial Fellow in the Department of Engineering Science at the University of Oxford. In 1995, he joined the University of Reading as a lecturer in the Department of Computer Science. In 2004, he became a professor in the School of Computer Science and Information Systems, Birkbeck College. His research interests include the geometry of multiple images, camera calibration, visual surveillance, information geometry, and the applications of statistics to computer vision. He is a senior member of the IEEE.