Expert Systems With Applications 90 (2017) 427–438

Contents lists available at ScienceDirect

Expert Systems With Applications

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

An online spatio-temporal tensor learning model for visual tracking and its applications to facial expression recognition

Sheheryar Khan[a,*], Guoxia Xu[a], Raymond Chan[b], Hong Yan[a]

[a] Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong
[b] Department of Mathematics, Chinese University of Hong Kong, Shatin, Hong Kong

ARTICLE INFO

Article history:
Received 12 May 2017
Revised 3 August 2017
Accepted 21 August 2017
Available online 23 August 2017

Keywords:
Object tracking
Appearance model
Incremental N-mode SVD
Facial expression recognition

ABSTRACT

Robust visual tracking remains a technical challenge in real-world applications, as an object may involve many appearance variations. In existing tracking frameworks, objects in an image are often represented as vector observations, which discounts the 2-D intrinsic structure of the image. By considering an image in its actual form as a matrix, we construct the 3rd order tensor based object representation to preserve the spatial correlation within the 2-D image and fully exploit the useful temporal information. We perform incremental update of the object template using the N-mode SVD to model the appearance variations, which reduces the influence of template drifting and object occlusions. The proposed scheme efficiently learns a low-dimensional tensor representation through adaptively updating the eigenbasis of the tensor. Tensor based Bayesian inference in the particle filter framework is then utilized to realize tracking. We present the validation of the proposed tracking system by conducting the real-time facial expression recognition with video data and a live camera. Experiment evaluation on challenging benchmark image sequences undergoing appearance variations demonstrates the significance and effectiveness of the proposed algorithm.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

Visual tracking in image sequences is amongst the dominant bottom-up units in computer vision applications such as surveillance, robotics, intelligent transportation, and human computer interaction (HCI). A critical requirement of these applications is to track the desired target region of interest for a long period in unconstrained environments. For instance, in face based-HCI (facial expression or identity recognition), the need for accurate face tracking along with head orientations has been widely acknowledged (Jo, Lee, Park, Kim, & Kim, 2014). Despite much progress in visual tracking and endeavours to improve face based-HCI, modelling the appearance variability of target remains imperative due to the intrinsic (e.g. pose variation deformations in shape, scale, and out-of-plane rotations) and extrinsic (e.g. occlusions, illumination changes, and different camera viewpoint) variations.

Many researchers have attempted to address these issues in visual tracking and proposed various complex models to deal with target appearance variations (Wu, Lim, & Yang, 2015). These studies revealed the performances of existing methods, each of which has its own advantages and drawbacks (Smeulders et al., 2014). In visual tracking for face based-HCI such as facial expression recognition (FER), template drift (Matthews, Ishikawa, & Baker, 2004) is one of the common issues because of the accumulation of small errors in the template updating process. For effective appearance representation of a target face, frequent template update is usually necessary to cope with varying pose and head orientations. An inadequate updating strategy will ruin the purpose of appearance representation. In order to obtain a good trade-off between the processing time and accuracy of tracker, the template-updating process must be developed carefully.

Secondly, the template-updating strategies based on the image-as-vector form (Ning, Yang, Jiang, Zhang, & Yang, 2016; Ross, Lim, Lin, & Yang, 2008) ignore the fact that image is intrinsically a matrix, or a 2nd order tensor. A significant amount of spatial correlation within the original structure of a 2-D image remained unexploited in the image-as-vector form, which makes the appearance model less discriminative in tracking against occlusions. Alternately, the multiway or tensor based image representation can provide a better appearance structure for tracking by preserving the actual 2-D structure of an image to facilitate visual tracking.

This paper focuses on building a tracking system based on the tensor framework and presents a real-time application for fa-

* Corresponding author.
E-mail addresses: shehekhan2-c@my.cityu.edu.hk, sheheryar1984@gmail.com (S. Khan), guoxiaxu@cityu.edu.hk (G. Xu), rchan@math.cuhk.edu.hk (R. Chan), h.yan@cityu.edu.hk (H. Yan).

http://dx.doi.org/10.1016/j.eswa.2017.08.039
0957-4174/© 2017 Elsevier Ltd. All rights reserved.

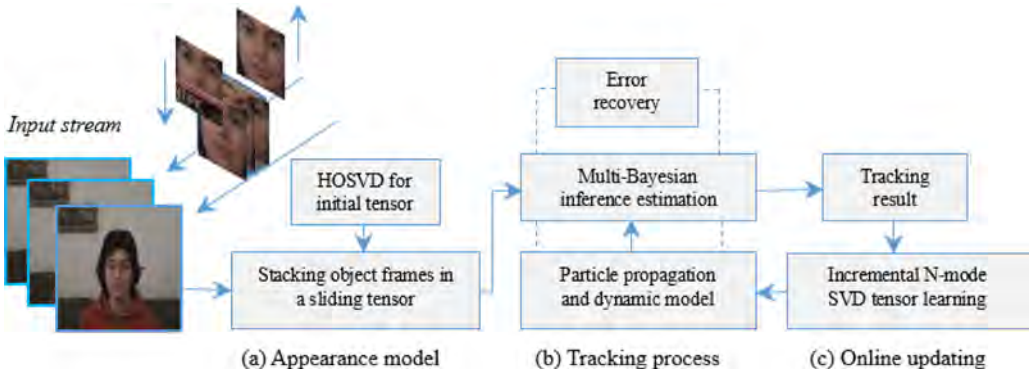

Fig 1. The architecture of the proposed online spatio-temporal tensor based learning model for visual tracking.

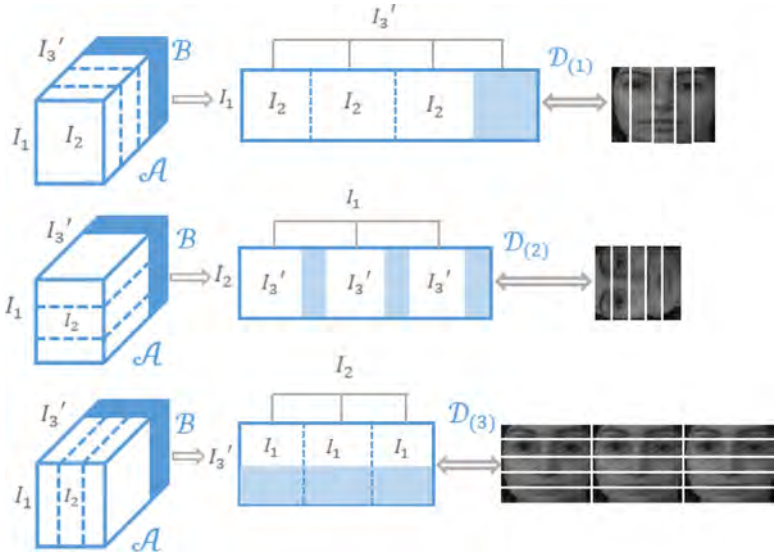

Fig 2. Tensor unfolding in respective modes and the analogous association with an image in terms of slices.

cial expression recognition. An important aspect of object motion in videos is their local similarity among several regions of the same frame. More importantly, an object in video also possesses the strong temporal correlation among succeeding frames. Based on these spatial and temporal correlation priors of a video, we construct the *spatio-temporal* tensor appearance model for object tracking. The proposed method effectively combines the dynamic model with a robust online tensor based eigen-basis updating strategy to better cope with scaling and geometric normalization issues of human faces, which is a key step in face-based HCI systems.

Performing the proposed learning procedure using the tensor representation will not only preserve the 2D structure of an image but also significantly circumvent the large dimensionality problem. For example, it can convert a 3rd order tensor of size $30 \times 30 \times 30$ to a smaller dimension of $10 \times 10 \times 10$. The image-as-vector form would require a $27{,}000 \times 1000$ basis matrix, but the tensor formulation requires only three sets of 30×10 basis matrices. Intuitively, the tensor based learning along with effective appearance updating yields fast and more reliable target localization, which is useful for building a robust real-time FER system.

1.1. Related works and context

A variety of tracking approaches have been proposed to achieve improved robustness, accuracy and computational efficiency. However, the performance of most trackers is constrained with certain conditions which makes them inapt for real applications (Wu et al., 2015). Discriminative and generative appearance models are widely accepted tracking approaches that effectively model the target appearance based on spatial information. Discriminative trackers treat a tracking task as a classification problem and discriminates the target from surroundings. Following the discriminative framework, an online supervised boosting method was proposed (Grabner & Bischof, 2006), whereas the semi-supervised tracker was introduced in (Grabner, Leistner, & Bischof, 2008), in which only the initial frame label was provided. Later Babenko, Yang, and Belongie (2009) introduced the Multiple Instance Learning (MIL) tracker to deal with unreliable positive and negative labels in an online manner and uncovered the drift issues in tracking. Recently the work presented in Ning et al. (2016) proposed the online structured support vector machine based discriminative tracking framework with fast learning and addressed the drift issues. The author in Bae, Kang, Liu, and Chung (2016) presented real-time object tracking framework based on discrete swarm optimization. However, the proposed strategies cannot deliver the orientation information of the target, or the degree of rotation of the tracking window.

On the other hand, several methods made use of the appearance modelling of an object based on the generative framework (Black & Jepson, 1998; Hu et al., 2011; Ross et al., 2008). Among them, sub-space learning-based models have gained much attention in visual tracking against model drifting due to the constant subspace assumption instead of the constant brightness assump-

tion. Moreover, the task of subspace learning is memory efficient, thus yielding comparatively faster processing. For instance, authors in Black and Jepson (1998) proposed view-based appearance models for tracking with sub-space learning. A view-based eigenbasis model of the target is trained off-line and tracking is performed on matching sequential views of a target. However, the lack of training samples along with maximum possible viewing conditions is still a challenging task. The work presented by Lim, Ross, Lin, and Yang (2004) accomplished tracking by incremental subspace learning, in which target subspace is updated during the tracking process to deal with appearance changes. Their work developed an updating strategy by extending the SKL (sequential Karhunen–Loeve) algorithm (Levy & Lindenbaum, 2000) but focused only on the similarity between candidate and target subspace.

Later, Ross et al. (2008) introduced an adaptive image-as-vector subspace learning model Incremental Visual Tracker (IVT), which gained much popularity. The IVT introduced the eigenbasis and mean updating strategy in tracking for updating the appearance variations sequentially. The template-updating strategy followed in IVT flattens the target regions to retain the vectorised shape, which yields the extrinsic information about the target subregions. Several other improved versions (Hu et al., 2011) of this model were further proposed. However, their performances degraded in unrestricted conditions. Extended version of IVT was proposed by Wang, Lu, and Yang (2013) under the Gaussian-Laplacian noise assumption to enhance the robustness in the presence of outliers. The representation of a target by the flattened intensity vector can result in a large dimensionality.

In visual tracking, the multilinear extension of object tracking based on online leaning is also introduced to capture spatio-temporal appearance and has gained much success. For instance, an online tensor decomposition based tracking framework was reported by Hu, Li, Zhang, Shi, Maybank, and Zhang (2011), in which target image-as-matrix is proposed for a better representation of spatial layout. For incremental updating, the R-SVD (Van Loan, 1996) approach is utilized to update the subspace along with sample mean against each tensor mode. However, the process of updating only considered the top eigenvalues and eigenvectors and small weights were discarded in order to meet the real-time requirements, which may cause error accumulation and model drift. Recently, the author in Ma, Huang, Shen, and Shao (2016) proposed the incremental tensor learning based pooling strategy and considered the target and template as sparse coding tensors. Although good results are achieved on tracking image sequences by using tensor pooling, however, a major concern of TPT is its extensive computing procedure consisting of: (1) the 4th order online tensor learning along with the 3rd order direct tensor decomposition on every upcoming frame. (2) K-means based dictionary learning in tensor pooling. These factors result in slower tracking frame rate and therefore make it feasible mainly for offline tracking.

An incremental N-mode SVD was proposed in Lee and Choi (2014) and tested on 3D face reconstruction to update the result of N-mode SVD with the arrival of new training data. The incremental N-mode SVD presented full factorization and more accurate calculation of the eigenstructure of the training tensor. In this work, we focus on N-mode SVD for incremental learning for online visual tracking. Unlike TPT, which uses the standard R-SVD to calculate the entire N-mode immediately, we conduct separate spatial and temporal factorization of the appearance tensor and adopt the incremental N-mode SVD for calculating the eigenstructure of the unfolded matrices. Our method captures variants of every mode independently for calculating the residue error prediction of Bayesian posterior probability. Compared with other incremental tensor subspace learning and vector-based methods, N-mode SVD delivers more accurate approximation of the unfolded tensors in each mode and updates the target appearance variations more effectively. Thus, it makes the tracking more robust against variations in pose, geometry and illumination.

For the task of expression recognition, the face detectors are usually employed to extract the facial region first. A well-known detection algorithm by Viola and Jones (2004) is extensively used over the years, which works on learning the classifiers based on Haar features. Despite the high detection rate of this technique, the performance degrades noticeably for occluded and profile faces. A real-time facial expression approach was presented (Geetha, Ramalingam, Palanivel, & Palaniappan, 2009), in which head contours were extracted to locate the face region in images. Color space information is further utilized to extract the location of face parts and then expression recognition is performed. This method heavily depends on morphological image operations such as thresholding of image pixel values and is therefore not suitable for low intensity and occluded images. (Wan, Shaohua, & Aggarwal, 2014) proposed a robust metric learning approach for spontaneous facial expression recognition, and (Owusu, Zhan, & Mao, 2014) developed a facial expression analysis system based on neural-AdaBoost. Recently the authors in Ali, Hariharan, Yaacob, and Adom (2015) used empirical mode decomposition to conduct facial expression recognition. In these reported studies, face detection was performed individually on every frame. To deal with the face orientation changes, generally several pre-processing techniques are combined carefully with this face detection framework to register the faces based on the locations of eyes and nose before the expression recognition. However, this stage demands accurate detection of facial parts, which is generally not possible in real-time applications.

Contributions: Based on the above-mentioned discussion, the correlation between the motion of object and its context is still hard to capture. We propose to address the visual tracking problem with an effective online spatio-temporal tensor learning framework, which not only takes into account the spatio-temporal information but also effectively combines the updating procedure with appearance modelling to achieve real-time tracking. The proposed tracking algorithm produces a low dimensional tensor representation of the target online by following an incremental update procedure of the mean and eigen-basis using the N-mode SVD of unfolded matrices. When estimating the target, the likelihood of a candidate is evaluated on the learned tensor subspace repeatedly based on the reconstruction error to avoid missing the target position. The spatio-temporal appearance feature and fast incremental update provide an improved tracking performance along with better computational efficiency when compared to vector based subspace methods. Then we specifically designed a real-time FER system based on the proposed tracking strategy. The tracking window obtained from the proposed tracker is further processed to align the face geometry and then the task of expression recognition is performed. Experiments revealed that the integrated system performs effectively in both recorded videos as well as on live camera enabled videos.

The remaining of this work is organized as follows. In Section 2, we describe the material and methods utilized in our proposed framework with a complete outline of our tracking approach. Section 3 provides experiment results of the proposed tracking algorithm. Section 4 discusses the applications of FER using our method. Finally, we present the conclusion in Section 5.

2. Proposed framework for tracking

2.1. Outline of our tracking method

The proposed tracking framework is built on three main stages as shown in Fig. 1: (a) spatio-temporal tensor based target appearance model, (b) Bayesian inference coupled with particle filter,

Table 1
The incremental N-mode SVD algorithm for updating the tensor based appearance model.

Algorithm 1: ***Incremental N-Mode SVD***
Function: Incremental N-Mode SVD($C_{(k)}$, $\mathcal{U}_k$, $A_{(k)}, B_{(k)}, M_k$, ff, t)
Input: Unfolded core tensor $\mathbf{C}_{(k)}$, projection matrix $\mathcal{U}_k$, unfolded stacked tensor $\boldsymbol{A}_{(k)}$, unfolded newly added tensor $\boldsymbol{B}_{(k)}$, forgetting factor $\boldsymbol{ff}$, updated mean $\boldsymbol{M}_k$, time t.
Repeat
If t = 0, then:
1: Apply **HOSVD** using Eq. (2) to get [$\mathcal{U}_n$, $C_{(n)}$]
2: $M_{(t)} = \frac{n_a}{n_a+n_b}\overline{\mathcal{A}_{(k)}} + \frac{n_b}{n_a+n_b}\overline{\mathcal{B}_{(k)}}$ the mean of concatenated matrices $\mathcal{A}_{(k)}$ and $\mathcal{B}_{(k)}$
3: t=t+1
Else:
1: Apply N-mode updating strategy:
(a) Spatial update (when $\boldsymbol{k} \neq \boldsymbol{N}$); Eqs. (4)–((7))
(b) Temporal update (where k = N); Eqs. (8)–((11))
2: $M_{(t)} = \frac{n_a}{n_a+n_b}\overline{\mathcal{A}_n} + \frac{n_b}{n_a+n_b}\overline{\mathcal{B}_n}$
3: t=t+1
End if
Iteration until the end of video.
End

and (c) N-mode SVD for incremental updating tensor. In Fig. 1(a), we propose a tensor based approach to construct the target template appearance as a reservoir of 3rd order tensors. A tensor is initially decomposed using Higher-Order Singular Value Decomposition (HOSVD) (Kolda, Tamara, & Brett, 2009), where the appearance model only takes into consideration of the initial region of interest to be tracked in subsequent frames. In Fig. 1(b), we consider the tracking problem in a generative framework as an online tensor learning task. An accurate subspace representation is learned online, and the updating procedure is carried out in the temporal direction. The processes (b) and (c) are combined such that the subspace of the target is computed by incremental N-mode SVD over the target's intensity-value template and is stored in a leaking memory to gradually forget old observations. The procedure is summarized in Table 1. Sampling of the candidate window, which is assumed to follow a Gaussian distribution around the preceding position, is carried out by using particle filtering. When predicting the target, the confidence of each sample in terms of distance from candidate to learned tensor subspace is computed. The sample with lowest error is then selected. Furthermore, the error reconstruction stage allows us to repeat the sampling when the confidence level is not sufficient enough to identify the candidate as the target. The whole process is repeated, where the new frame is added to the reservoir and the last frame is removed to provide sufficient spatiotemporal information and to avoid unnecessary storage.

2.2. Tensor decomposition

A higher order tensor can be viewed as a generalization of a vector (first-order tensor) and a matrix (second-order tensor). An N-th order tensor can be denoted as $\mathcal{A} \in \mathcal{R}^{I_1 \times I_2 \times \cdots \times I_N}$, each element of which can be represented as $a_{i_1 \cdots i_n \cdots i_N}$ for $1 \leq i_n \leq n_N$. The n-mode (Nth dimension) matrix unfolding of a tensor $\mathcal{A}$, denoted as $\mathcal{A}_{(n)} \in \mathcal{R}^{n_n \times (\prod_{i \neq n} n_i)}$, is obtained by fixing the index term i_n while unfolding all other modes and combining all other indices into one index. For better visualization, let us consider the process of unfolding the tensor $\mathcal{D}$ into its 1st, 2nd and 3rd modes. The mode-n folding process of a tensor is the reverse process of mode-n unfolding, which restores the actual tensor. The entries of the mode-n product of $\mathcal{A}$ and a matrix $M \in \mathcal{R}^{n_n \times m_n M}$ are:

$$(\mathcal{A} \times_n M)_{i_1 \cdots i_{n-1} i_{m_n} i_{n+1} \cdots i_N} = \sum_{i_n} a_{i_1 \cdots i_n \cdots i_N} M_{n_n m_n} \tag{1}$$

In a tensor and matrix multiplication, the resulting tensor C can also be computed by matrix multiplication $C_{(n)} = M\mathcal{A}_{(n)}$ followed by mode-n folding. Note that for tensors and matrices of the appropriate sizes, $\mathcal{A} \times_m M \times_n N = \mathcal{A} \times_n N \times_m M$ and $(\mathcal{A} \times_n M) \times_n N = \mathcal{A} \times_n (MN)$.

HOSVD is a multilinear extension of the conventional matrix singular value decomposition (SVD). For an N-th order tensor, HOSVD produces N orthonormal matrices, $U^{(1)}, U^{(2)}, \ldots U^{(N)}$, spanning N spaces. An N-th order tensor $\mathcal{A} \in \mathcal{R}^{I_1 \times I_2 \times \cdots \times I_N}$ can be decomposed as follows:

$$\min_{\mathcal{S}, U^{(1)}, U^{(2)}, \ldots U^{(N)}} \left\| \mathcal{A} - C \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_N U^{(N)} \right\| \tag{2}$$

where $C \in \mathcal{R}^{R_1 \times R_2 \times \cdots \times R_N}$ is called the core tensor, and $U^{(n)} \in \mathcal{R}^{I_n \times R_n}$ contain singular vectors. The solution of the above equation can be given by Tucker Decomposition (Tucker, 1966), which is the preliminary step in obtaining the start-up tracking procedure and incremental tensor learning in our tracking algorithm.

2.3. Online tensor learning for tracking

Online tensor learning model for tracking is built from the streaming data. The first K target frames warped from sliding data are stored in terms of image gray levels in the initial window as $\mathcal{A} \in \mathcal{R}^{I_1 \times I_2 \times I_3}$, where I_1 and I_2 are the width and height of target and I_3 is the number of frames stacked in the tensor. Subsequently, the tensor $\mathcal{A}$ is decomposed using HOSVD into three orthogonal spaces by three orthonormal matrices $\mathcal{U}_1$, $\mathcal{U}_2$, and $\mathcal{U}_3$. When a new frame comes from the video stream, the last frame from the sliding block is removed. For each new frame, we may only need a portion of the mode matrices to further compute the SVD, rather than re-computing the whole tensor. Incremental SVD, in this case, serves the purpose to update the previous mode matrices with the arrival of new data.

The classic R-SVD algorithm operates on newly accessorial columns and rows in the matrix, but is based on the zero mean assumption. In multi-linear generalization (Lee & Choi, 2014) introduced the N-mode SVD to compute the eigen-basis of a tensor with the mean update (Hall, Marshall, & Martin, 2002) and therefore can keep tracking the subspace variations in each mode. For incremental updating of basis matrices, the incremental N-Mode SVD (Lee & Choi, 2014) is utilized. An extensive procedure for the incremental N-Mode SVD is followed to compute the eigen-basis with mean updating simultaneously. The process is summarized in Table 1. This algorithm can approximate the N-mode SVD efficiently with less memory and operate on a smaller portion of data each time from a relatively larger dataset. In this section, we provide an overview of N-Mode incremental update adapted for tensor based appearance model in context of visual tracking. The complete derivation of mode matrices can be found in Lee and Choi (2014).

After preparing the reservoir tensor, let $\mathcal{A} \in \mathcal{R}^{I_1 \times I_2 \times \cdots \times I_n^a}$ be the current tensor, and $\mathcal{B} \in \mathcal{R}^{I_1 \times I_2 \times \cdots \times I_n^b}$ be the data tensor with new video frame added, then the incremental procedure includes updating the mean and the total number of samples. $\mathcal{A}$ and $\mathcal{B}$ can be concatenated to form: $\mathcal{D} = [\mathcal{A}\,\mathcal{B}]_N$, where $\mathcal{D} \in \mathcal{R}^{I_1 \times I_2 \times \cdots \times (I_n^a + I_n^b)}$. We only need to compute the k-mode projection matrix $\mathcal{U}_k$ of $\mathcal{D}$ and the unfolding matrix $\mathcal{D}_k$ for $k = 1, 2 \ldots N-1$. The process is illustrated in Fig. 2, where three identical tensors are shown along with their respective unfolded matrices in three modes. The white region corresponds to the original tensor whereas the shaded portion represents the newly added tensor.

In order to update the appearance tensor based on previous projection matrices, when a new sample arrives, the incremental update procedure mainly consists of two parts: (1) spatial update, and (2) temporal update. In spatial update, we consider only

mode-1 and mode-2 unfolding of the tensor for updating. In temporal update, we consider the third mode for updating and the process is slightly different. The steps below formulate the factorization of both parts.

(1) Spatial update (when k ≠ N)

In the incremental procedure for unfolded tensor in mode-1 and mode-2, we compute the new projection mode matrix $\mathcal{U}_k^* = [\mathcal{U}_k \quad Q_k]\mathcal{U}_k'$, where $Q_k = orth(B_{(k)} - \mathcal{U}_k \mathcal{U}_k^T B_{(k)})$ and $orth()$ is obtained by the standard QR decomposition (Hall et al., 2002). The concatenation of previous tensor and newly arrived frame can be formed as: $= [\mathcal{A} \quad \mathcal{B}]_k$, replacing the tensor with projection matrices and core in the first two modes as:

$$\mathcal{D} = \left[\mathcal{U}_k \; C_{(k)} \left(\otimes_{j \neq k} \mathcal{U}_j \right)^T \quad B_{(k)} \right] \tag{3}$$

$$\mathcal{D} = [\; \mathcal{U}_k \quad Q_k \;] \begin{bmatrix} C_{(k)} & \mathcal{U}_k^T B_{(k)} \\ 0 & Q_k^T B_{(k)} \end{bmatrix} \left[\mathcal{U}_k \; C_{(k)} \left(\otimes_{j \neq k} \mathcal{U}_j \right)^T B_{(k)} \right] \tag{4}$$

$$\mathcal{U}_k' = orth\left(\begin{bmatrix} ff * C_{(k)} & \mathcal{U}_k^T \mathcal{B}_{(k)} \\ 0 & Q_k^T \mathcal{B}_{(k)} \end{bmatrix} \right) \tag{5}$$

$$\mathcal{U}_k^* = \left[\mathcal{U}_k \quad \text{orth}\left(\left[ff\mathcal{D}_{(k)} \quad \sqrt{\frac{ff * n_a * n_b}{ff * n_a + n_b}} \left(\bar{\mathcal{A}} - \bar{\mathcal{B}} \right) \right] \right) \right] * \mathcal{U}_k' \tag{6}$$

where *ff* is the forgetting factor (Ross et al., 2008) for concentrating more effect on newly arrived sample.

(2) Temporal Update (where k = N)

For incremental update of 3rd mode, the updating structure for $\mathcal{U}_N^*$ is different from the spatial structure in terms of concatenation and can be given as:

$$\mathcal{D} = \begin{bmatrix} \mathcal{A} \\ \mathcal{B} \end{bmatrix}_{k=N} \tag{7}$$

$$\mathcal{D} = \begin{bmatrix} \mathcal{U}_N & 0 \\ 0 & E \end{bmatrix} \begin{bmatrix} C_N \left(\otimes_{j \neq N} \mathcal{U}_j \right)^T \\ \mathcal{B}_{(N)} \end{bmatrix} \tag{8}$$

$$\mathcal{B}' = \mathcal{B}_{(N)} \left(\otimes_{j \neq N} \mathcal{U}_j \right) C_N^T \tag{9}$$

$$\mathcal{U}_N^* = \begin{bmatrix} \mathcal{U}_N & 0 \\ 0 & E \end{bmatrix} orth\left(\begin{bmatrix} C_N C_N^T & \mathcal{B}'^T \\ \mathcal{B}' & \mathcal{B}_{(N)} \mathcal{B}_{(N)}^T \end{bmatrix} \right) \begin{bmatrix} C_N \left(\otimes_{j \neq N} \mathcal{U}_j \right)^T \\ \mathcal{B}_{(N)} \end{bmatrix} \tag{10}$$

Instead of using the standard R-SVD (Hall et al., 2002) that calculates the entire N-mode $\mathcal{U}_{new}$ immediately, we propose to utilize spatial as well as temporal factorization of appearance tensor model that is updated dynamically using N-mode SVD. Tracking is achieved by the actual factorization of each unfolded matrix. The method provides an accurate approximation by keeping the dominant singular subspaces of current updating model. It incrementally builds Gaussian mixture models on each mode to describe the data falling into several classes in spatial domain and temporal domain incorporating the tensor multi-linear representations.

In the tracking framework, the newly arrived sample is categorized by evaluating its likelihood with the estimated subspace. The likelihood can be determined by the sum of the reconstruction residual error norms of the predictive Gaussian distribution. By the means of the notation of orthogonality in the tensor decomposition, we associate the dynamic Bayesian inference to approximate the distribution over the location of target. Let $\bar{X}$ be the mean of observations, where X represents the center of the distribution for each class. Assume $g_k = U_k^T (X - \bar{X})$, which represents the shift of observations to the mean $\bar{X}$. The residual error vector H_k, orthogonal to every vector in U_k, is given by:

$$H_k = \left(X - \bar{X}\right) - U_k U_k^T \left(X - \bar{X}\right) \tag{11}$$

The sum of residual error norms of a predictive state can be represented as:

$$error = \sum_{k=1}^{2} \left\| \left(X_{(k)} - \bar{X}_{(k)}\right) - \left(X_{(k)} - \bar{X}_{(k)}\right) \times_k \left(U_k U_k^T\right) \right\|^2 + \left\| \left(X_{(N)} - \bar{X}_{(N)}\right) - \left(X_{(N)} - \bar{X}_{(N)}\right) \times_N \left(U_N U_N^T\right) \right\|^2 \tag{12}$$

2.4. Tensor likelihood Bayesian inference

We use online tensor learning to model the tracking process under the assumption that the object state in the tracking framework exhibits the Markov chain state transition process, where the present state can be effectively estimated from its past states. In this model, the motion of target among consecutive frames is usually considered as an affine motion. Assume that a given state of target is $X_t = \{x_t, y_t, \theta_t, s_t, \beta_t, \varphi_t\}$ at time t, where the parameters are the x and y translations, rotation angle, scale, aspect ratio, and skewness, respectively. Now we consider the tracking formulation in Bayesian filtering framework, in which the hidden state of X_t of the target object at each time t is estimated with one of the k image observations $\mathcal{Z} = \{Z_1, Z_2, \ldots Z_k\}$, $\{Z_t \mid t = 1, 2 \ldots k\}$. Under this framework, the filtering Bayesian estimate of posterior $P(X_t|Z_t)$ can be given as:

$$P(X_t|Z_t) \propto P(Z_t|X_t) \int P(X_t \;|X_{t-1}) P(X_{t-1} \;|Z_t) d_{X_{t-1}} \tag{13}$$

where $P(Z_t|X_t)$ refers to the observation likelihood at time t, and $P(X_t \,|X_{t-1})$ corresponds to the motion model. In order to approximate the distribution over the target position and to draw the set of samples $X = \{X_t^{(i)}\}_{i=1}^{N_s}$, particle filter (Isard & Blake, 1996) is utilized. The optimal object state of the target X'_t in the present frame can be inferred by the maximum a posteriori (MAP) criterion:

$$X'_t = \; \text{argmax}_{X_t \epsilon X} P(X_t|Z_t) \tag{14}$$

In our implementation, the residual error determines the measure of similarity among candidate and the learned subspace. Thus, $P(Z_{t_i}|X_t)$ in our case is given by:

$$P(Z_{t_i}|X_t) \propto \exp(-\text{error}) \tag{15}$$

3. Experiments

For performance evaluation, the proposed online spatio-temporal tensor learning based tracker is validated on seven challenging videos. The chosen sequences comprise of various variations, including partial occlusion, pose and scale variations, illumination changes, background cluttering, rotations and impulse motions. For comparison, we conducted experiments on several related tracking methods. (1) Fragments-based tracking (FRAG Tracker) (Adam, Rivlin, & Shimshoni, 2006) (2) Vector subspace learning-based tracking algorithm (IVT tracker) (Ross et al., 2008). (3) Adaptive structural local sparse appearance model (ASLA Tracker) (Jia, Lu, & Yang, 2012), (4) Discriminant tracker based on circulant structure with kernels (CSK Tracker) Henriques, Caseiro, Martins, & Batista, 2012), and (4) Sparsity based collaborative model for tracking (SCM Tracker) (Zhong, Lu, & Yang, 2012). For a fair comparison, the proposed tracker is evaluated against these methods using the results provided by authors in the benchmark (Wu et al., 2015). In our experiments, the target region obtained from the video frames is normalized to the size of 32×32 pixels, and an initial tensor of length 15 in the 3rd mode is built for the representation of object appearance. The forgetting factor in the incremental N-mode SVD is set to 0.9, where the number of particles in the particle filter is set to 500. The assigned affine parameter values are [9,5,0.05,0.005,0,0].

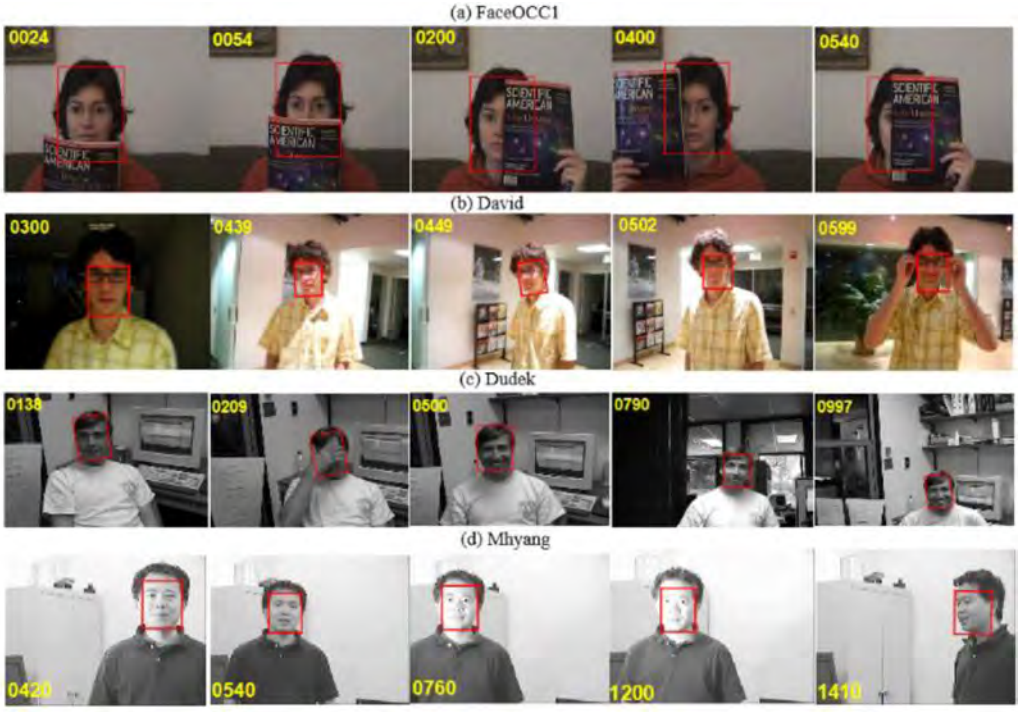

Fig 3. Screen shots of tracking results obtained by our method from key frames on face based videos in challenging sceneries.

Table 2
Comparison of tracking results obtained using existing trackers and the proposed tensor based tracker (TTracker) in terms of position error on seven videos. The best and second best results are shown in bold and italic fonts respectively.

Sequences	Tracking Methods					
	IVT	CSK	ASLA	SCM	Frag	TTracker
David	11.44	38.52	*5.59*	22.13	91.57	**5.17**
Fish	18.22	7.32	**3.40**	6.32	25.21	*5.08*
Mhyang	7.44	9.12	*2.03*	8.85	15.51	**1.91**
Twinnings	10.75	10.23	16.73	**8.04**	22.47	*9.24*
Clifbar	59.46	47.54	57.51	**31.67**	*40.83*	44.87
Dudek	*10.03*	19.76	14.95	27.61	87.70	**9.49**
Faceocc1	*18.74*	17.45	78.16	22.06	51.88	**15.98**
Average	19.44	21.42	25.48	*18.09*	47.88	**13.09**

3.1. Evaluation

As discussed above, the proposed methodology based on the image as a 2nd order tensor and appearance model as a 3rd order tensor can preserve more compact and useful information as compared to an image represented as a vector. In addition, the incremental N-mode SVD delivers a more robust tensor updating procedure. To evaluate the effectiveness of our proposed schemes, we present both quantitative and qualitative comparison with related methods against dominant challenges, such as occlusion, illumination changes, target scaling (deformation) and rotations.

3.1.1. Quantitative analysis

We used the conventional metric position error, precision plots of one pass evaluation (OPE) and success plot of OPE (Wu et al., 2015) to evaluate the tracking performance. The tracking windows obtained from IVT, FRAG, CSK, ASLA, SCM and our proposed algorithm are compared with the available ground truth to generate the mean square error. The average location error of each method on each video is listed in Table 2. Fig. 3 shows the screen shots of the proposed tracker under several challenging conditions. The relative position error per frame (in pixels) between the tracking result and ground truth is reported in Fig. 4, whereas the visual comparison on key frames is presented in Fig. 5.

It is evident from Table 2 that the proposed tracking method is more effective than other vector based methods. The advantage of our method is much notable with face based videos, for which our tracker achieved better performance. For other videos, our method provides the second best results with insignificant margins from the best results. The reason behind is that, other videos have fewer challenging conditions and do not contain abrupt motions. However, when the impulse motion and orientation changes occur, tensor based image representation can provide a better performance. SCM achieved the second best result following the tensor based tracker due to its frequent model updating strategy.

Overall performance: The overall performance of the related trackers is evaluated using the precision plots and success plots. The precision plot evaluates the robustness in terms of percentage of video frames whose recorded location is within the provided threshold distance to the ground-truth. Whereas the success plot is the measure of area under the curve (AUC) of each tracker. Success plot indicates the ratio among correctly tracked frames whose overlap threshold is larger than the given threshold. In terms of accuracy and overlap precision, the overall performance of the tracker was also found to be better. Fig. 6 shows precision plot ranking with the threshold of 20 pixels. The proposed tracker achieves 6% better precision as compared to SCM, whereas in success plot the proposed tracker achieves 5% better ranking over CSK in terms of AUC.

3.1.2. Qualitative analysis

The visual analysis of the proposed tracking method is also carried out under several challenging conditions.

Occlusion: Fig. 3(a) shows the results of tracking key frames with occlusion. The subject face in this sequence is being severely occluded by a book. The target in frame 50 covered major part of her face by the book, but accurate tracking is still achieved using

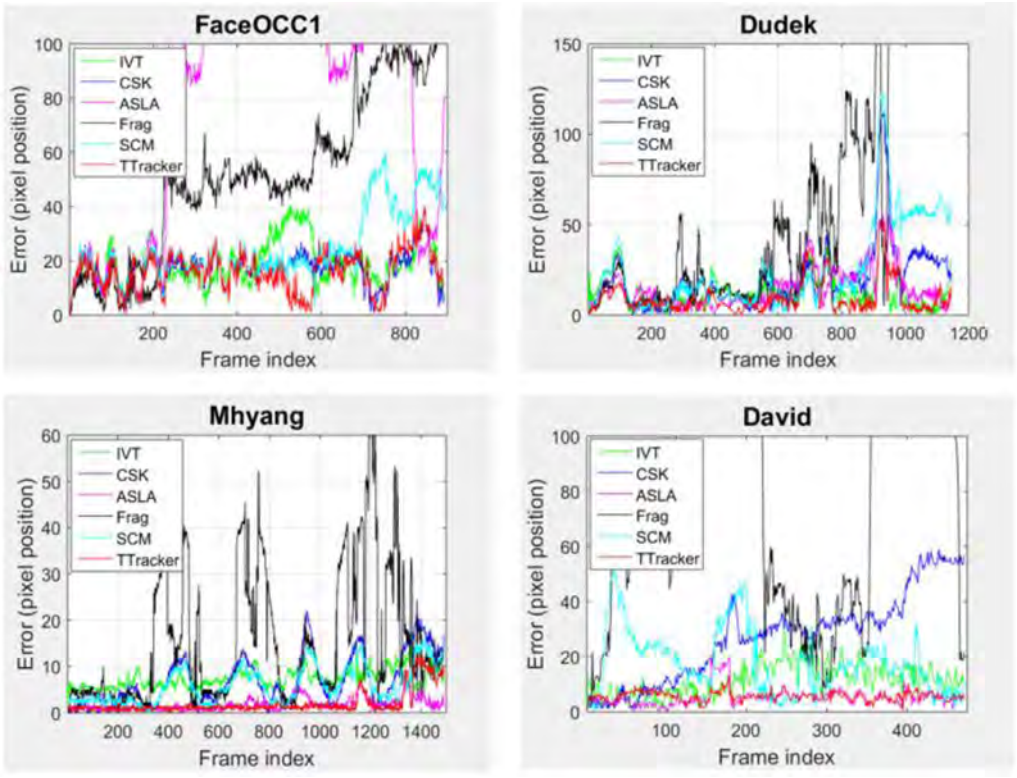

Fig 4. Position errors (pixels) with respect to ground truth and comparison of different trackers on face based videos.

Fig 5. Screen shots of tracking results obtained by existing trackers in comparison with the proposed tensor based tracker (TTracker).

our method as the tensorial information of the target is retained to compute the similarity, which makes it less susceptible to noise.

In terms of locating tracked objects, our method also provides good accuracy. Fig. 4 shows the result on the faceOCC1 sequence, in which the proposed method performed better than ASLA and FRAG due to effective updating. The error rates of CSK and IVT are comparable to ours. FRAG performs poorly in the occlusion scenario, due to the lack of mechanism to deal with the appearance changes.

Illumination variation and scale changes: The tracking results for videos with severe illumination changes is depicted in the video sequence named *Mhyang* as shown in Fig. 3(d). In frame *540*, the

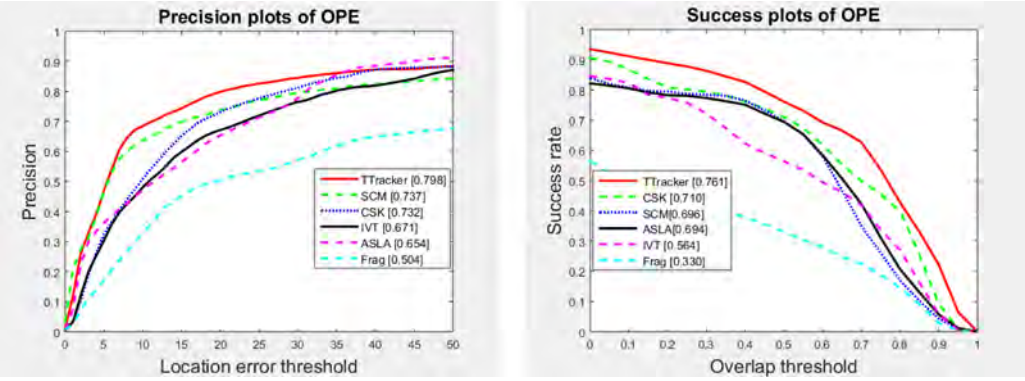

Fig 6. Comparison of different trackers in terms of the precision and success plots of the one pass evaluation (OPE) measure.

Fig 7. Results obtained by the proposed tracker on key frames taken from sample videos of MMI and MUG facial expression database under head rotations and varying expressions (upper row). Images on the lower row show cropped faces with rotation corrected face geometry.

target moves backward from its position and resulting in scale changes, whereas in frames *760* and *1200*, the illumination effect is more significant, which makes the tracking process more challenging. Frame *1410* is chosen to indicate the target rotation. The proposed method performs effectively under these conditions.

The video sequence provided in Fig. 3(b) shows a person walking out of the dark place into an area with spot lights, where the motion of target and camera is also found. In this sequence, our tracker and IVT successfully retained the tracking region throughout the sequence, whereas FRAG lost the tracking process and drifted away from its position. As evident form Fig. 5(c) and (d), our tracker best places the tracking window on the target and accurately captures the face rotation along with the side pose, while the ASLA and IVT windows are scaled slightly larger than the target.

The sample results shown in Fig. 3(c) are taken from video sequence *Dudek*, which involves illumination, scale and pose variations. Frame *209* shows the tracking result under quick face occlusion, whereas frames *500* and *997* show the expression variation. The target is also under the motion with changing background and scale. The proposed tracker quickly adapts the changes in scale and poses as evidenced in the results.

The effect of head pose changing on tracking result in the comparison to others is more evident in Fig. 5(b). In frame *250*, all other trackers captured the face location but the head rotation is more accurately located by TTracker. Moreover, the scale changes in subsequent frames of Fig. 5(b) is modelled effectively with TTracker, where ASLA and CSK localized a larger region and FRAG tracker completely lost the target. This is due to the proposed incremental template updating mechanism, which empowers the tracker to cope with gradual appearance changes in our method, whereas the classical approaches were not able to do this.

The good performance of the proposed tracking framework can be credited by the fact that the tensor framework delivers more structural motion information of the target as compared to vector based strategies. In terms of adaptability, the incremental tensor based learning through the N-mode SVD could learn more specific appearance changes of the target by capturing the variations with the passage of time. Meanwhile, the tensor reservoir in our proposed model serves the purpose of retaining the information in each mode. In case the information content related to one mode of

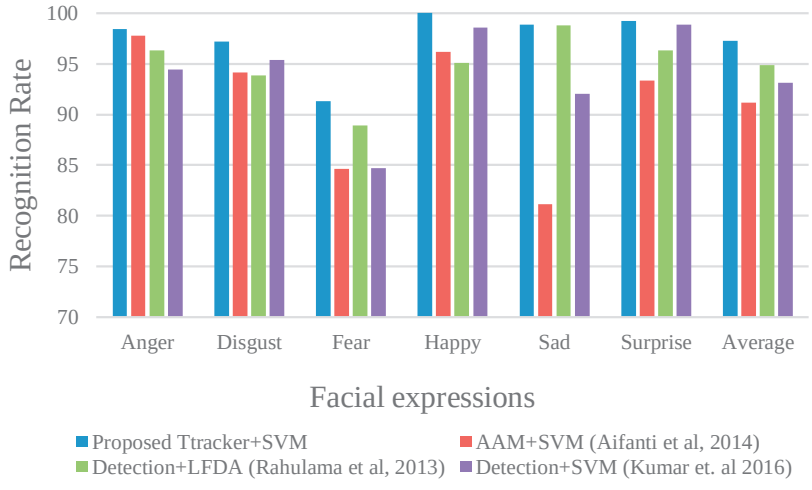

Fig 8. Performance comparison of the proposed TTracker based FER with existing facial expression recognition methods on MUG dataset.

Fig 9. Screen shot of GUI for expression recognition using the proposed tracker for offline videos as well as online ones taken by a camera.

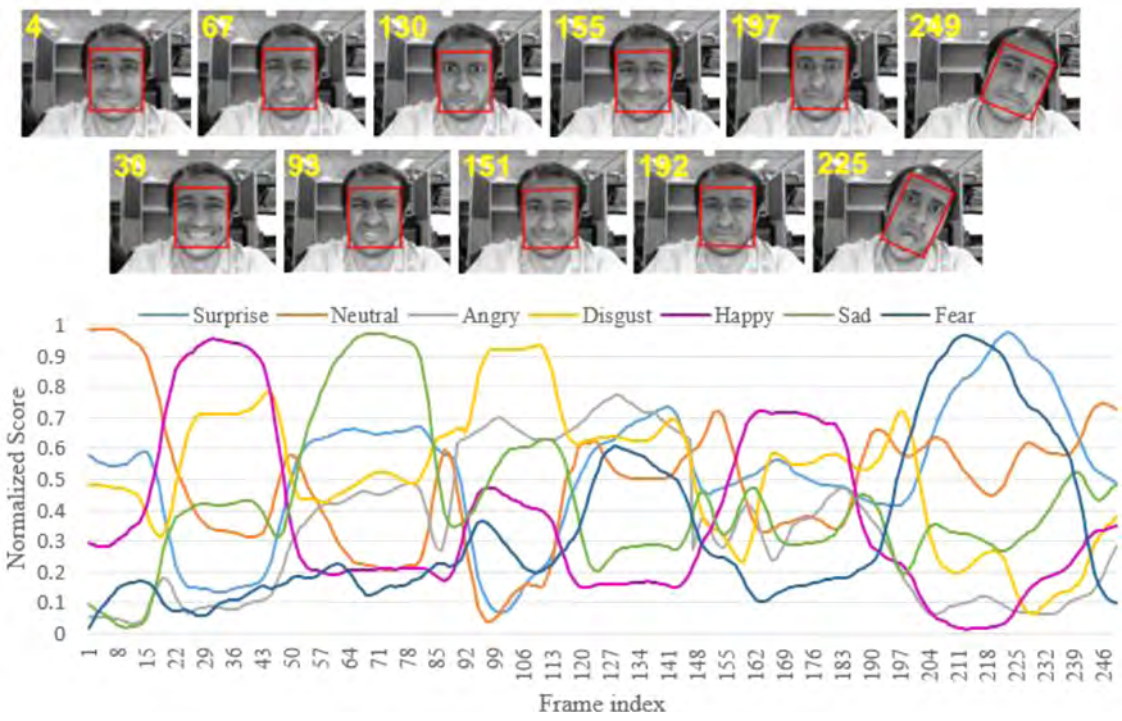

Fig 10. Test results of a subject posing facial expressions in front of a live camera. The waveform indicates the expression confidence in terms of normalized score for each frame, while the test expressions belonging to corresponding key frames along with tracking results are displayed in the images above.

the tensor is affected, the remaining modes are still enough to restore the subspaces contents. Hence, the minor imprecisions in the template location do not accumulate further, and thus the procedure assists the tracker to tolerate the template drift. We presented the visual results related to face based images, so as to emphasize the tracking performance under the natural head movements and orientation changes. Our method is particularly useful for face based HCI, such as expression recognition.

4. Application to human facial expression recognition

Conducting real-time face tracking to examine the facial expressions is precluded by problems such as head pose orientation, scale variation and at the same time the paucity of fast processing framework. On the other hand, face tracking provides a promising tool to track the initial face position of subject for subsequent frames with less computational complexity. Meanwhile, the scale

and rotation information can also be effectively determined along with face tracking, which can sufficiently avoid the need of face registration.

In this section, we present the real-time facial expression recognition system based on the proposed tracker. We have considered a seven-class recognition problem including the neutral expression, and six prototype expressions, Happy, Sad, Fear, Disgust, Anger and Surprise. We also present the performance evaluation on publically available facial expression datasets and also on self-collected videos.

4.1. Facial expression recognition

Generally, a facial expression recognition system involves two key phases: effective facial representation and accurate classifier design.

4.1.1. Feature extraction

In our work, we used Gabor filters to extract the facial information from the tracked face region. Gabor filters are widely accepted in many face based HCI systems, owing to their robustness against photometric disturbances and invariance to image registration issues (Amin & Yan, 2009; Haghighat, Zonouz, & Abdel-Mottaleb, 2015). The Gabor function in the spatial domain characterizes a Gaussian-shaped envelop modulated by a complex sinusoidal signal:

$$g(x,y) = \frac{1}{2\pi\delta_x\delta_y}\exp\left\{-\frac{1}{2}\left[(x/\delta_x)^2 + (y/\delta_y)^2\right] + i(ux+vy)\right. \quad (16)$$

A set of Gabor functions with multi-scales and orientations are employed for extracting more compact and effective image representation. We used 3 scales and 5 orientations of Gabor function in our experiment to extract the frequency contents of the image. Furthermore, 8-bit down-sampling is carried out to reduce the neighbourhood pixel redundancy.

4.1.2. Classification

The final stage of the FER system is based on classifier design. We used support vector machines (SVMs) for generalized performance. SVM is a binary discriminant classifier which is based on structural risk minimization principle that produces the maximum margin hyperplane between two classes. As we considered 7-class recognition problem, a multiclass SVM classifier can be constructed by using the one-against-all strategy (Weston & Watkins, 1998). Here we briefly present the multiclass SVMs used in our experiments. Given the training data of size N as; (g_1, l_1),..., (g_N, l_k) where, $g_j \in \Re^R$ feature vector and $l_j \in \{1, \ldots, 7\}$ represent the corresponding expression labels. Multiclass SVMs defines only one optimization problem but constructs seven class rules. So to follow the kth function $\boldsymbol{w}_k^T\phi(g_j) + b_k$ to partition training vectors of class k from remaining feature vectors, we minimize the following objective function:

$$\min_{w,b,\xi} \frac{1}{2}\sum_{k=1}^{7}\boldsymbol{w}_k^T\boldsymbol{w}_k + \sum_{j=1}^{N}\sum_{k\neq l_j}\xi_j^k \quad (17)$$

Subject to the constraints:

$$\boldsymbol{w}_{l_j}^T\phi\left(g_j\right) + b_{l_j} \geq \boldsymbol{w}_k^T\phi\left(g_j\right) + b_k + 2 - \xi_j^k$$
$$\xi_j^k \geq 0,\ j = 1, \ldots,\ N,\ \left\{k = 1, \ldots, 7\ |l_j\right\} \quad (18)$$

where, φ is the mapping function, C penalizes the training errors, $b = [b_1 \ldots b_7]^T$ is the bias vector and ξ is a slack variable, and $\xi = [\xi_1^1, \ldots, \xi_i^k, \ldots, \xi_N^7]^T$, whereas the decision function can be given by:

$$h(g) = \underset{k=1,\ldots,7}{\operatorname{argmax}}\left(\boldsymbol{w}_k^T\phi\left(g_j\right) + b_k\right) \quad (19)$$

Table 3
Comparison between Gabor wavelet based features and LBP features in terms of average recognition rate for different types of kernels used in the SVM classifier.

Feature Extraction	Recognition Rate (%)		
	SVM (linear)	SVM (polynomial)	SVM (RBF)
Gabor	94.16	92.31	91.04
LBP	90.87	89.54	92.24

After training the multiclass SVMs, a new feature vector from test image is classified using the equation above to recognize the facial expression.

4.2. Experiments on facial expression recognition

In this work, we used the proposed tracker to track the face location over several benchmark facial expression video datasets and performed the online facial expression recognition. We considered three widely used FER datasets, extended Cohn–Kanade dataset (CK+) (Lucey et al., 2010), MMI dataset (Pantic, Valstar, Rademaker, & Maat, 2005) and MUG dataset (Aifanti, Papachristou, & Delopoulos, 2010). In order to guarantee the generalization performance of the system, one of the dataset was used to prepare the training images, and the other two datasets were used for testing. As CK+ contains more subjects and videos, we used it for training. Challenging subject videos that contain sufficient head rotations from other datasets were used for testing. Fig. 7 shows the tracking result obtained from the proposed tracker on images taken from the MMI and MUG datasets with apex frames. The second row of Fig. 7 demonstrates the cropping result from the tracker window. It is evident that despite the presence of scale variation and head rotations, the cropped images are well aligned in terms of face geometry.

Table 3 presents the result of SVMs in terms of recognition rate against different kernels used. Apart from Gabor features, we also recorded the recognition rate for histogram based features based on local binary patterns (LBP) (Zavaschi, Britto, Oliveira, & Koerich, 2013). LBP features can be computed more efficiently than Gabor features, but are more sensitive to illumination variations and rotations. Gabor features with linear SVMs have the highest recognition rate compared with other kernels and LBP.

Tracking results from the proposed tracker were used to compare the performance of FER with the results obtained by other methods. We evaluated the results from several approaches, including face detector based method (Rahulamathavan, Phan, Chambers, & Parish, 2013), active appearance model based feature detector, (Aifanti & Delopoulos, 2014) and feature point based face model (Kumar, Bhuyan, & Chakraborty, 2016). Fig. 8 compares the results of 6 expressions on the MUG database using the leave-one-out validation strategy. The reason for using only 6 expressions here is that several methods above do not consider the neutral expression. It can be seen that the results from the proposed method were consistently higher for each expression. For the surprise expression, our recognition rate is comparable with the feature detection method, whereas for the fear expression, the margin between our method and the feature detection method is higher. The overall average recognition rate of our method is also better than all other algorithms. This fact can be credited by the accurate face registration using our proposed method, which yields comparatively better feature representation for FER.

4.3. Online experiments

A graphic user interface (GUI) is designed and utilized for conducting online experiments with videos taken by a camera.

Table 4
Confusion matrix of recognition rates obtained using linear SVM for seven facial expressions.

	Average recognition rate = 94.16%						
	Angry	*Disgust*	*Fear*	*Happy*	*Sad*	*Surprise*	*Neutral*
Angry	**91.16**	5.51	0	0	0	3.33	0
Disgust	7.84	**92.16**	0	0	0	0	0
Fear	0	0	**89.67**	0	0	6.64	3.69
Happy	0	0	0	**97.18**	0	0	2.82
Sad	0	0	0	0	**95.83**	0	5.17
Surprise	0	0	2.66	0	0	**97.34**	0
Neutral	0	0	0	0	4.21	0	**95.79**

Fig. 9 shows a snapshot of the GUI. A subject is asked to perform expressions in front of the camera and real-time expression recognition is performed. The figure shows the input frame with the tracking window and corresponding the aligned and cropped face region. The bottom graph indicates the confidence level of each expression for a test frame. A higher score indicates the presence of particular expression. The tracking window is obtained on every frame and is then processed further to obtain the affine transformed tracked image. Classification is done later and the expression label is generated. Fig. 10 shows the result of proposed tracker on 250 frames of a recorded video. The subject starts from neutral expression and plays 7 expressions. The top image shows the key frames of video with expression variation and the tracking window. It can be seen that the tracker follows the face region accurately over all 250 frames, despite head rotations. The bottom curves show the normalized score of each expression. It is interesting to note that, the neutral class, which is a transition expression between two universal expressions and is played for a very short time, is also being effectively recognized by the system. The main reason behind these refined results is the perfect face tracking that reduces the problems of miss alignment and registration of the face.

Apart from benchmark videos from MMI and MUG datasets, we also tested the proposed tracker on self-collected videos in order to quantify the recognition performance. Table 4 shows the confusion matrix for 7 expressions associated with these test data of 13 persons in total 30 videos. As can be seen from the table, the average recognition rate of 94.16% is achieved, where diagonal entries indicate the recognition rate of each expression. The surprise and happy expressions can be recognized with the highest accuracy, while we noticed that sad and neutral expressions are sometimes difficult to recognize, due to the fact that the sad expression is highly subject dependent, which sometimes resembles the neutral state.

5. Conclusion

In this paper, we propose a tensor based method to construct the target template appearance as a reservoir of 3rd order tensors and consider the object tracking problem in a generative framework as an online tensor learning task. An effective N-mode SVD based tensor eigenspace representation is learned online, and the updating procedure is carried out over the time span. The proposed multi-mode model is demonstrated for object tracking to better deal with large appearance variations caused by shape deformations, occlusions and drifts. Experiment comparisons with existing tracking strategies revealed the effectiveness of the proposed method, especially when the target motion is under rotational changes. Finally, the task of facial expression recognition is investigated by integrating the proposed tracking strategy with an expression recognition module. A GUI is developed and used for evaluation of our method on public and self-prepared videos. Our system can effectively recognize human facial expressions from videos and streaming camera with encouraging recognition rate of 94.16% for 7 classes of basic expressions. We believe that the proposed method will facilitate future development of face based-HCI applications and find useful applications to other object tracking and recognition systems.

Acknowledgement

This work is supported by Hong Kong Research Grants Council (Project C1007-15G).

References

Adam, A., Rivlin, E., & Shimshoni, I. (2006). Robust fragments-based tracking using the integral histogram. In *Computer vision and pattern recognition, 2006 IEEE computer society conference on, June: Vol. 1* (pp. 798–805). IEEE.

Aifanti, N., & Delopoulos, A. (2014). Linear subspaces for facial expression recognition. *Signal Processing: Image Communication, 29*(1), 177–188.

Aifanti, N., Papachristou, C., & Delopoulos, A. (2010). The MUG facial expression database. In *Image analysis for multimedia interactive services (WIAMIS), 2010 11th international workshop on, April* (pp. 1–4). IEEE.

Ali, H., Hariharan, M., Yaacob, S., & Adom, A. H. (2015). Facial emotion recognition using empirical mode decomposition. *Expert Systems with Applications, 42*(3), 1261–1277.

Amin, M. A., & Yan, H. (2009). An empirical study on the characteristics of Gabor representations for face recognition. *International Journal of Pattern Recognition and Artificial Intelligence, 23*(3), 401–431.

Babenko, B., Yang, M. H., & Belongie, S. (2009). Visual tracking with online multiple instance learning. In *Computer vision and pattern recognition, 2009. CVPR 2009. IEEE conference on, June* (pp. 983–990). IEEE.

Bae, C., Kang, K., Liu, G., & Chung, Y. Y. (2016). A novel real time video tracking framework using adaptive discrete swarm optimization. *Expert Systems with Applications, 64*, 385–399.

Black, M. J., & Jepson, A. D. (1998). Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision, 26*(1), 63–84.

Geetha, A., Ramalingam, V., Palanivel, S., & Palaniappan, B. (2009). Facial expression recognition–A real time approach. *Expert Systems with Applications, 36*(1), 303–308.

Grabner, H., & Bischof, H. (2006). On-line boosting and vision. In *Computer vision and pattern recognition, 2006 IEEE computer society conference on, June: Vol. 1* (pp. 260–267). IEEE.

Grabner, H., Leistner, C., & Bischof, H. (2008). Semi-supervised on-line boosting for robust tracking. *Computer Vision–ECCV, 2008*, 234–247.

Haghighat, M., Zonouz, S., & Abdel-Mottaleb, M. (2015). CloudID: Trustworthy cloud-based and cross-enterprise biometric identification. *Expert Systems with Applications, 42*(21), 7905–7916.

Hall, P., Marshall, D., & Martin, R. (2002). Adding and subtracting eigenspaces with eigenvalue decomposition and singular value decomposition. *Image and Vision Computing, 20*(13), 1009–1016.

Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2012). Exploiting the circulant structure of tracking-by-detection with kernels. In *European conference on computer vision, October* (pp. 702–715). Berlin, Heidelberg: Springer.

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

Isard, M., & Blake, A. (1996). Contour tracking by stochastic propagation of conditional density. In *European conference on computer vision, April* (pp. 343–356). Berlin Heidelberg: Springer.

Jia, X., Lu, H., & Yang, M. H. (2012). Visual tracking via adaptive structural local sparse appearance model. In *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on, June* (pp. 1822–1829). IEEE.

Jo, J., Lee, S. J., Park, K. R., Kim, I. J., & Kim, J. (2014). Detecting driver drowsiness using feature-level fusion and user-specific classification. *Expert Systems with Applications, 41*(4), 1139–1152.

Kolda, T. G., & Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review, 51*(3), 455–500.

Kumar, S., Bhuyan, M. K., & Chakraborty, B. K. (2016). An efficient face model for facial expression recognition. In *Communication (NCC), 2016 twenty second national conference on, March* (pp. 1–6). IEEE.

Lee, M., & Choi, C. H. (2014). Incremental N-mode SVD for large-scale multilinear generative models. *IEEE Transactions on Image Processing, 23*(10), 4255–4269.

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

Lim, J., Ross, D. A., Lin, R. S., & Yang, M. H. (2004). Incremental learning for visual tracking. In *Nips, December: Vol. 17* (pp. 793–800).

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer vision and pattern recognition workshops (CVPRW), 2010 IEEE computer society conference on, June* (pp. 94–101). IEEE.

Ma, B., Huang, L., Shen, J., & Shao, L. (2016). Discriminative tracking using tensor pooling. *IEEE Transactions on Cybernetics, 46*(11), 2411–2422.
Matthews, I., Ishikawa, T., & Baker, S. (2004). The template update problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26*(6), 810–815.
Ning, J., Yang, J., Jiang, S., Zhang, L., & Yang, M. H. (2016). Object tracking via dual linear structured SVM and explicit feature map. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4266–4274).
Owusu, E., Zhan, Y., & Mao, Q. R. (2014). A neural-AdaBoost based facial expression recognition system. *Expert Systems with Applications, 41*(7), 3383–3390.
Pantic, M., Valstar, M., Rademaker, R., & Maat, L. (2005). Web-based database for facial expression analysis. In *Multimedia and expo, 2005. ICME 2005. IEEE international conference on, July* (pp. 5–pp). IEEE.
Rahulamathavan, Y., Phan, R. C. W., Chambers, J. A., & Parish, D. J. (2013). Facial expression recognition in the encrypted domain based on local fisher discriminant analysis. *IEEE Transactions on Affective Computing, 4*(1), 83–92.
Ross, D. A., Lim, J., Lin, R. S., & Yang, M. H. (2008). Incremental learning for robust visual tracking. *International Journal of Computer Vision, 77*(1), 125–141.
Smeulders, A. W., Chu, D. M., Cucchiara, R., Calderara, S., Dehghan, A., & Shah, M. (2014). Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(7), 1442–1468.
Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika, 31*(3), 279–311.
Van Loan, C. F. (1996). *Matrix computations*. Johns Hopkins studies in mathematical sciences.
Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision, 57*(2), 137–154.
Wan, S., & Aggarwal, J. K. (2014). Spontaneous facial expression recognition: A robust metric learning approach. *Pattern Recognition, 47*(5), 1859–1868.
Wang, D., Lu, H., & Yang, M. H. (2013). Online object tracking with sparse prototypes. *IEEE Transactions on Image Processing, 22*(1), 314–325.
Weston, J., & Watkins, C. (1998). Multi-class support vector machines. Technical Report CSD-TR-98-04, May, Department of Computer Science, Royal Holloway, University of London.
Wu, Y., Lim, J., & Yang, M. H. (2015). Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(9), 1834–1848.
Zavaschi, T. H., Britto, A. S., Oliveira, L. E., & Koerich, A. L. (2013). Fusion of feature sets and classifiers for facial expression recognition. *Expert Systems with Applications, 40*(2), 646–655.
Zhong, W., Lu, H., & Yang, M. H. (2012). Robust object tracking via sparsity-based collaborative model. In *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on* (pp. 1838–1845). IEEE.