Signal Processing 161 (2019) 74–88

Contents lists available at ScienceDirect

Signal Processing

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

FERLrTc: 2D+3D facial expression recognition via low-rank tensor completion

Yunfang Fu[a,b,c,*], Qiuqi Ruan[a,c], Ziyan Luo[d], Yi Jin[a,c], Gaoyun An[a,c], Jun Wan[e]

[a] Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China
[b] School of Computer Science & Engineering, Shijiazhuang University, Shijiazhuang 050035, China
[c] Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China
[d] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China
[e] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:
Received 16 September 2018
Revised 11 March 2019
Accepted 18 March 2019
Available online 20 March 2019

Keywords:
Tensor Tucker decomposition
2D+3D Facial expression recognition
Tensor low-rank representation
Tensor completion
Multi-modality

ABSTRACT

In this paper, a 4D tensor model is firstly constructed to explore efficient structural information and correlations from multi-modal data (both 2D and 3D face data). As the dimensionality of the generated 4D tensor is high, a tensor dimensionality reduction technique is in need. Since many real-world high-order data often reside in a low dimensional subspace, Tucker decomposition as a powerful technique is utilized to capture multilinear low-rank structure and to extract useful information from the generated 4D tensor data. Our goal is to use Tucker decomposition to obtain a set of core tensors with smaller sizes and factor matrices which are projected into the 4D tensor data for classification prediction. To characterize the involved similarities of the 4D tensor, the low-rank and sparse representation is built in terms of the low-rank structure of factor matrices and the sparsity of the core tensor in the Tucker decomposition of the generated 4D tensor. A tensor completion (TC) framework is embedded to recover the missing information in the 4D tensor modeling process. Thus, a novel tensor dimensionality reduction approach for 2D+3D facial expression recognition via low-rank tensor completion (FERLrTC) is proposed to solve the factor matrices in a majorization–minimization manner by using a rank reduction strategy. Numerical experiments are conducted with a full implementation on the BU-3DFE and Bosphorus databases and synthetic data to illustrate the effectiveness of the proposed approach.

1. Introduction

Facial expression is the most cogent and naturally preeminent way for humans to communicate emotions and to regulate interactions with the environment or other people. Nowadays, the facial expression recognition (FER) has received enormous attention and played an important role in computer vision, affective computing and multimedia research [1].

From the data modality perspective, existing FER approaches can be roughly classified into 2D FER, 3D FER and 2D+3D FER [2]. 2D FER methods often use 2D face images, while 3D FER approaches generally utilize 3D face shape models. 2D+3D FER methods usually employ both 2D and 3D face data (i.e., textured 3D face scans).

In last several decades, research on 2D facial expression recognition were mostly focused on the 2D modality of images and videos [3,4] or multi-modality fusion of visual and audio data [5]. To satisfy the requirements beyond the lab environment and to improve the FER accuracy, not only distinguishing spontaneous and posed expressions, but also removing non-expression related head movement were taken into consideration. The infrared facial images were also utilized in [6] to solve the illumination issue. With a limited capacity on capturing subtle facial deformations and on handling complicated circumstances such as wearing glasses, 2D FER methods that rely on facial texture analysis were largely affected by pose and illumination variations, which often appear in real scenes.

With the rapid development of 3D acquisition devices, multimodal FER (such as RGB, depth and RGB-D) has gained a lot of attention. Compared with the traditional 2D facial expression [7,8], 3D Facial expression exhibits more resistance to interference from the illumination and head pose variations. For example, the depth (z coordinate) information of its 3D physical coordinates (x, y, z) can capture the subtle face deformations caused by the movements

* Corresponding author at: Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China.
E-mail addresses: fu_yunfang@126.com (Y. Fu), qqruan@bjtu.edu.cn (Q. Ruan), zyluo@bjtu.edu.cn (Z. Luo), yjin@bjtu.edu.cn (Y. Jin), gyan@bjtu.edu.cn (G. An), jun.wan@nlpr.ia.ac.cn (J. Wan).

https://doi.org/10.1016/j.sigpro.2019.03.015
0165-1684/

of facial muscles. Various multi-modality data have then been used in multi-modal FER, including both 2D and 3D face data [2,9], visible and infrared face images [6], visual and audio [5], two-dimensional (2D) and 3D videos [10]. It has become a potential research interest because of the complementarity between different modalities. However, data representation in existing methods for FER do not maintain intrinsic structural information between multi-modal data.

To make up this deficiency, a new attempt is to construct a 4D tensor model in this paper. This 4D tensor model is composed of both 2D and 3D face data. The generated issues for such a data combination structure are mainly reflected in three aspects: (1) How to construct and represent a 4D tensor model from both 2D and 3D face data? (2) How to propose a tensor optimization model and solve it for FER? (3) How to evaluate the tensor optimization model? Nowadays, all kinds of features by geometric maps or texture maps are utilized to describe 3D facial shape information or 2D appearance information for FER, respectively. However, to the best of our knowledge, these generated features are not combined to construct a 4D tensor model by being stack for FER. Inspired by this fact, it is feasible to construct a 4D tensor model by stacking some discriminative features from 2D and 3D face data, which overcomes the issues that the number of training 3D faces is not enough and the dimensionality disaster due to high-dimensional vectorization features. This is the first contribution in the paper.

As the dimensionality of the generated 4D tensor is high, a tensor dimensionality reduction technique is in need. For many high-order data in the real world, they have the spatial redundancy information and are often located in a low dimensional subspace. And tensor decomposition, which is based on low-rank approximation, is a powerful technique to capture intrinsic multi-dimensional structure and to extract useful information from the high-order data. Meanwhile tensor decomposition is widely applied for tensor recovery [11], data classification [12], and harmonic retrieval [13]. Tucker decomposition and CANDECOMP/PARAFAC (CP) decomposition [14] are the two popularly adopted low-rank tensor decomposition forms. Tucker decomposition decomposes a tensor into a product of a core tensor and a number of factor matrices, whereas CP decomposes a tensor as a sum of rank-one Kronecker bases. For different types of data, Tucker decomposition has a better generalization ability compared with CP decomposition [15]. Thus, we focus on Tucker decomposition in this paper. The relative tensor basics, Tucker decomposition, tensor projection and tensor reconstruction are introduced in Section 2. Based on a Tucker decomposition of the high-order tensor, its spatial redundancy information is reflected in the spatial structure of the generated factor matrices that are used for projection. Hence, the Tucker decomposition technique is applied to multi-modal FER for the first time, which is the second contribution.

As one sees, when extracting all kinds of features by geometric maps (3D face data) or texture maps (2D face data), similarities among samples will unavoidably be generated and some useful intrinsic information will possibly be missed. To characterize the involved similarities of the 4D tensor model, the low-rank and sparse representation [16] is built in terms of the low-rank structure of factor matrices and the sparsity of the core tensor in the Tucker decomposition of the generated 4D tensor. A tensor completion (TC) framework is embedded to recover the facial expression data. Thus, a new tensor dimensionality reduction approach for 2D+3D FER via low-rank tensor completion (FERLrTC) is proposed, in which our goal is to find a set of core tensors with smaller sizes and a set of factor matrices which are projected into the generated 4D tensor for classification prediction, i.e., the dimensionality of the 4D tensor is reduced. This is the third contribution. Its detailed flowchart in Fig. 1. Our proposed optimization algorithm is employed to handle the 4D tensor model which is also introduced to overcome the drawback of the missing information. Meanwhile a rank reduction strategy is designed to retain strong interactions among factor matrices and the core tensor by removing the redundancy information and to speed up the convergence processing, which is the fourth contribution. Our goal is to To verify the efficiency of our approach, the multi-class-SVM is utilized for the final facial expression prediction. In addition, the effectiveness of the 4D tensor model based on feature-level fusion, the complexity and convergency analyses and the effectiveness of the rank reduction strategy will be discussed accordingly for further validation.

The remainder of the paper is organized as follows. In Section 2, some related works including 3D and 2D+3D FER, and tensor representations are reviewed. A new model based on low-rank tensor completion is proposed and solved in Section 3. Experiment results and analysis are given in Section 4. Conclusions are drawn in Section 5.

2. Related works

Some related existing works on 3D and 2D+3D FER are reviewed and some preliminaries on tensors are recalled in this section.

2.1. Related works on 3D and 2D+3D FER

Existing approaches for 3D FER can be roughly categorized into two main streams [3,17,18]: the feature-based and model-based approaches. The feature-based approaches mainly extract local expression features around facial landmarks employing different surface geometric or differential quantities. For instance, local surface patch-based distances [3,9,19], 3D landmark distances [20–22], the depth [23,24], conformal images [25], surface normal [26,27], normal maps [2,17,28], curvatures [2] and mean curvature [29] are some popular expression features. In [18], a 4D tensor structure from 3D face data was constructed to explore efficient structural information. After a low-rank approximation method has reduced dimension of the original tensor data, nonnegative tensor factorization which is based on graph-preserving extracted local geometric and discriminant information. NN classifier was used to recognize facial expression. Note that [18] is the first time to utilize high-order Tucker decompositions for 3D FER. In [2,9,30], the merits of both 2D and 3D face data are combined for exploring multimodal 2D+3D FER approaches. For instance, Li et al. [2] proposed multimodal 2D+3D FER with deep fusion convolutional neural network (DF-CNN). DF-CNN consists of a feature extraction subnet, a feature fusion subnet, and a softmax-loss layer. Six types of facial attribute maps from each textured 3D face scan are then jointly sent into DF-CNN for feature extraction and feature fusion. Expression prediction is accomplished through two classifiers: one is to utilize the 32-dimensional fused deep features for learning linear SVM classifiers; the other is to use the 6-dimensional expression probabilities for softmax prediction.

For the model-based approaches, a generic 3D face template is obtained by averaging a quantity of neutral samples for training. And it fits to match unknown 3D face scans for testing. Meanwhile, the corresponding parameters or coefficients are employed for expression prediction. For example, a bilinear model was proposed to simultaneously recognize 3D face and 3D FER in [31], and it was generated from facial correspondence based on elastic deformation. After establishing a PCA-based deformation subspace, an asymmetric bilinear model is involved in a maximum-likelihood frame work for 3D FER. Zhao et al. [30] proposed a Statistical Facial Feature Model (SFAM) for automatic facial landmarking, both 2D texture and 3D shapes features were used around these landmarks

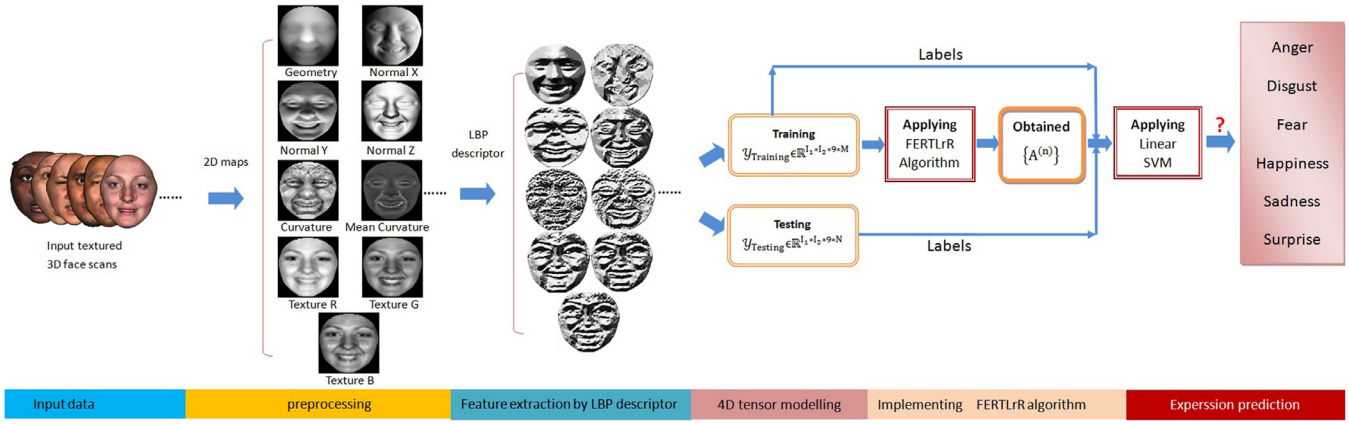

Fig. 1. A flowchart of the proposed approach (FERLrTC) on BU-3DFE database.

for expression prediction. In [32], Gong et al. described facial expression with the Expressional Shape Component (ESC). Due to the linear combination of neutral faces, the Basic Facial Shape Component (BFSC) did not contain expression information. ESC was then obtained by subtracting the depth map of BFSC from that of an input scan and was further utilized to build the feature vector.

The aforementioned approaches, either feature-based or model-based, have their own disadvantages. For example, most of the existing model-based approaches not only require high computation, but also are very sensitive to topological changes, such as mouth opening. The feature-based approaches generally need relatively low computation, while they substantially rely on the discriminative power of the local feature and mostly require manual or automatic landmarks. 2D+3D FER has emerged and becomes a hot research topic in pattern recognition due to the complementarity between different modalities. Along this research line, we focus on 2D+3D FER in this paper.

2.2. Related works on tensors

2.2.1. Notations and tensor basics

Throughout the paper, vectors, matrices and tensors will be represented by lowercase letters (e.g., y), capital letters (e.g., Y) and calligraphic letters (e.g., $\mathcal{Y}$), respectively. We use symbols $\otimes$, $\circ$ and $*$ to indicate the Kronecker, outer and Hadamard product, respectively. An N-order tensor can be denoted as $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \cdots \times I_N}$ with entries $\mathcal{Y}_{i_1 \cdots i_n} \in \mathbb{R} (1 \le i_n \le I_N)$. The Frobenius norm of $\mathcal{Y}$ is defined by $\|\mathcal{Y}\|_F^2 = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} \mathcal{Y}_{i_1 i_2 \ldots i_N}^2$.

In tensor operations, $Y_{(n)} \in \mathbb{R}^{I_n \prod_{k \ne n, k=1}^{N} I_k}$ is defined by the mode-n unfolding of $\mathcal{Y}$. The operator $\times_n$ shows the mode-n product, and $\mathcal{X} = \mathcal{Y} \times_n A^{(n)}$ indicates the mode-n product of $\mathcal{Y}$ with a matrix $A^{(n)}$ where $A^{(n)} \in \mathbb{R}^{I_n \times R_n}$ and $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \ldots \times R_n \times \ldots \times I_N}$. The rank of $\mathcal{Y}$ is defined as a N-tuple $(r_1(\mathcal{Y}), \ldots, r_N(\mathcal{Y}))$, where $r_n(\mathcal{Y}) = rank(Y_{(n)})$ for all $n = 1, \ldots, N$.

2.2.2. Multilinear tensor definitions

Tucker decomposition: Suppose $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_N}$, we can get a core tensor $\mathcal{X} \in \mathbb{R}^{R_1 \times R_2 \ldots \times R_N}$ and factor matrices $\{A^{(n)}\} \in \mathbb{R}^{I_n \times R_n}$ by a Tucker decomposition of $\mathcal{Y}$, which can be defined as follows:

$$\mathcal{Y} = \mathcal{X} \prod_{n=1}^{N} \times_n A^{(n)}. \tag{1}$$

Tensor projection: Given $\mathcal{Y}$ and the generated factor matrices $\{A^{(n)}\}$ obtained by a Tucker decomposition of $\mathcal{Y}$, then the projection of $\mathcal{Y}$ onto $\{A^{(n)}\}$ along each mode of $\mathcal{Y}$ is defined as $\mathcal{Y} \prod_{n=1}^{N} \times_n A^{(n)T}$.

Tensor reconstruction: The tensor $\mathcal{O} \in \mathbb{R}^{R_1 \times R_2 \ldots \times R_N}$ can be reconstructed by projecting $\mathcal{Y}$ onto its generated factor matrices $\{A^{(n)}\}$ by a Tucker decomposition of $\mathcal{Y}$, its definition is shown as follows

$$\mathcal{O} = \mathcal{Y} \prod_{n=1}^{N} \times_n A^{(n)T}. \tag{2}$$

2.2.3. Tensor low-rank representation

Low-rank representation (LrR) method, which is the most commonly used method of low-rank matrix recovery (LrMR) [33], stems from compressed sensing (CS) [34] and has been widely applied in many fields, such as image segmentation, motion segmentation and face recognition, etc [35]. Based on Tucker decomposition of $\mathcal{Y}$, a reasonable and favorable way to get a low-rank representation or approximation is to find some low-rank $A^{(n)}$'s to store and analyze the information of $\mathcal{Y}$, especially for large-scale cases. In this sense, we can adopt the following minimization problem to get such a low-rank representation of $\mathcal{Y}$:

$$\begin{aligned} &\min_{\mathcal{X}, \{A^{(n)}\}} \quad \sum_{n=1}^{N} \lambda_n \|A^{(n)}\|_* \\ &\text{s.t.} \quad \left\| \mathcal{Y} - \mathcal{X} \prod_{n=1}^{N} \times_n A^{(n)} \right\|_F^2 \le \varepsilon, \end{aligned} \tag{3}$$

where $\lambda_n > 0$ is the weight parameter for $A^{(n)}$ for each $n = 1, \ldots, N$ and ε is a prescribed accuracy parameter. $\|\cdot\|_*$ stands for the matrix nuclear norm equal to the sum of all singular values of the matrix.

2.2.4. Tensor sparse representation

Sparse representation (SR) method is also rooted from compressed sensing (CS) and has been broadly applied into machine learning, pattern recognition, signal processing, image processing, computer vision [36], etc. Based upon the Tucker decomposition, a sparse representation or approximation of a given N-order tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_N}$ can be achieved by

$$\begin{aligned} &\min_{\mathcal{X}, \{A^{(n)}\}} \quad \sum_{n=1}^{N} \|z_n\|_0 \\ &\text{s.t.} \quad \left\| \mathcal{Y} - \mathcal{X} \prod_{n=1}^{N} \times_n A^{(n)} \right\|_F^2 \le \varepsilon, \end{aligned} \tag{4}$$

where z_n is an I_n-dimensional vector with its ith element obtained by $z_{n,i} \triangleq \|\mathcal{X}_{(n,i)}\|^2$, where $\mathcal{X}_{(n,i)}$ represents the ith row of the n-mode unfolding matrix of $\mathcal{X}$ and $\|\mathcal{X}_{(n,i)}\|^2$ indicates the sum of

the squares of each element of $\mathcal{X}_{(n,i)}$. Obviously, the involved ℓ_0 norm [37,38] with combinatoric properties makes this problem NP-hard generally. Many different relaxation strategies have been proposed in the literature, such as the ℓ_1 norm relaxation, the log-sum penalty function, etc. In [39,40], the log-sum penalty function has been shown to be more sparsity-encouraging than the ℓ_1 norm. The detailed model takes the form of

$$\begin{aligned}\min_{\mathcal{X},\{A^{(n)}\}} \quad & \sum_{n=1}^{N}\sum_{i=1}^{I_n}\log(\|\mathcal{X}_{(n,i)}\|^2+\epsilon)\\ \text{s.t.} \quad & \left\|\mathcal{Y}-\mathcal{X}\prod_{n=1}^{N}\times_n A^{(n)}\right\|_F^2\le\varepsilon,\end{aligned} \tag{5}$$

where $\epsilon>0$ is some given approximation parameter.

3. The proposed FERLrTC approach

3.1. The low-rank tensor completion model

Here we propose a new tensor dimensionality reduction approach for 2D+3D FER, which deals with higher order tensors directly instead of converting them into vectors. Now, given M samples of 3D facial expressions with N features of size $I_1\times I_2$, a 4D tensor $\mathcal{Y}_0$ of size $I_1\times I_2\times N\times M$ is then constructed to store the feature information of all samples. Our goal is to find the low-rankness of factor matrices for projection by a Tucker decomposition of $\mathcal{Y}_0$. The resulting tensor $\mathcal{Y}_0$ will naturally admit some low-rank representation due to the high similarities among samples. Inspired by the way of using a group-based log-sum function to place structural sparsity over the core tensor in [40], we combine the low-rank structure of factor matrices and a group-based log-sum function over the core tensor to together characterize the involved similarities of the 4D tensor data $\mathcal{Y}_0$ based on Tucker decomposition. To achieve the required low-rankness of $\{A^{(n)}\}\in\mathbb{R}^{I_n\times R_n}(R_n<=I_n)$, the trace-norm, other than the Frobenius norm as introduced in [40], is imposed in the tensor optimization model. As information will partially be missed in the tensor modeling process, a tensor completion (TC) framework based Tucker decomposition is embedded, which is actually one of our major contributions. Thus, the general tensor optimization model of the proposed approach is as follows:

$$\begin{aligned}\min_{\mathcal{X},\{A^{(n)}\},\mathcal{Y}} \quad & \sum_{n=1}^{4}\sum_{i=1}^{I_n}\log(\|\mathcal{X}_{(n,i)}\|_F^2+\epsilon)+\gamma\sum_{n=1}^{4}\lambda_n\|A^{(n)}\|_*\\ \text{s.t.} \quad & \left\|\mathcal{Y}-\mathcal{X}\prod_{n=1}^{4}\times_n A^{(n)}\right\|_F^2\le\varepsilon,\\ & \Omega(\mathcal{Y})=\Omega(\mathcal{Y}_0),\end{aligned} \tag{6}$$

where γ ($\gamma>0$) is a tradeoff parameter to compromising the sparsity of the core tensor and the low-rank of factor matrices, λ_n is the weight of $A^{(n)}$, $\mathcal{Y}$ is the required reconstructed tensor, $\mathcal{Y}_0$ represents the facial expression data, and $\Omega(\mathcal{Y}_0)$ represents the nonzero entries in $\mathcal{Y}_0$.

3.2. Solving the tensor completion model

To effectively solve the problem (6), the majorization-minimization method (MM) [41] is employed to optimization the given objective function through iteratively minimizing a simple surrogate function. Its advantage is shown that the iterative process generates a non-increasing objective function value.

Before using the MM scheme, the Tikhonov regularization is utilized to approximate the original constrained problem (6) by

$$\begin{aligned}&\min_{\mathcal{X},\{A^{(n)}\},\mathcal{Y},\Omega(\mathcal{Y})=\Omega(\mathcal{Y}_0)} L(\mathcal{X},\{A^{(n)}\}_{n=1}^4,\mathcal{Y})\\ &=\sum_{n=1}^{4}\sum_{i=1}^{I_n}\log(\|\mathcal{X}_{(n,i)}\|_F^2+\epsilon)+\gamma\sum_{n=1}^{4}\lambda_n\|A^{(n)}\|_*\\ &\quad+\mu\left\|\mathcal{Y}-\mathcal{X}\prod_{n=1}^{4}\times_n A^{(n)}\right\|_F^2,\end{aligned} \tag{7}$$

where μ ($\mu>0$) is the regularization parameter. Obviously, the minimization function $L(\mathcal{X},\{A^{(n)}\}_{n=1}^4,\mathcal{Y})$ contains a joint term of factor matrices $\{A^{(n)}\}_{n=1}^4$, a core tensor $\mathcal{X}$ and a reconstructed tensor $\mathcal{Y}$, which is difficult to be minimized.

The inexact alternating direction method (IADM) [42] is then embedded into the MM algorithm, which can solve the optimization problems (7) subject to certain inexactness criteria by breaking it into small subproblems which are easier to handle. With the initial tuple $(\mathcal{X}^{[0]},\{(A^{(n)})^{[0]}\}_{n=1}^4,\mathcal{Y}^{[0]})$, the resulting iteration scheme is

$$\begin{cases}\mathcal{X}^{[t+1]}\approx\arg\min_{\mathcal{X}} L(\mathcal{X},\{(A^{(n)})^{[t]}\}_{n=1}^4,\mathcal{Y}^{[t]}),\\ (A^{(n)})^{[t+1]}\approx\arg\min_{A^{(n)}} L(\mathcal{X}^{[t+1]},\{(A^{(k)})^{[t+1]}\}_{k<n},A^{(n)},\\ \qquad\qquad \{(A^{(k)})^{[t]}\}_{k>n},\mathcal{Y}^{[t]}),n=1,\ldots,4;\\ \mathcal{Y}^{[t+1]}\approx\arg\min_{\Omega(\mathcal{Y})=\Omega(\mathcal{Y}_0)} L(\mathcal{X}^{[t+1]},\{(A^{(n)})^{[t+1]}\}_{n=1}^4,\mathcal{Y}).\end{cases} \tag{8}$$

The subproblems in (8) will be carefully treated as follows.

3.2.1. Optimization of $\mathcal{X}$

Followed by Yang et al. [40], the update of $\mathcal{X}$ is obtained by solving the following optimization problem

$$\min_{\mathcal{X}} \quad \langle\mathcal{X},\mathcal{D}^{[t]}*\mathcal{X}\rangle+\mu\left\|\mathcal{Y}-\mathcal{X}\prod_{n=1}^{4}\times_n A^{(n)}\right\|_F^2, \tag{9}$$

where $\mathcal{D}^{[t]}$ is a tensor of the same size of $\mathcal{X}$ with its $(i_1,i_2,\ldots,i_4)$th element given by

$$\mathcal{D}^{[t]}_{i_1,\ldots,i_n,\cdots,i_4}=\sum_{n=1}^{4}\left(\left\|\mathcal{X}^{[t]}_{(n,i_n)}\right\|_F^2+\epsilon\right)^{-1}. \tag{10}$$

The details for a majorization function of $L(\mathcal{X},\{A^{(n)}\}_{n=1}^4,\mathcal{Y})$ in (7) will be found in Appendix A.

Let $x\triangleq\text{vec}(\mathcal{X})$, $D\triangleq\text{diag}(\text{vec}(\mathcal{D}^{[t]}))$, $y\triangleq\text{vec}(\mathcal{Y})$, and $H\triangleq(\bigotimes_n A^{(n)})$. The above optimization can be expressed as

$$\min_{x} \quad \mu\|y-Hx\|_2^2+x^TDx, \tag{11}$$

which has a unique optimal solution of the form

$$x=(H^TH+\mu D)^{-1}H^Ty. \tag{12}$$

It is expensive to get x directly from (12) since the computation complexity for the inverse of the matrix $H^TH+\mu D$ is $O\left(\prod_{n=1}^4 I_n^3\right)$. To accelerate the computation, an iterative algorithm called the over-relaxed MFISTA approach (e.g. [43]) is employed which not only guarantees the monotonically decreasing in the objective function, but also admits a variable stepsize in a broader range with the convergence rate $O(1/k^2)$. For notational convenience, denote

$$f(x)=\mu\|y-Hx\|_2^2,\ \text{and}\ g(x)=x^TDx.$$

Direct calculation leads to

$$\nabla f(x)=2\mu(H^THx-H^Ty),$$

or a more efficient tensor version as

$$\nabla f(\mathcal{X})=2\mu\left(\mathcal{X}\prod_{n=1}^{4}\times_n A^{(n)}-\mathcal{Y}\right)\prod_{n=1}^{4}\times_n A^{(n)T}. \tag{13}$$

Additionally, for any given positive scalar β, the proximal operator $prox_{\beta g}(x)$ has the following closed form

$$prox_{\beta g}(x) := \arg\min_{z}\left\{g(z)+\frac{1}{2\beta}\|z-x\|_2^2\right\} = (2\beta D+I)^{-1}x, \tag{14}$$

where I is the identity matrix with the same size of D. Since D is a diagonal matrix, the inverse of $(2\beta D+I)$ can be easily obtained. Note that ∇f is Lipschitz continuous with the Lipschitz constant $L(f)$ being

$$L(f)=\lambda_{\max}(2\mu H^T H)=2\mu\prod_{n=1}^{4}\lambda_{\max}(A^{(n)T}A^{(n)}), \tag{15}$$

where $\lambda_{\max}(X)$ stands for the largest eigenvalue of the matrix X, and details of the Eq. (15) are shown in Appendix B.

The procedure to update $\mathcal{X}$ is illustrated by Algorithm 1.

Algorithm 1 Solving (9) by the over-relaxed MFISTA.

Input: two tensors $\mathcal{Y}$ and $\mathcal{X}$; Factor matrices $\{A^{(n)}\}_{n=1}^4$; Parameters δ, $k_{\max}$;
Output: $\mathcal{X}$;

Step 0 Compute D by (10) and $L(f)$ by (15);
Step 1 select β from $(0, (2-\delta)/L(f)]$ with some $\delta\in(0,2)$;
Step 2 Set $x^{[0]}$=vec($\mathcal{X}$), $w^{[1]}=x^{[0]}$, $\eta^{[1]}$=1;
Step 3 For k=1 to $k_{\max}$ do
- Calculate $\nabla f(\mathcal{X})$ using (13);
- $z^{[k]}=prox_{\beta g}(w^{[k]}-\beta\nabla f(w^{[k]}))$ or using (14);
- $x^{[k]}$=arg $\min\{F(z)\mid z\in\{z^{[k]},x^{[k-1]}\}\}$;
- $\eta^{[k+1]}=\frac{1+\sqrt{1+4(\eta^{[k]})^2}}{2}$;
- $w^{[k+1]}=x^{[k]}+\frac{\eta^{[k]}}{\eta^{[k+1]}}(z^{[k]}-x^{[k]})+\frac{\eta^{[k]}-1}{\eta^{[k+1]}}(x^{[k]}-x^{[k-1]})+\frac{\eta^{[k]}}{\eta^{[k+1]}}(1-\delta)(w^{[k]}-z^{[k]})$;

End for
Step 4 Set $\mathcal{X}$=tensor($x^{[k_{\max}]}$)

3.2.2. Optimization of $A^{(n)}$'s

For $\{n_1,n_2,n_3,n_4\}\in\{1,2,3,4\}$, any given $\mathcal{X}$, $\{A^{(n_j)}\}_{n_j\neq n_i,n_j=1}^4$, $\mathcal{Y}$, the update of $A^{(n_i)}$ is obtained from

$$\widehat{A^{(n_i)}} \approx \arg\min_{A^{(n_i)}} L(\mathcal{X},\{A^{(n_i)}\}_{n_i=1}^4,\mathcal{Y}) =: \arg\min_{A^{(n)}}\{f_1(A^{(n_i)})+f_2(A^{(n_i)})\}, \tag{16}$$

where $f_1(A^{(n_i)})=\gamma\lambda_n\|A^{(n_i)}\|_*$ is a closed convex but not differentiable function, and

$$f_2(A^{(n_i)})=\mu tr(A^{(n_i)T}A^{(n_i)}\Phi_{n_i}\Phi_{n_i}^T-2A^{(n_i)T}Y_{(n_i)}\Phi_{n_i}^T)$$

with $\Phi_{n_i}=(\mathcal{X}\prod_{k\neq n_i}\times_k A^{(k)})_{(n_i)}$, is a convex quadratic function. To get a closed form approximation of $\widehat{A^{(n_i)}}$, a majorization technique for $f_2(A^{(n_i)})$ based upon its first-order Taylor expansion at the current $(A^{(n_i)})^{[t]}$ is utilized, and $\widehat{A^{(n_i)}}$ is then obtained via

$$\widehat{A^{(n_i)}} \approx \arg\min_{A^{(n_i)}} f_1(A^{(n_i)})+f_2((A^{(n_i)})^{[t]})+\langle\nabla f_2((A^{(n_i)})^{[t]}), A^{(n_i)}-(A^{(n_i)})^{[t]}\rangle+\frac{\zeta_n}{2}\|A^{(n_i)}-(A^{(n_i)})^{[t]}\|_F^2 = \Theta_{\gamma_{n_i}}\left((A^{(n_i)})^{[t]}-\frac{1}{\zeta_{n_i}}\nabla f_2((A^{(n_i)})^{[t]})\right) \tag{17}$$

with $\nabla f_2((A^{(n_i)})^{[t]})=2\mu(A^{(n_i)}\Phi_{n_i}-Y_{(n_i)})\Phi_{n_i}^T$, where $\gamma_{n_i}=\frac{\gamma}{\zeta_{n_i}}\lambda_{n_i}$, $\zeta_{n_i}=2\mu\|\Phi_{n_i}\|_2^2$, $\Theta_\alpha(Z)=US_\alpha(\Sigma)V^T$ for any matrix Z with its singular value decomposition (SVD) $Z=U\Sigma V^T$, and $S_\alpha(\Sigma_{ij})=sign(\Sigma_{ij})*\max(0,|\Sigma_{ij}|-\alpha)$ is the soft-thresholding operator.

3.2.3. Optimization of $\mathcal{Y}$

Given $\mathcal{X}$, $\{A^{(n)}\}_{n=1}^4$, the update $\widehat{\mathcal{Y}}$ can be easily obtained by the projection property in the following way:

$$\begin{cases}\Omega(\widehat{\mathcal{Y}})=\Omega(\mathcal{Y}_0),\\ \overline{\Omega}(\widehat{\mathcal{Y}})=\overline{\Omega}\left(\mathcal{X}\prod_{n=1}^{4}\times_n A^{(n)}\right).\end{cases} \tag{18}$$

Now, we are in a position to establish the proximal version of MM framework with IADM in the following Algorithm 2.

Algorithm 2 Solving (7) by the MM algorithm with IADM.

Input: A tensor $\mathcal{Y}_0\in\mathbb{R}^{I_1\times I_2\ldots\times I_4}$; Parameters λ_n, γ, μ, $t_{\max}$;
Output: Factor matrices $\{A^{(n)}\}_{n=1}^4$;

Step 0 Initialization: choose $\{(A^{(n)})^{[0]}\}_{n=1}^4$, $\mathcal{X}^{[0]}$, $\mathcal{Y}^{[0]}=\mathcal{Y}_0$ and set $t=0$;
Step 1 Update $\mathcal{X}$ by Algorithm 1;
Step 2 Update $\{A^{(n)}\}_{n=1}^4$ by (17);
Step 3 Update $\mathcal{Y}$ by (18);
Step 4 (Rank Reduction Strategy) Remove negligible rows of each mode unfolding of $\mathcal{X}$ and the corresponding columns of $\{A^{(n)}\}_{n=1}^4$ according to the rank reduction strategy with a given threshold θ (details in next subsection);
Step 4 $t=t+1$; while some stop criteria are not satisfied, go to Step 1.

In the above algorithm, the iteration process will be terminated once the iteration number reaches some prescribed $t_{\max}$ or

$$\left\|\Omega\left(\mathcal{Y}^{[t+1]}-\mathcal{X}^{[t]}\prod_{n=1}^{4}\times_n(A^{(n)})^{[t]}\right)\right\|_F^2/\|\mathcal{Y}_0\|_F^2<\eta, \tag{19}$$

with some sufficiently small $\eta>0$.

3.2.4. Rank reduction strategy (RRS)

After achieving $\mathcal{X}$, $\{A^{(n)}\}$ and $\mathcal{Y}$ at each iteration, negligible columns of $\{A^{(n)}\}$ may exist, which is according to rows of each mode unfolding of $\mathcal{X}$. The strict definition of unnecessary rows of $\mathcal{X}_{(n,i)}$ is shown

$$\mathcal{H}^{(n)}:=\left\{i\,\middle|\,\left\|\mathcal{X}_{(n,i)}\right\|_F^2=0;\ n=1,\ldots,4,\ \forall i\right\}. \tag{20}$$

If $j\in\mathcal{H}^{(n)}$, jth column of $A^{(n)}$ can be clearly ignored. However, its criterion is difficult to remove the negligible components. Thus, a more relaxed criterion is utilized by

$$\mathcal{M}^{(n)}:=\left\{i\,\middle|\,1-\frac{\|\mathcal{X}_{(n,i)}\|_F^2}{\max_i(\|\mathcal{X}_{(n,i)}\|_F^2)}\geq\theta;\ n=1,\ldots,4,\ \forall i\right\}, \tag{21}$$

where $\theta\in[0.7,\ 1]$ is a threshold (e.g., θ=0.9980), which means a "big" gap between $\|\mathcal{X}_{(n,i)}\|_F^2$ and $\max_i(\|\mathcal{X}_{(n,i)}\|_F^2)$. Meanwhile, strong interactions among factor matrices and the core tensor will be retained naturally. Finally, negligible complements are removed by

$$\begin{cases}\mathcal{X}_{(n)}\leftarrow\mathcal{X}_{(n)}(\mathcal{M}_\perp^{(n)},:),\\ A^{(n)}\leftarrow A^{(n)}(:,\mathcal{M}_\perp^{(n)}),\ n=1,\ldots,4;\end{cases} \tag{22}$$

where $\mathcal{M}_\perp^{(n)}$ is the complement operator of $\mathcal{M}^{(n)}$.

4. Experimental evaluation

To evaluate the effectiveness of our proposed approach (FERL-rTC), we will compare its performance in terms of different experimental protocols and other methods of the state-of-the-art over

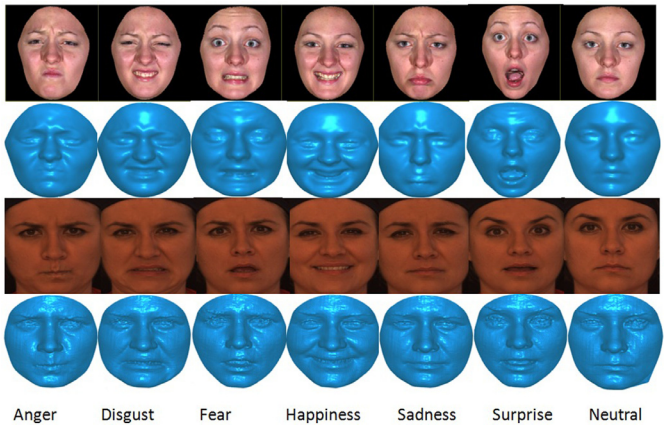

Fig. 2. Seven basic expressions with face images and facial models in BU-3DFE and Bosphorus databases, respectively.

Fig. 3. Visualization of the nine types of 2D maps and 2D texture information of two textured 3D face scans (subject bs000 on Bosphorus database in the first line and subject M0031 with 4 levels of expression intensity on BU-3DFE dataset in rest lines) with happiness expression. Each line shows: the depth (the geometry maps), the three normal maps (x, y, and z), curvature maps (curvature and mean curvature), 2D texture information (components R, G, B) (from the second row to the fifth row: level 4 to level 1).

two 3D face databases including BU-3DFE [44] and Bosphorus [45]. The details will be focused on BU-3DFE database. We also validate FERLrTC on synthetic data of the third-order and fourth-order tensor. Finally, we will discuss the following five issues: the effectiveness of the 4D tensor model based on feature-level fusion for 2D+3D FER, the selection of feature descriptors, complexity and convergence analysis, the effectiveness of the rank reduction strategy.

4.1. Implementation details

4.1.1. Databases and preprocessing

BU-3DFE database: As a database with the prototypical expressions of seven basic emotions (i.e., anger, disgust, fear, happiness, sadness, surprise and neutral), the BU-3DFE database [44] has become the actual test bed where FER researchers evaluate their approaches. It contains 2500 3D face models of 100 subjects with 56 females and 44 males, aging from 18 to 70, with a variety of ethnic/racial ancestries, including East-Asian, White, Indian Middle-east Asian, Hispanic Latino, and Black. For every subject, there are 25 scans of which one is the neutral expression and the rest are six prototypical expressions (except neutral) of four levels of intensity. The seven basic expressions consist of anger (AN), disgust (DI), fear (FE), happiness (HA), sadness (SA), surprise (SU) and neutral (see Fig. 2).

Bosphorus 3D face database: The Bosphorus database [45] has been widely applied in 3D human face processing tasks including facial action unit detection, expression recognition, face recognition under adverse conditions, etc. It is composed of 105 subjects and 4666 pairs of 3D face models and 2D face images in various poses, expressions, and occlusion conditions. Different from BU-3DFE database, Bosphorus database has not provided facial expression with intensity information and has only 65 subjects accomplishing six prototypical expressions.

Preprocessing: The preprocessing of the data in both BU-3DFE [44] and Bosphorus [45] databases are similar and includes: pose correction based on the Iterative Closest Point (ICP) algorithm [46], nose detection, face cropping, resample and projection procedures with cubic interpolation for 3D face normalization. According to *x*, *y*, and *z* coordinates obtained, geometry map I_g, three normal component maps I_n^x, I_n^y and I_n^z, and curature maps (i.e., curvature I_c and mean curvature I_{mc}) can be achieved by the methods introduced in [2,29]. The 3-channel 2D texture information I_t^r, I_t^g and I_t^b of BU-3DFE database are obtained by projecting 3D texture images with linear interpolation. These generated features are used with LBP (Local Binary Pattern) descriptor [47] which has been widely employed in both 2D and 3D FER. Samples of preprocessed facial attribute maps and 2D texture information of BU-3DFE and Bosphorus databases are shown in Fig. 3.

4.1.2. Algorithm initialization

To alleviate the sensitivity to the algorithm's parameters, the insensitive parameters δ and β are simply set to be 0.1, and $(2-\delta)/L(f)$, respectively. λ_n $(n=1,2,3,4)$ is set to be $\frac{\prod_{k\neq n,k=1}^{4} a_k * \|A^{(k)}\|_*}{\sum_{n=1}^{4}\prod_{k\neq n,k=1}^{4} a_k * \|A^{(k)}\|_*}$ $(a_1=0.1, a_2=0.1, a_3=80, a_4=0.01)$ to better characterize the similarities of samples. The choices of μ and γ are more crucial than the others, and they depend on the data missing ratio and the value of facial expression data. A stable recovery performance can be obtained when μ and γ are set to be $\frac{1}{w_1\|\mathcal{Y}_0\|_F^2}$ and $w_2\frac{\mu\|\Phi_{(4)}\|_2^2}{\lambda_4}$, respectively. w_1 is set to be 1e-12 according to the size of the facial expression data, and w_2 is set to be 5e−2 on the basis of $\gamma_n=\frac{\gamma}{\zeta_n}\lambda_n$ to avoid severe truncation. $\mathcal{Y}_0$ is set to be $\mathcal{Y}_{\text{Training}}$, $\{(A^{(n)})^{[0]}\}_{n=1}^4$ are obtained from the high order singular value decomposition (HOSVD) of $\mathcal{Y}_0$ ([48]), and $\mathcal{X}^{[0]}$ is set to be $\mathcal{Y}_0\times_1((A^{(1)})^{[0]})^\top\times_2((A^{(2)})^{[0]})^\top\cdots\times_4((A^{(4)})^{[0]})^\top$. The maximal iteration numbers $k_{\max}$ and $t_{\max}$ are set to be 5 and 100, respectively, and the accuracy parameter η in Algorithm 2 is set to be 1e-4. θ is set to be 0.99875 when the rank reduction strategy is implemented.

4.1.3. Tensor reconstruction and classification prediction

Given a $\mathcal{Y}_{\text{Training}}$ and a $\mathcal{Y}_{\text{Testing}}$ as the training and the testing, respectively, we can get the corresponding estimated factor matrices $\{A^{(n)}\}_{n=1}^4$ by Algorithm 2. Then the reconstruction tensors $\mathcal{Y}_{\text{Tr}}$ and $\mathcal{Y}_{\text{Te}}$ are obtained by $\mathcal{Y}_{\text{Training}}\times_1 A^{(1)T}\times_2 A^{(2)T}\times_3 A^{(3)T}$ and $\mathcal{Y}_{\text{Testing}}\times_1 A^{(1)T}\times_2 A^{(2)T}\times_3 A^{(3)T}$, respectively.

Set X_{Training} be the mode-4 unfolding matrix of $\mathcal{Y}_{\text{Tr}}$ and X_{Testing} be that of $\mathcal{Y}_{\text{Te}}$. These two matrices, together with the labels of the corresponding samples are sent to the Multi-Class-SVM classifier based on linear SVM with default parameter of C for classification prediction.

4.2. Evaluation and comparison on BU-3DFE database

4.2.1. Experimental protocol

Three experimental protocols, termed as Setup I, II, and III, are used in this paper. Among them, the two highest levels of intensity are considered in Setup I and II, while all four levels of intensity are utilized in Setup III. The details of protocols are as below:

Table 1
Average confusion matrix for feature-level fusion FER on BU-3DFE database with Setup I, II and III.

%	AN	DI	FE	HA	SA	SU
AN	**80.92**	4.58	3.83	0.58	10.09	0.00
DI	5.58	**78.67**	7.17	2.67	1.83	4.08
FE	4.50	5.91	**70.75**	10.00	5.17	3.67
HA	0.00	1.75	5.67	**92.25**	0.00	0.33
SA	11.75	2.50	6.67	0.17	**78.91**	0.00
SU	0.25	1.00	2.00	0.92	0.00	**95.83**
Setup I 82.89%						
%	AN	DI	FE	HA	SA	SU
AN	**76.75**	6.75	2.67	0.75	13.08	0.00
DI	10.92	**76.28**	6.58	2.92	2.33	0.97
FE	2.42	8.63	**69.12**	6.75	9.33	3.75
HA	1.58	0.30	3.60	**93.65**	0.17	0.70
SA	15.65	4.42	2.83	1.05	**76.05**	0.00
SU	0.85	0.50	2.75	2.27	0.00	**93.63**
Setup II 80.91%						
%	AN	DI	FE	HA	SA	SU
AN	**76.55**	6.70	4.50	0.00	12.25	0.00
DI	9.20	**79.25**	7.20	0.15	2.05	2.15
FE	3.05	9.25	**67.00**	9.75	6.85	4.10
HA	0.75	1.90	7.80	**89.25**	0.00	0.30
SA	19.75	3.00	6.35	0.65	**70.25**	0.00
SU	0.20	1.70	5.65	1.00	0.00	**91.45**
Setup III 78.96%						

i) Setup I fixes 60 subjects in all the experiments, while Setup II selects randomly 60% of 100 subjects in each round. The 10-fold cross-validation scheme is adopted in which 60 subjects selected are randomly divided into 10 subsets, and one subset (6 subjects with 72 scans) is retained as the testing and the other 9 subsets (648 scans of 54 subjects) as the training for each time; ii) Setup III utilizes the 10-fold cross-validation scheme which 100 subjects are randomly divided into 10 subsets, and 9 subsets are utilized for training (i.e., 12960 attribute maps of 90 subjects) and the remaining is used for testing (i.e., 1,440 attribute maps of 10 subjects). The experiment is repeated for 10 times so that every subset is treated exactly once as the testing and that the training and the testing own no overlap. Then the results averaged from 10 splits are the final estimation. Experiments in Setup I, II and III are repeated 100 times for the average performance, and the linear SVM is used as the classifier in all experiments. The experimental analysis below is based on the results generated in Setup I, II and III.

In addition, to facilitate comparison with some existing approaches, Setup V is added, which runs less than 20 times.

4.2.2. Results on BU-3DFE database

Comparison for different experimental protocols: Table 1 shows the average confusion matrices for feature-level fusion by Setup I, II and III. It is easy to find from Table 1 that happiness and surprise are the two expressions easiest to be recognized due to their high facial deformations, whereas sadness and fear are two which are more difficult to be recognized except in Setup I. Among the three strategies, Setup I achieves the best result with recognition accuracy only 1.98% higher than Setup II, and 3.93% higher than Setup III. For sadness recognition, Setup I obtains better result compared with Setup II and III, and even shows some improvement compared with those in [17,29,49]. Disgust expression of Setup III achieves higher recognition accuracy compared with those of Setup I and II. From Table 1, we can learn that facial expressions with lower levels of expression intensity (i.e., 1-level and 2-level) are actually much more difficult to be recognized than those with the higher levels.

Comparison with different classifiers: It is known that there are various classifiers such as SVM, k-Nearest Neighbor (k-NN), Bayesian Belief Net (BBN), Neural Networks (NN), multi-boosting, Maximal Likelihood (ML), random forest, collaborative representation based classification (CRC) and Sparse representation based classification (SRC), among which the SVM, k-NN, CRC and random forest are all capable of high-dimensional data and multi-class classification. To show the superiority of utilizing linear SVM classifier, experiments are implemented on BU-3DFE database with Setup I. The parameters are set in the following: (i) the parameter C is taken the default value 1 in SVM; (ii) $k = 1$ is set in k-NN; (iii) λ and the eigenfaces dimension are set to be 1e-3 and 200, respectively for a best performance in CRC; iv) the number of trees are taken the default value 500, and the number of variables used for the binary tree in the nodes is set to be 200 in random forest. Accuracy results as reported in Table 2 shows that the linear SVM is generally regarded as the best candidate classifier in expression prediction.

Table 2
Comparison of different classifiers on BU-3DFE database with Setup I.

Classifier	k-Nearest neighbor	CRC	Random forest	Linear SVM
Accuracy(%)	78.91	80.37	80.22	**82.89**

Comparison with tucker decomposition algorithms in other applications: Our proposed approach (FERLrTC) is compared with IRTD [40], APG_NTDC [50], WTucker [51] and KBR_TC [11] on BU-3DFE database. The IRTD proposes that the sparsity over the core tensor $\mathcal{X}$ is replaced with a group-based log-sum penalty function for Tucker decomposition of incomplete tensors and a Frobenius norm is imposed on factor matrices $\{A^{(n)}\}$ to avoid a trivial solution $\{\mathcal{X} \to 0, \{A^{(n)}\} \to \infty\}$, the APG_NTDC utilizes alternating proximal gradient (APG) method to decompose a tensor into a core tensor and several factors with sparsity and non-negativity constraints. The WTucker proposes a Tucker factorization method based on predefined multilinear rank and applies it to low-n-rank completion, and the KBR_TC proposes a tensor sparsity measure based on Kronecker-basis-representation (KBR) for tensor recovery, in which the number of rank-1 Kronecker bases are utilized for representing the tensor. To speed up the convergence processing, KBR_TC adopts the rank-increasing scheme [52] and IRTD uses the rank reduction strategy as FERLrTC does. It should be noted that the APG_NTDC requires to predefine the multilinear rank, and the KBR_TC provides a smaller estimate of multilinear rank, and the multilinear rank of the WTucker needs to be over-estimated.

All the parameters utilized in these three algorithms are adjusted carefully according to the suggestions in the related literatures for achieving their best performances: (i) IRTD, we set $\delta = 0.1$, $\beta = (2-\delta)/L(f)$, $\lambda_1 = 0.1$, $\lambda_2 = 1$ and $\gamma = 1.25\text{e}{-3}$; (ii) APG_NTDC, λ_n (n= 1, 2, 3, 4) and λ_c are all set to be 0.5, the predefined multilinear rank is set to be (15, 11, 8, 12); (iii) WTucker, the multilinear rank to be (30, 22, 9,24); (iv) KBR_TC, we set $\lambda = 10$, $c = 1e-3$, $v = 0.1$, $\rho = 1.05$ and $\mu = 250$, the provided multilinear rank is set to be (12, 12, 9, 12). The comparisons on recognition accuracy (RA), relative error(RE), the rank variations (RV) of factor matrices and feature reconstruction (FR) are shown as below.

(1) *RA:* The comparison results on average recognition accuracy (%) in Setup I are reported in Fig. 4(a), from which one can see that the FERLrTC achieves the best performance on average recognition accuracy, while APG_NTDC gains a relatively worse performance. The results obviously indicates our proposed tensor dimensionality reduction approach based on Tucker decomposition can extract the more effective low dimensional features from the generated 4D tensor for multi modal FER, and the low-rank tensor completion model for multimodal FER are more effective than other methods.

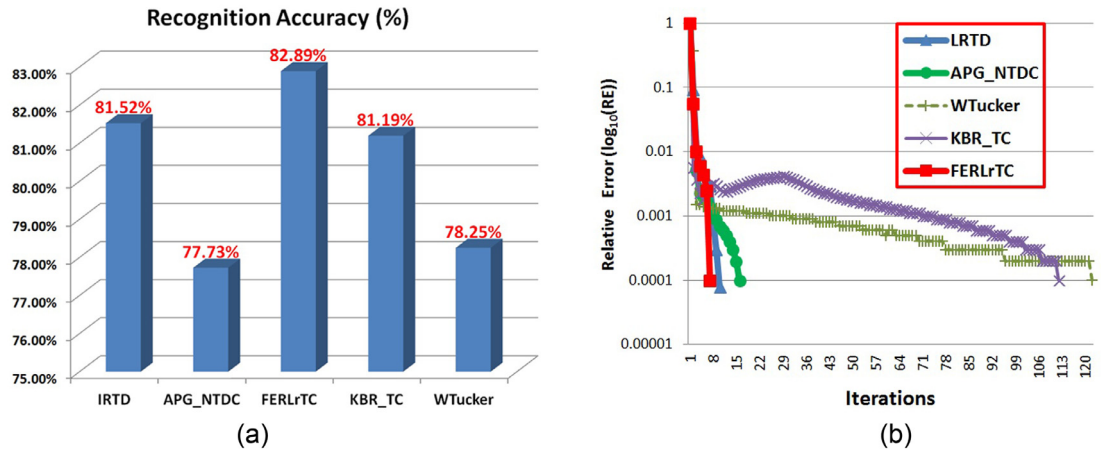

Fig. 4. Comparison of average recognition rate(%) and convergence behavior with IRTD, APG_NTDC, WTucker and KBR_TC on BU-3DFE database with Setup I.

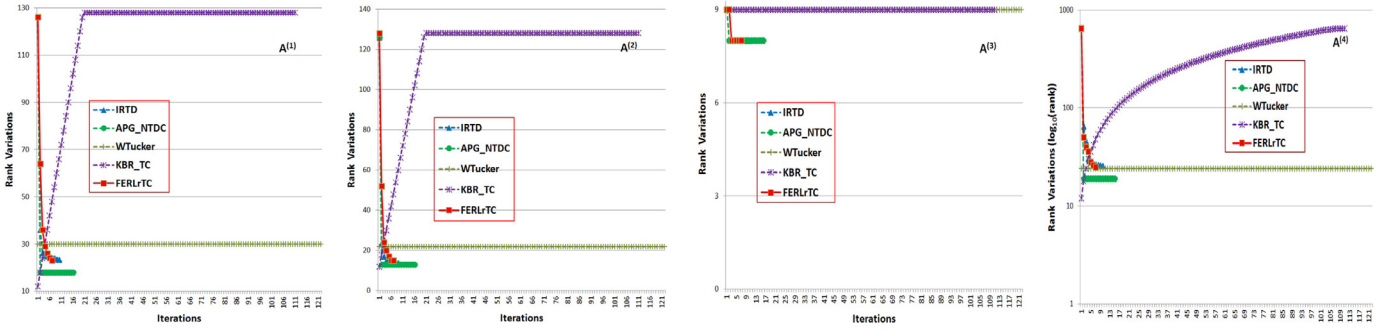

Fig. 5. The comparison results of the rank variations of factor matrices ($A^{(n)}$, n= 1, 2, 3, 4) with IRTD, APG_NTDC, WTucker and KBR_TC on BU-3DFE database by Setup I.

(2) *RE:* RE $= \|\widehat{\mathcal{Y}}^t - \widehat{\mathcal{Y}}^{t-1}\|_F / \|\mathcal{Y}_0\|_F$ is often utilized to verify convergence behavior. Fig. 4(b) shows that RE can converge fast in a number of iterations. The comparison results manifest our approach has better convergence behavior, which further indicates the rank reduction strategy is more efficient compared with the rank-increasing scheme adopted by KBR_TC.

(3) *RV:* Fig. 5 reports the comparison results of rank variations of factor matrices. It is noted that the spatial redundancy information of the 4D tensor data is reflected in the spatial structure of the generated factor matrices by Tucker decomposition. Hence, our aim is to obtain the low-rankness of factor matrices for projection and then to achieve the goal of the 4D tensor dimensionality reduction. From Fig. 5, we can know the rank variations of factor matrices are stable: (i) $A^{(4)}$ that represents the number of samples varies the fastest and also verifies the high similarities among samples; (ii) $A^{(3)}$ that indicates the number of features only changes a little; (iii) $A^{(1)}$ and $A^{(2)}$ that show the size of 2D features vary slower than $A^{(4)}$. It is clear that the rank of factor matrices will no longer change after several iterations. The comparison results indicate that rank variations of factor matrices of our proposed approach are relatively slower than IRTD, which is because the desired low-rank structure among samples with the high similarities is characterized in terms of the low-rankness of the involved factor matrices and the structured sparsity of the involved core tensor, effectively avoiding the overpruning of the factor matrices. On the other hand, KBR_TC based on the rank-increasing scheme shows the bad performance in approximating the multilinear rank.

(4) *FR:* Fig. 6 shows some examples of original features by 2D maps, features with LBP descriptor (here after LBP feature for short), LBP feature with 70% sampling ratio (SR) selected randomly, and their reconstruction features by APG_NTDC, IRTD and FERLrTC, respectively. And it is noted that WTucker and KBR_TC could not carry out successfully face reconstruction. Meanwhile we can easily observe that the LBP features of our proposed approach FERLrTC could be reconstructed by over 70% SR, and the reasons lie in two aspects: one is that nine kinds of features without LBP descriptor contain some NaN values (i.e., indeterminate values) which are formed when the 3D face scans map into 2D planes and set to be zero, the other is that nine kinds of features with LBP descriptor contains averagely less than 65% useful formation

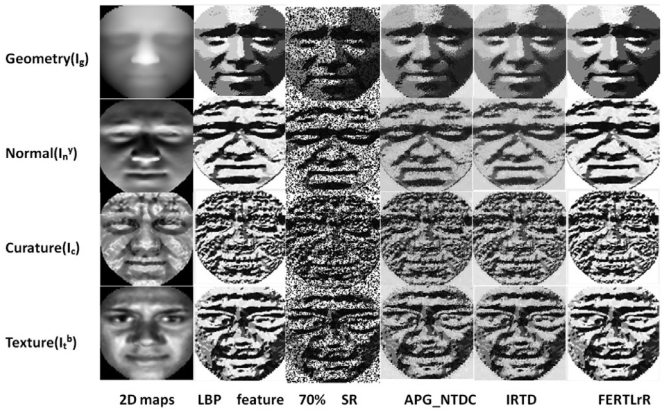

Fig. 6. The comparison results for LBP feature reconstruction with 70% SR on BU-3DFE database with Setup I (the subject F0003 with angry expression of 4-level intensity).

Table 3
Comparisons of the significant test with IRTD, KBR_TC, WTucker and APG_NTDC, respectively.

	IRTD	KBR_TC	WTucker	APG_NTDC
P	1.85E−08	1.29E−08	3.15E−54	2.01E−67
H	1	1	1	1

which are larger than zero. In short, our proposed approach obtains better reconstruction result, which indicates that our proposed approach can make use of more information contained in higher dimensional tensors.

In addition, in order to show the significant difference between our proposed method and other methods, we use Wilcoxon rank-sum method [61] to test the statistical significance because the five sets of recognition accuracy data generated by these five methods are non-normal distribution. Assuming that there is an equal median between our method and other methods, i.e., the null hypothesis, and the significance level is set to be 5%, we use the rank-sum function in MATLAB to carry out the significant tests between our proposed approach and other four methods respectively. Table 3 shows the comparison results. From this table, we can observe that the values of P are much less than 5% and all the values of H are all ones, in which the value of P is the probability of achieving a result equal to or more extreme than what is observed actually when the null hypothesis is true, and the results $H = 0$ and H=1 illustrate an acceptance and a rejection of the null hypothesis under the 5% significance level, respectively. As we know, both P<0.05 and H=1 indicate that the null hypothesis of equal medians is rejected under the 5% significance level. Meanwhile we can observe that the lower the recognition accuracy of the compared method, the smaller the generated value of P. Thus, there are significant differences between our proposed approach and other methods.

Comparison with other methods: In order to comprehensively evaluate the validity of the proposed method (FERLrTC), we compare it with some state-of-the-art approaches in four aspects including data modality, expression features, expression classifiers and recognition accuracy shown in Table 4(a). The result indicates that our proposed method (FERLrTC) obtains the higher accuracies of 82.89%, 80.91%, 78.96%, and 95.28% compared with all state-of-the-art approaches by Setups I, II, III and V, respectively. From Table 4(a), it is observed that the recognition accuracies of [21,53,54] obtained by utilizing the unstable experiment protocols (i.e., 10-fold or 20-fold cross-validation) decline more than 20% significantly than those of utilizing a more stable experimental protocol (i.e., Setup I). Therefore, we can see that our proposed method can obtain the stably good performance under different setups.

Although our proposed method (FERTLrR) based on Tucker decomposition outperforms the state-of-the-art in Table 4(a) by using Setups I, II and III, there are still some approaches which surpass our method in Table 4(b). As we know, the state-of–art algorithms in Table 4(b) could produce high accuracy with feature vectorization and concatenation because of larger samples and high complexity to construct the networks, such as [2,10,17,60], or facial landmark localization [59]. In addition, compared with the approaches in Table 4(b), our method needs fewer parameters, less complexity, smaller samples and no key landmarks. In spite of a certain gap between our method and the approaches [2] in terms of recognition accuracy, how to improve the recognition accuracy is one of our future directions.

Model-based methods of the state-of-the-art are shown in Table 5 for running only once with 10-fold cross-validation method. The data include single modality (3D) or multi-modality (2D+3D). From Table 5, we can learn that our feature-based method can outperform the model-based methods except for [3].

Table 4
Comparison of data modality, expression features, classifiers, and accuracies with the state-of-the-art on BU-3DFE database (T means the repeated times).

(a)

Method	Data	Feature	Classifier	Setup I	Setup II	Setup III	Setup V
Wang et al. [53]	3D	curvatures/histogram	LDA	61.79	-	-	83.60(20T)
Soyel et al. [54]	3D	points/distance	NN	67.52	-	-	91.30(10T)
Tang et al. [21]	3D	points/distance	LDA	74.51	-	-	95.10(10T)
Gong et al. [32]	3D	depth/PAC	SVM	76.22	-	-	-
Berretti et al. [55]	3D	depth/SIFT	SVM	-	77.54	-	-
Li et al. [28]	3D	normals/LBP	MKL	-	80.14	78.50	-
Zeng et al. [56]	3D	curvatures/LBP	SRC	-	70.93	-	-
Lemaire et al. [29]	3D	mean curvature/HOG	SVM	-	76.61	-	-
Yurtkan et al. [57]	3D	points/histogram	SVM	-	-	-	88.28(8T)
Yurtkan et al. [22]	3D	points/histogram	SVM	-	-	-	90.8(10T)
Azazi et al. [58]	3D	landmark	RBF-SVM	-	79.36	-	-
Fu et al. [18]	3D	normals,curvature	NN	-	-	-	85.802(10T)
Zhao et al. [30]	2D+3D	intensity,coordinates, shape index/LBP	BBN	-	-	-	82.30(10T)
FERLrTC	2D+3D	depth,normals,curvatures,textures/LBP	SVM	**82.89**	**80.91**	**78.96**	95.28(10T)

(b)

Method	Data	Feature	Classifier	Setup I (%)	Setup II (%)	Setup III (%)	Setup V (%)
Yang et al. [17]	3D	depth, normals, shape index/scattering	SVM	84.80	82.73	-	-
Li et al. [59]	2D+3D	meshHOG/SIFT, meshHOS/HSOG	SVM	86.32	-	80.42	-
Li et al. [2]	2D+3D	32-D deep feature 6-D deep feature	SVM Softmax	86.86 86.20	-	81.04 81.33	-
Chen et al. [60]	3D	deep feature	Softmax	86.67	85.96	-	-
Yao et al. [10]	2D+3D	NOM Static+Dynamic SIM	MKL	-	90.12	-	-
	Static+Dynamic	Static+Dynamic 3D Static+Dynamic 2D Static+Dynamic					

Table 5
Model-based methods of the state-of-the-art on BU-3DFE database.

Method	Data	Methodology	Setup V(%)
Soyel et al. [62]	3D	Neural network	87.9(10T)
Tang et al. [63]	3D	SVM	94.7(10T)
Tang et al. [21]	3D	AdaBoost	87.1(10T)
Mpiperis et al. [31]	3D	Bilinear model	90.5(10T)
Zhao et al. [30]	2D+3D	BBN+SFAM	87.2(10T)
Zhen et al. [3]	3D	Logistic Regression	96.4(10T)

Table 6
Average confusion matrix for feature-level fusion FER on Bosphorus database with Setup IV.

%	AN	DI	FE	HA	SA	SU
AN	**77.37**	6.23	3.23	0.13	12.91	0.13
DI	11.43	**67.03**	5.7	5.23	7.87	2.74
FE	7.63	4.87	**63.83**	1.53	1.61	20.53
HA	0	3.57	1.73	**92.97**	0	1.73
SA	15.03	11.8	5.67	0	**65.97**	1.53
SU	1.73	3.97	5.67	0.23	0	**88.4**
Setup IV	**75.93**					

In [3], it requires to annotate manually landmarks around some key region and to a dense correspondence among face scans for achieving a better recognition accuracy. However, our proposed method is not sensitive to topological changes and don't require a dense correspondence among face scans and the artificial landmark annotation, which are often needed in practice for model-based methods.

4.3. Evaluation and comparison on bosphorus database

4.3.1. Experimental protocol

Setup IV also utilizes the 10-fold cross-validation scheme. 60 subjects are randomly selected and divided into 10 subsets, and for each subset, 2916 features including 2D maps and 2D texture information of 54 subjects are utilized for training and the remaining 324 features of 6 subjects are used for testing. Thus, the experiment needs to be repeated for 10 times, and then the generated results are the final estimation by being averaged from 10 splits. Experiments of Setup IV are repeated for 100 times. Linear SVM is utilized for expression prediction.

4.3.2. Results on bosphorus database

The average confusion matrix for feature-level fusion with Setup IV is shown in Table 6. From this table, we can observe that: (i) Happy expression is the easiest to recognize, while fear expression is the most difficult to recognize; (ii) The RAs of disgust, sadness and fear expressions are all lower than 67.10%; (iii) Angry, disgust and fear expressions can be confused into any other expressions; (iv) The confusion probability of fear expression with surprise expression is higher than others, and vice versa. Meanwhile the same is true of confusion of angry expression with sadness expression. From Fig. 7, we can see that there are only very slight differences between fear and surprise pairs of the same person, and these very slight differences also occur between angry and sad pairs. Thus, Bosphorus database is very difficult for expression recognition compared with BU-3DFE database in this paper.

4.3.3. Comparison with other methods

Table 7 shows the performance comparisons of our proposed approach with state-of-the-art methods (i.e., [27,28]) on Bosphorus database in four aspects: data modality, expression features, classifiers, and accuracies. It is easily found from this table that method [27] obtains the lowest accuracy, and methods of [28] and our proposed approach obtain very similar results (75.83% vs. 75.93%).

(a) Angry and sadness pairs of six persons

(b) Fear and surprise pairs of six persons

Fig. 7. Twelve pairs of 2D texture images on Bosphorus database.

Among three methods, our proposed approach achieves the best results on Bosphorus database with Setup IV.

4.4. Validation of FERLrtc on synthetic data

4.4.1. Synthetic method

In this section, we validate FERLrTC on synthetic third-order tensor of size $10 \times 20 \times 30$ and fourth-order tensor of size $20 \times 20 \times 3 \times 50$ based on the Tucker decomposition model utilizing a random core tensor multiplied by random factor matrices along each mode under the noiseless condition. Note that the core tensor and all the factor matrices are drawn from a normal distribution. Assume that the sizes of two core tensors are of size (3, 4, 5) and (8, 8, 2, 10) according to the third-order and four-order tensor, respectively. Obviously, the groundtruth for these generated tensor ranks is (3, 4, 5) or (8, 8, 2, 10).

4.4.2. Compared algorithms and experiment protocol

Here, five Tucker decomposition algorithms (i.e., APG_NTDC, IRTD, WTucker, HaLRTC [64] and KBR_TC) are compared with FERLrTC under 30%, 50% and 80% SRs selected randomly. On the basis of tensor nuclear-norm, the HaLRTC proposes a tensor completion method for estimating missing data in tensors of visual data. Noted that the HaLRTC does not provide an explicit estimate of multilinear rank. For each setup, results are averaged by running 100 times independently.

4.4.3. Parameters setting

In our experiments, parameters are set in the following: i) APG_NTDC, λ_n (n= 1, 2, 3) and λ_c are all set to be 0.2, the predefined multilinear rank is set to be (3, 4, 5) and (8, 8, 2, 10), respectively; ii) IRTD, we let $\lambda_2 = 1$, $\lambda_1 = 0.1$ and γ = 1e-2; iii) WTucker, we predefined the multilinear rank to be (6, 8, 10) and (12, 12, 5, 15), respectively; iv) HaLRTC, ρ is set to be 1e−5; v) KBR_TC, λ, c, v, ρ and μ are all set to be 1e−1, 1e−3, 0.1, 1.05 and 1e−5, respectively; vi) FERLrTC, for the third-order tensor, we set w_1 = 1e−2, w_2 = 5e−3, $a_1 = 0.07$, $a_2 = 4$ and $a_3 = 8$. Meanwhile 0.8333, 0.8976, and 0.9583 of θ values correspond to 30%, 50% and 80% SRs, respectively. For the fourth-order tensor, w_1, a_1, a_2, a_3, a_4 and w_2 are set to be 1e−3, 0.7, 0.7, 10, 0.1 and 0.88, respectively. At

Table 7
Comparison of data modality, expression features, classifiers, and accuracies with the state-of-the-art on Bosphorus database with Setup IV.

Method	Data	Feature	Classifier	Setup IV
Li et al. [28]	3D	normals/LBP	MKL	75.83
Ujir et al. [27]	3D	surface normals	AdaBoosting	63.63
Ours	2D+3D	depth,normals, curvatures, textures/LBP	SVM	**75.93**

Table 8
RSE, running time (seconds) and rank on synthetic tensors.

(a) Tensor size: 10 × 20 × 30

	30% (SR)			50% (SR)			80% (SR)		
Method	RSE	TIME(s)	Rank	RSE	TIME(s)	Rank	RSE	TIME(s)	Rank
APG_NTDC	0.9806	0.1613	(3 4 5)	0.9738	0.2473	(3 4 5)	0.9580	0.3050	(3 4 5)
IRTD	0.2179	10.225	(3 4 4)	0.0951	9.5803	(3 4 5)	0.0432	9.1732	(3 4 5)
HaLRTC	0.8816	1.6851	–	0.8234	1.7429	–	0.6734	1.8132	–
WTucker	0.7376	174.6073	(6 8 10)	0.0176	163.3026	(6 8 10)	0.0076	119.1621	(6 8 10)
KBR_TC	0.2051	9.0345	(10 20 30)	0.0915	7.2404	(10 20 30)	0.0403	2.6978	(10 20 30)
FERLrTC	**0.0086**	**0.0393**	(3 4 5)	**0.0035**	**0.0395**	(3 4 5)	**0.0029**	**0.0397**	(3 4 5)

(b) Tensor size: 20 × 20 × 3 × 50

	30%(SR)			50%(SR)			80%(SR)		
Method	RSE	TIME(s)	Rank	RSE	TIME(s)	Rank	RSE	TIME(s)	Rank
APG_NTDC	0.0993	0.6063	(8 8 2 10)	0.0989	0.6665	(8 8 2 10)	0.0983	0.2464	(8 8 2 10)
IRTD	0.0277	45.2841	(8 8 2 10)	0.0134	41.1770	(8 8 2 10)	0.0079	50.286	(8 8 2 10)
HaLRTC	0.8660	5.9706	–	0.6933	5.9756	–	0.4082	5.9783	–
WTucker	0.2540	186.5053	(12 12 3 15)	0.0212	87.3351	(12 12 3 15)	0.0062	44.2379	(12 12 3 15)
KBR_TC	**0.0167**	13.2277	(20 20 3 50)	0.0124	17.7085	(20 20 3 50)	0.0089	17.5450	(20 20 3 50)
FERLrTC	0.0256	**0.1677**	(8 8 2 10)	**0.0120**	**0.1378**	(8 8 2 10)	**0.0053**	**0.0763**	(8 8 2 10)

the same time 0.7143, 0.8462, and 0.9333 of θ values correspond to 30%, 50% and 80% SRs, respectively. Other parameters are same as the settings of initialization in this section.

4.4.4. Results on synthetic data

Table 8 shows the comparison of IRTD, APG_NTDC, WTucker, HaLRTC, KBR_TC and our proposed approach (FERLrTC) in terms of the recovery accuracy (RSE), running time and rank on synthetic data of the third-order tensor size 10 × 20 × 30 and the fourth-order tensor size 20 × 20 × 3 × 50 for 100 independent runs.

From Table 8, we can observe that:

(i) FERLrTC presents the best performance of tensor completion in most cases on whether the third-order or fourth-order tensor. Also, FERLrTC can reliably estimate the true rank of both the third-order and fourth-order tensor. The running time of FERLrTC is the least compared with those of other competing algorithms.

(ii) Compared with IRTD, FERLrTC shows a better performance advantage in terms of RSE, running time and rank. The comparison results indicates fully that the potential low-rank structure can be characterized in terms of the low-rankness of the involved factor matrices and the structured sparsity of the involved core tensor under Tucker decomposition, and this low-rank structure is more characterized than that of IRTD, in which the group log-sum function imposed on the generated core tensor is combined with a Frobenius norm imposed on the generated factor matrices by Tucker decomposition.

(iii) FERLrTC surpasses APG_NTDC, WTucker and HaLRTC by a big margin, specially when SR is less than 50%. This corroborates that the low-rank tensor completion model of our proposed approach is more efficient than those of the other three algorithms. Meanwhile, this reflects the effectiveness of the rank reduction strategy.

(iv) FERLrTC performs better than KBR_TC in most cases in terms of RSE, running time, rank. Particularly, FERLrTC indicates its strong advantage in estimating the true multilinear rank compared with KBR_TC that makes use of the rank-increasing scheme, which shows our rank reduction strategy is more effective.

Table 9
Comparison with each single feature on BU-3DFE database with Setup I, II and III.

%	I_g	I_n^x	I_n^y	I_n^z	I_c
Setup I	71.36	71.62	**72.81**	71.22	72.54
Setup II	70.27	70.39	**71.35**	70.15	71.23
Setup III	69.37	**70.12**	69.48	69.16	69.29
%	I_{mc}	I_t^r	I_t^g	I_t^b	**All**
Setup I	70.83	71.03	71.25	72.38	**82.89**
Setup II	68.09	70.15	70.69	71.19	**80.91**
Setup III	67.38	68.46	68.61	69.17	**78.96**

Overall, our proposed approach (FERLrTC) presents a clear performance advantage over IRTD, APG_NTDC, WTucker, HaLRTC, and KBR_TC in terms of RSE, running time and rank.

4.5. Discussion

In this subsection, five issues will be discussed on BU-3DFE database: the effectiveness of a 4D tensor model based on feature-level fusion for 2D+3D FER, the selection of feature descriptors, complexity and convergence analysis, the effectiveness of the rank reduction strategy for 2D+3D FER.

4.5.1. Effectiveness of a 4D tensor model based on feature-Level fusion for 2D+3D FER

In order to better predict facial expressions, we have selected some discriminative features with LBP descriptor, such as the geometry map I_g, three normal component maps I_n^x, I_n^y and I_n^z, curvature maps (i.e., curvature I_c and mean curvature I_{mc}), and the 3-channel 2D texture information I_t^r, I_t^g and I_t^b. At the same time, the nine features are together combined to construct a 4D tensor model based on feature-level fusion by being stack.

Table 9 indicates the average recognition accuracies (RA) of six expressions with LBP descriptor by Setup I, II and III. From Table 9, we can conclude that: (i) Normal map I_n^y, curature map I_c and texture map I_t^b generally perform better than other facial attribute fea-

Table 10
Comparison with different single feature on Bosphorus database with Setup IV.

%	I_g	I_n^x	I_n^y	I_n^z	I_c
Setup IV	**70.47**	69.15	69.75	69.39	68.46
%	I_{mc}	I_t^r	I_t^g	I_t^b	**All**
Setup IV	67.31	64.53	64.85	65.68	**75.93**

Table 11
Recognition accuracies for one feature excluded at one time on BU-3DFE database with Setup I.

%	$-I_g$	$-I_n^x$	$-I_n^y$	$-I_n^z$	$-I_c$
Setup I	80.21	80.06	79.29	80.89	79.61
Difference	−2.68	−2.83	**−3.60**	−2.00	−3.28
%	$-I_{mc}$	$-I_t^r$	$-I_t^g$	$-I_t^b$	**All**
Setup I	80.2	80.55	80.13	79.50	**82.89**
Difference	−2.69	−2.34	−2.77	−3.39	0.00

tures with Setup I and II, while in Setup III, Normal map I_n^x obtains the best recognition rate; (ii) The fusion of all nine attribute features obtains the best performance; (iii) These results manifest that different features actually embody large supplemental information between 2D and 3D data for FER.

Table 10 reports the comparison results of our proposed approach (FERLrTC) with single feature on Bosphorus database. The conclusions are similar to those on BU-3DFE database with Setup I, II and III except the first conclusion. It is clear that geometry map I_g performs better than other facial attribute features.

Nine experiments have been carried out to validate the combination effectiveness with one feature excluded at one time. Table 11 shows recognition accuracies for one feature excluded at one time on BU-3DFE database with Setup I and their differences with the average recognition accuracy 82.89% in Table 1. From this table, it is easily found that the differences are in the range $[-3.60, -2.00]$, among which the differences of I_n^y and I_n^z achieve the lowest and highest, respectively. This comparison result manifests sufficiently the large complementarity among different modalities and verifies that any of nine kinds of features can not be excluded. From Tables 9, 10 and 11, we can see that utilizing a 4D tensor model based on feature-level fusion obtains better recognition accuracy than that of any other single feature.

4.5.2. Selection of feature descriptors

It is well known that there are other popular local descriptors, such as Dense-SIFT [65], HOG [66] and Gabor [67]. Here, our proposed approach is implemented by extracting nine kinds of features with the three descriptors respectively on BU-3DFE database with Setup I, the comparison results with LBP descriptor are shown in Table 12. From this table, it is easily observed that LBP and Dense-SIFT perform better than Gabor and HOG. In particular, LBP obtains the highest recognition accuracies of 82.89%, which performs Dense-SIFT, HOG, and Gabor by 1.73%, 11.32%, and 5.17%, respectively. Therefore, the LBP descriptor is effective and efficient to encode local structure of textons within an image patch [10].

Table 12
Comparison of different feature descriptors on BU-3DFE database with Setup I.

Feature Descriptor	HOG	Gabor	Dense-SIFT	LBP
Accuracy(%)	71.57	77.72	81.16	**82.89**

4.5.3. Complexity analysis

The main computation is taken when updating $\mathcal{X}^{[t]}$ and $\{(A^{(n)})^{[t]}\}$'s at each iteration. The computational complexity for updating $\mathcal{X}^{[t]}$ in Algorithm 1 is mainly reflected in the evaluation of gradient (13) and is of order $O(\sum_{n=1}^{4}(\prod_{k=1}^{n} I_k)(\prod_{j=n}^{4} R_j) + \sum_{n=1}^{4}(\prod_{k=1}^{n} R_k)(\prod_{j=n}^{4} I_j))$ that scales linearly with the data size. Meanwhile, the main computation complexity for updating $A^{(n)}$ via (17) is of order $O(2R_n\prod_{k=1}^{4} I_k + \sum_{k=1,k\neq n}^{4} R_n(\prod_{m=1,m\neq n}^{k} I_m)(\prod_{j=k,j\neq n}^{4} R_j))$, where the first term comes from $\nabla f_2((A^{(n)})^{[t]})$, and the second term stems from the computation of Φ_n, and its complexity scales linearly with the data size. Therefore, the overall computational complexity of every iteration is of order $O(\sum_{n=1}^{4}(\prod_{k=1}^{n} I_k)(\prod_{j=n}^{4} R_j) + \sum_{n=1}^{4}(\prod_{k=1}^{n} R_k)(\prod_{j=n}^{4} I_j))$ that scales linearly with the data size.

4.5.4. Convergence analysis

For the convergence analysis, the problem (6) is much more complicated. However, we have proven that MM algorithm with IADM generates a non-increasing objective function value by updating a variable while keeping other variables fixed. The proof is shown in Appendix C.

4.5.5. Effectiveness of the rank reduction strategy for 2D+3D FER

To validate the effectiveness of the rank reduction strategy (RRS) for 2D+3D FER, we compare the performance of FERLrTC with that achieved without RRS. The experiments are carried out on BU-3DFE database with Setup I. Table 13 indicates the comparison results in terms of recognition accuracy, iterations and computation cost at each iteration. From this table, we can observe that the recognition accuracy with RRS achieves higher 0.33% than that without RRS, and the corresponding number of iterations with RRS, however, have less 8 times than that without RRS. Meanwhile the computation cost at each iteration with RRS obtains lower than that without RRS. Thus, the performance of FERLrTC with RRS achieves better results, which fully illustrates that using RRS can not only retain strong interactions among factor matrices and the core tensor for 2D+3D FER, but also accelerate the convergency processing.

5. Conclusion and future work

In this paper, a new 4D tensor model has been built upon multimodal data including 2D face images and 3D face models to explore efficient structural information and correlations between different modalities, and a novel tensor dimensionality reduction approach for 2D+3D facial expression recognition via low-rank tensor completion (FERLrTC) is proposed and solved. The capability in tensor recovery is enhanced and the accuracy of expression

Table 13
Comparison results in terms of recognition accuracy, iterations and computation cost at each iteration on BU-3DFE database with Setup I.

	Recognition accuracy	Iterations	Computation cost at each iteration
With RRS	82.89%	7	$O\left(\sum_{n=1}^{4}\left(\prod_{k=1}^{n} I_k\right)\left(\prod_{j=n}^{4} R_j\right) + \sum_{n=1}^{4}\left(\prod_{k=1}^{n} R_k\right)\left(\prod_{j=n}^{4} I_j\right)\right)$
Without RRS	82.56%	15	$O\left(2\left(\sum_{n=1}^{4} I_n\right)\left(\prod_{n=1}^{4} I_n\right)\right)$

classification is promoted as the numerical results on BU-3DFE and Bosphorus databases. Meanwhile synthetic data also shows that our proposed approach could have competitive performance compared with other existing methods. To further improve the recognition accuracy, more effective features need to be appropriately extracted and a higher order tensor model will then be built. The resulting tensor optimization will be of a relatively large scale and more efficient and robust algorithms are then in need. All of these will be our future research topic.

Acknowledgments

The authors sincerely appreciated the editor and anonymous referees for their valuable comments and suggestions. This work was partly supported by the National Natural Science Foundation of China (61471032, 11771038, 11431002, 61403024, 61472030, 61772067, 61502491), Program for Innovative Research Team in University of Ministry of Education of China (IRT201206), Program for New Century Excellent Talents in University (NCET-12-0768), and the Fundamental Research Funds for the Central Universities (2017JBZ108).

Appendix A. A majorization function of $L(\mathcal{X}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y})$ in (7)

Followed by Yang et al. [40], the involved log-sum function in (7) can be approximated by

$$\begin{aligned} &Q(\mathcal{X}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y}|\mathcal{X}^{[t]}) \\ &= \langle \mathcal{X}, \mathcal{D}^{[t]} * \mathcal{X} \rangle + \gamma \sum_{n=1}^{4} \lambda_n \|A^{(n)}\|_* \\ &\quad + \mu \|\mathcal{Y} - \mathcal{X} \prod_{n=1}^{4} \times_n A^{(n)}\|_F^2 + \alpha, \end{aligned} \tag{23}$$

where $\alpha = \sum_{n=1}^4 \sum_{i=1}^{I_4} \log(\|\mathcal{X}_{(n,i)}^{[t]}\|_F^2 + \epsilon) - \sum_{n=1}^4 I_n$. Apparently, $Q(\mathcal{X}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y})$ is a majorization function of $L(\mathcal{X}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y})$, i.e., $L(\mathcal{X}^{[t]}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y}) = Q(\mathcal{X}^{[t]}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y}|\mathcal{X}^{[t]})$, and

$$Q(\mathcal{X}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y}|\mathcal{X}^{[t]}) \geqslant L(\mathcal{X}, \{A^{(n)}\}_{n=1}^4, \mathcal{Y}), \tag{24}$$

Thus, solving (7) converts to minimize the surrogate function (23) iteratively.

Appendix B. Derivation of the Eq. (15)

$$\begin{aligned} L(f) &= \lambda_{\max}(\mu H^T H) \\ &= \mu \lambda_{\max}\left(\bigotimes_n (A^{(n)})^T \bigotimes_n A^{(n)}\right), \\ &= \mu \lambda_{\max}\left(\bigotimes_n ((A^{(n)})^T A^{(n)})\right), \\ &= \mu \prod_{n=1}^{4} \lambda_{\max}((A^{(n)})^T A^{(n)}). \end{aligned}$$

Appendix C. Proof of convergency of the objective function

For the current iteration $\{\mathcal{X}^{[t]}, \{(A^{(n)})^{[t]}\}_{n=1}^4, \mathcal{Y}^{[t]}\}$, the non-increasing objective function value at the new iteration can be derived as follows

$$\begin{aligned} &L(\mathcal{X}^{[t]}, \{(A^{(n)})^{[t]}\}_{n=1}^4, \mathcal{Y}^{[t]}) \\ &\overset{(a)}{=} Q(\mathcal{X}^{[t]}, \{(A^{(n)})^{[t]}\}_{n=1}^4, \mathcal{Y}^{[t]}|\mathcal{X}^{[t]}), \\ &\overset{(b)}{\geqslant} Q(\mathcal{X}^{[t+1]}, \{(A^{(n)})^{[t]}\}_{n=1}^4, \mathcal{Y}^{[t]}|\mathcal{X}^{[t]}), \\ &\overset{(c)}{\geqslant} Q(\mathcal{X}^{[t+1]}, \{(A^{(n)})^{[t]}\}_{n=1}^4, \mathcal{Y}^{[t]}|\mathcal{X}^{[t+1]}), \\ &= L(\mathcal{X}^{[t+1]}, \{(A^{(n)})^{[t]}\}_{n=1}^4, \mathcal{Y}^{[t]}), \\ &\geqslant L(\mathcal{X}^{[t+1]}, (A^{(1)})^{[t+1]}, \{(A^{(n)})^{[t]}\}_{n=2}^4, \mathcal{Y}^{[t]}), \\ &\qquad \vdots \\ &\geqslant L(\mathcal{X}^{[t+1]}, \{(A^{(n)})^{[t+1]}\}_{n=1}^4, \mathcal{Y}^{[t]}), \\ &\geqslant L(\mathcal{X}^{[t+1]}, \{(A^{(n)})^{[t+1]}\}_{n=1}^4, \mathcal{Y}^{[t+1]}), \end{aligned}$$

where the equality (a) is from (24) when $\mathcal{X} = \mathcal{X}^{[t]}$, the first inequality (b) from (9) and the second inequality (c) from the definition of the majorization function Q.

References

[1] C.A. Corneanu, M. Oliu, J.F. Cohn, S. Escalera, Survey on RGB, 3d, thermal, and multimodal approaches for facial expression recognition: history, trends, and affect-related applications, IEEE Trans. Pattern Anal. Mach. Intell. 38 (8) (2016) 1548–1568.
[2] H. Li, J. Sun, Z. Xu, L. Chen, Multimodal 2D+3D facial expression recognition with deep fusion convolutional neural network, IEEE Trans. Multimed. 19 (12) (2017) 2816–2831.
[3] Q. Zhen, D. Huang, Y. Wang, L. Chen, Muscular movement model-based automatic 3d/4d facial expression recognition, IEEE Trans. Multimed. 18 (7) (2016) 1438–1450.
[4] Z. Zeng, M. Pantic, G.I. Roisman, T.S. Huang, A survey of affect recognition methods: audio, visual, and spontaneous expressions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (1) (2009) 39–58.
[5] A. Tawari, M.M. Trivedi, Face expression recognition by cross modal data association, IEEE Trans. Multimed. 15 (7) (2013) 1543–1552.
[6] S. Wang, Z. Liu, Z. Wang, G. Wu, P. Shen, S. He, X. Wang, Analyses of a multimodal spontaneous facial expression database, IEEE Trans. Affect. Comput. 4 (1) (2013) 34–46.
[7] M. Dahmane, J. Meunier, Prototype-based modeling for facial expression analysis, IEEE Trans. Multimed. 16 (6) (2014) 1574–1584.
[8] S. Zafeiriou, I. Pitas, Discriminant graph structures for facial expression recognition, IEEE Trans. Multimed. 10 (8) (2008) 1528–1540.
[9] H. Li, H. Ding, D. Huang, Y. Wang, X. Zhao, J.M. Morvan, L. Chen, An efficient multimodal 2d + 3d feature-based approach to automatic facial expression recognition, Comput. Vis. Image Understand. 140 (SCIA) (2015) 83–92.
[10] Y. Yao, D. Huang, X. Yang, Y. Wang, L. Chen, Texture and geometry scattering representation-based facial expression recognition in 2d+3d videos, ACM Trans. Multimed. Comput. Commun. Appl. 14 (1s) (2018) 18:1–18:23.
[11] Q. Xie, Q. Zhao, D. Meng, Z. Xu, Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery, IEEE Trans. Pattern Anal. Mach. Intell. 40 (99) (2017) 1888–1902.
[12] Q. Li, D. Schonfeld, Multilinear discriminant analysis for higher-order tensor data classification., IEEE Trans. Pattern Anal. Mach. Intell. 36 (12) (2014) 2524–2537.
[13] M. Haardt, F. Roemer, G.D. Galdo, Higher-order SVD-based subspace estimation to improve the parameter estimation accuracy in multidimensional harmonic retrieval problems, IEEE Trans. Signal Process. 56 (7) (2008) 3198–3213.
[14] T.G. Kolda, B.W. Bader, Tensor decompositions and applications, SIAM Rev. 51 (3) (2009) 455–500.
[15] A. Cichocki, D. Mandic, L.D. Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, H.A. Phan, Tensor decompositions for signal processing applications: from two-way to multiway component analysis, IEEE Signal Process. Mag. 32 (2) (2015) 145–163.
[16] Y. Fu, J. Gao, D. Tien, Z. Lin, H. Xia, Tensor LRR and sparse coding-based subspace clustering, IEEE Trans. Neural Netw. Learn. Syst. 27 (10) (2016) 2120–2133.
[17] X. Yang, D. Huang, Y. Wang, L. Chen, Automatic 3D facial expression recognition using geometric scattering representation, in: Proceedings of the IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, 2015, pp. 1–6.
[18] Y. Fu, Q. Ruan, G. An, Y. Jin, Fast nonnegative tensor factorization based on graph-preserving for 3D facial expression recognition, in: Proceedings of the IEEE International Conference on Signal Processing, 2017, pp. 292–297.
[19] A. Maalej, B.B. Amor, M. Daoudi, A. Srivastava, S. Berretti, Shape analysis of local facial patches for 3d facial expression recognition, Pattern Recognit. 44 (8) (2011) 1581–1589.
[20] H. Soyel, H. Demirel, 3D facial expression recognition with geometrically localized facial features, in: Proceedings of the International Symposium on Computer and Information Sciences, 2008, pp. 1–4.
[21] H. Tang, T.S. Huang, 3D facial expression recognition based on properties of line segments connecting facial feature points, in: Proceedings of the IEEE International Conference on Automatic Face & Gesture Recognition, 2008, pp. 1–6.
[22] K. Yurtkan, H. Demirel, Entropy-based feature selection for improved 3d facial expression recognition, Signal Image Video Process. 8 (2) (2014) 267–277.

[23] S. Berretti, B. Ben Amor, M. Daoudi, A. Del Bimbo, 3D facial expression recognition using sift descriptors of automatically detected keypoints, Visual Comput. 27 (11) (2011) 1021–1036.
[24] X. Li, Q. Ruan, G. An, Analysis of range images used in 3d facial expression recognition, Comput. Inf. 35 (2016) 1001–1019.
[25] W. Zeng, H. Li, L. Chen, J.M. Morvan, X.D. Gu, An automatic 3D expression recognition framework based on sparse representation of conformal images, in: Proceedings of the IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, 2013, pp. 1–8.
[26] H. Ujir, M. Spann, I.H.M. Hipiny, 3D facial expression classification using 3d facial surface normals, in: Proceedings of the Eighth International Conference on Robotic,Vision,Signal Processing & Power Applications, 2014, pp. 245–253.
[27] H. Ujir, M. Spann, Surface normals with modular approach and weighted voting scheme in 3d facial expression classification, Int. J. Comput. Inf. Technol. 3 (05) (2014) 909–918.
[28] H. Li, L. Chen, D. Huang, Y. Wang, 3D facial expression recognition via multiple kernel learning of multi-scale local normal patterns, in: Proceedings of the International Conference on Pattern Recognition, 2012, pp. 2577–2580.
[29] P. Lemaire, M. Ardabilian, L. Chen, M. Daoudi, Fully automatic 3D facial expression recognition using differential mean curvature maps and histograms of oriented gradients, in: Proceedings of the Joint ACM Workshop on Human Gesture and Behavior Understanding, 2013, pp. 1–7.
[30] X. Zhao, D. Huang, E. Dellandrĺęa, L. Chen, Automatic 3D facial expression recognition based on a Bayesian belief net and a statistical facial feature model, in: Proceedings of the International Conference on Pattern Recognition, 2010, pp. 3724–3727.
[31] I. Mpiperis, S. Malassiotis, M.G. Strintzis, Bilinear models for 3d face and facial expression recognition, IEEE Trans. Inf. Forensics Secur. 3 (3) (2008) 498–511.
[32] B. Gong, Y. Wang, J. Liu, X. Tang, Automatic facial expression recognition on a single 3D face by exploring shape deformation, in: Proceedings of the ACM International Conference on Multimedia, 2009, pp. 569–572.
[33] E.J. CandÍÍs, B. Recht, Exact matrix completion via convex optimization, Found. Comput. Math. 9 (6) (2009) 717.
[34] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306.
[35] Z. Lin, A review on low-rank models in data analysis, Big Data Inf. Anal. 1 (2/3) (2017) 139–161.
[36] Y. Xu, D. Zhang, J. Yang, J.Y. Yang, A two-phase test sample sparse representation method for use with face recognition, IEEE Trans. Circuits Syst. Video Technol. 21 (9) (2011) 1255–1262.
[37] O. Taheri, S.A. Vorobyov, Sparse channel estimation with l_p-norm and reweighted l_1-norm penalized least mean squares, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 2011, pp. 2864–2867.
[38] X. Wang, C. Navasca, Adaptive low rank approximation for tensors, in: Proceedings of the IEEE International Conference on Computer Vision Workshop, 2015, pp. 939–945.
[39] Y. Shen, J. Fang, H. Li, Exact reconstruction analysis of log-sum minimization for compressed sensing, IEEE Signal Process. Lett. 20 (12) (2013) 1223–1226.
[40] L. Yang, J. Fang, H. Li, B. Zeng, An iterative reweighted method for tucker decomposition of incomplete tensors, IEEE Trans. Signal Process. 64 (18) (2016) 4817–4829.
[41] D.R. Hunter, K. Lange, A tutorial on MM algorithms, Am. Stat. 58 (1) (2004) 30–37.
[42] S. Zhang, J. Ang, J. Sun, An alternating direction method for solving convex nonlinear semidefinite programming problems, Optimization 62 (4) (2013) 527–543.
[43] M. Yamagishi, I. Yamada, Over-relaxation of the fast iterative shrinkage-thresholding algorithm with variable stepsize, Inverse Probl. 27 (10) (2011) 105008–105022. (15)
[44] L. Yin, X. Wei, Y. Sun, J. Wang, M.J. Rosato, A 3D facial expression database for facial behavior research, in: Proceedings of the International Conference on Automatic Face and Gesture Recognition, 2006, pp. 211–216.
[45] A. Savran, N. Alyüz, H. Dibeklioğlu, O. Çeliktutan, B. Gökberk, B. Sankur, L. Akarun, Bosphorus database for 3D face analysis, in: Proceedings of the Biometrics and Identity Management, 2008, pp. 47–56.
[46] T.C. Faltemier, K.W. Bowyer, P.J. Flynn, A region ensemble for 3d face recognition, IEEE Trans. Inf. Forensics Secur. 3 (1) (2008) 62–73.
[47] C. Shan, S. Gong, P.W. Mcowan, Facial expression recognition based on local binary patterns: a comprehensive study, Image Vision Comput. 27 (6) (2009) 803–816.
[48] L.D. Lathauwer, B.D. Moor, J. Vandewalle, A multilinear singular value decomposition, SIAM J. Matrix Anal. Appl. 21 (4) (2000) 1253–1278.
[49] H. Li, J. Sun, D. Wang, Z. Xu, L. Chen, Deep representation of facial geometric and photometric attributes for automatic 3d facial expression recognition, arXiv:1511.03015 (2015).
[50] Y. Xu, Alternating proximal gradient method for sparse nonnegative tucker decomposition, Math. Program. Comput. 7 (1) (2015) 39–70.
[51] M. Filipović, A. Jukić, Tucker factorization with missing data with application to low-rank tensor completion, Multidimens. Syst. Signal Process. 26 (3) (2015) 1–16.
[52] Y. Xu, R. Hao, W. Yin, Z. Su, Parallel matrix factorization for low-rank tensor completion, Inverse Probl. Imaging 9 (2) (2017) 601–624.
[53] J. Wang, L. Yin, X. Wei, Y. Sun, 3D facial expression recognition based on primitive surface feature distribution, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2, 2006, pp. 1399–1406.
[54] H. Soyel, H. Demirel, Facial expression recognition using 3D facial feature distances, in: Proceedings of the International Conference Image Analysis and Recognition, 2007, pp. 831–838.
[55] S. Berretti, A.D. Bimbo, P. Pala, B.B. Amor, M. Daoudi, A set of selected sift features for 3D facial expression recognition, in: Proceedings of the International Conference on Pattern Recognition, 2010, pp. 4125–4128.
[56] W. Zeng, H. Li, L. Chen, J.M. Morvan, X.D. Gu, An automatic 3D expression recognition framework based on sparse representation of conformal images, in: Proceedings of the IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, 2013, pp. 1–8.
[57] K. Yurtkan, H. Demirel, Feature selection for improved 3d facial expression recognition, Pattern Recognit. Lett. 38 (1) (2014) 26–33.
[58] A. Azazi, S.L. Lutfi, I. Venkat, Analysis and evaluation of surf descriptors for automatic 3d facial expression recognition using different classifiers, in: Proceedings of the Information and Communication Technologies, 2014, pp. 23–28.
[59] H. Li, H. Ding, D. Huang, Y. Wang, X. Zhao, J.M. Morvan, L. Chen, An efficient multimodal 2d + 3d feature-based approach to automatic facial expression recognition, Comput. Vision Image Understand. 140 (SCIA) (2015) 83–92.
[60] Z. Chen, D. Huang, Y. Wang, L. Chen, Fast and light manifold CNN based 3d facial expression recognition across pose variations, in: Proceedings of the 2018 ACM Multimedia Conference on Multimedia Conference, ACM, 2018, pp. 229–238.
[61] M. Hollander, D.A. Wolfe, E. Chicken, Nonparametric Statistical Methods, Taylor & Francis Ltd., 1999.
[62] H. Soyel, H. Demirel, 3D facial expression recognition with geometrically localized facial features, in: Proceedings of the International Symposium on Computer and Information Sciences, 2008, pp. 1–4.
[63] H. Tang, T.S. Huang, 3D facial expression recognition based on automatically selected features, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008, pp. 1–8.
[64] L. Ji, M. Przemyslaw, W. Peter, Y. Jieping, Tensor completion for estimating missing values in visual data, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 208–220.
[65] C. Liu, J. Yuen, A. Torralba, Sift flow: dense correspondence across scenes and its applications., IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 978–994.
[66] D. Navneet, T. Bill, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Society Conference on Computer Vision & Pattern Recognition, 2005, pp. 886–893.
[67] Z. Zhang, M. Lyons, M. Schuster, S. Akamatsu, Comparison between geometry-based and Gabor-wavelets-based facial expression recognition using multi-layer perceptron, in: Proceedings of the IEEE International Conference on Automatic Face & Gesture Recognition, 1998.

Yunfang Fu is currently a Ph.D. candidate of Institute of Information Science at Beijing Jiaotong University, PR China. She is also an Associate Professor of School of Computer Science & Engineering at Shijiazhuang University, PR China. She received her master degree of engineering in Instrument Science and Technology from Yanshan University, PR China in 2013. Her research interests include tensor analysis, signal processing, pattern recognition.

Qiuqi Ruan received the B.S. and M.S. degree from Beijing Jiaotong University in 1969 and 1981, respectively. He has published 3 books in the image processing and information science and more than 100 papers, and achieved a national patent. He is currently a professor and doctorate supervisor in Beijing Jiatong Univerisity and a senior member of IEEE. His main research interests include digital signal processing, computer vision, pattern recognition, and virtual reality etc.

Ziyan Luo received her Ph.D. degree in Operations Research from Beijing Jiaotong University. She is currently an Associate Professor at the State Key Laboratory of Rail Traffic Control and Safety at Beijing Jiaotong University. She was a research associate at The Hong Kong Polytechnic University (2010,2015), and a visiting scholar at Stanford University (2011–2012) and at National University of Singapore (2015–2016). Her research interests include tensor analysis and computation, sparse and low-rank optimization methods, etc.

Yi Jin received the Ph.D. degree in Signal and Information Processing from the Institute of Information Science, Beijing Jiaotong University in 2010. She is currently an Assistant Professor in the School of Computer Science and Information Technology, Beijing Jiaotong University. She was a visiting scholar in Nanyang Technological University of Singapore (2013–2014). She has served as the guest editor for special issues in Mathematical Problems in Engineering. Her research interests include computer vision, pattern recognition, image processing and machine learning.

Gaoyun An received the B.S. degree in Biological Engineering and Ph.D. degree in Signal and Information Processing from Beijing Jiaotong University in 2003 and 2008, respectively, Beijing, China. Currently, he is an associate professor in Institute of Information Science, Beijing Jiaotong University, Beijing, China. His main research interests include image processing, computer vision and pattern recognition.

Jun Wan received his Ph.D. degree from the Institute of Information Science, Beijing Jiaotong University, Beijing, China, in 2015. He is currently an assistant professor at the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Science (CASIA). His main research interests include computer vision, machine learning, especially for gesture and action recognition, facial attribution analysis (i.e. age estimation, facial expression, gender and race classification). He has published papers in top journals, such as JMLR, TPAMI, TIP, and TCYB. He has served as the reviewer on several top journals and conferences, such as JMLR, TPAMI, TIP, TMM, TSMC, PR, ICPR2016, CVPR2017, ICCV2017, FG2017.