Pattern Recognition 66 (2017) 44–52

Contents lists available at ScienceDirect

Pattern Recognition

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

Multi-task mid-level feature learning for micro-expression recognition

Jiachi He[a], Jian-Fang Hu[b,c], Xi Lu[b,d], Wei-Shi Zheng[b,e,*]

[a] School of Electronic and Information Engineering, Sun Yat-sen University, Guangzhou 510006, China
[b] School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[c] Guangdong Province Key Laboratory of Information Security, PR China
[d] Collaborative Innovation Center of High Performance Computing, National University of Defense Technology, Changsha 410073, China
[e] Key Laboratory of Machine Intelligence and Advanced Computing (Sun Yat-sen University), Ministry of Education, China

ARTICLE INFO

Keywords:
Micro-expression recognition
Multi-task learning

ABSTRACT

Due to the short duration and low intensity of micro-expressions, the recognition of micro-expression is still a challenging problem. In this paper, we develop a novel multi-task mid-level feature learning method to enhance the discrimination ability of extracted low-level features by learning a set of class-specific feature mappings, which would be used for generating our mid-level feature representation. Moreover, two weighting schemes are employed to concatenate different mid-level features. We also construct a new mobile micro-expression set to evaluate the performance of the proposed mid-level feature learning framework. The experimental results on two widely used non-mobile micro-expression datasets and one mobile micro-expression set demonstrate that the proposed method can generally improve the performance of the low-level features, and achieve comparable results with the state-of-the-art methods.

1. Introduction

Emotion recognition has drawn more and more attention over the past few decades. One major research topic in the emotion analysis is to recognize facial expression [1–3]. Compared with the traditional facial expression problem, micro-expression recognition is still a relatively new topic with many challenges. Micro-expression, once called micro-momentary facial expression in 1966 [4], was renamed by Ekman [5] a few years later. According to Ekman's studies [6–8], micro-expression is defined as a very brief and subtle movement on the face, which is uncontrollable to human themselves. With its close relationship with genuine emotions, micro-expression can serve as an important cue to reveal the emotions people try to conceal, especially in some high-stake situations. For this characteristic, micro-expression has a wide range of potential applications in diverse fields including criminal interrogation and clinical diagnosis.

Studies show that muscles in human face cannot be fully stretched to form a perceptible facial expression within 0.5 s [9,10], so it is not easy for human beings without any professional knowledge to accurately detect and recognize micro-expressions. To help detecting and recognizing micro-expressions, Ekman [11] introduced a Micro-Expression Training Tool (METT), in which the professional knowledge about seven subtle facial expressions are taught to the participants. However, the recognition performance was still unsatisfactory even after METT's training [12]. Moreover, recognition by human beings is easily affected by human's perception, making the results diverse among different subjects and at different time.

Recently, a lot of efforts have been made to develop computer vision techniques for the micro-expression recognition. Most existing methods [13–15] intend to simply concatenate the low-level features extracted from different local regions together for recognition. These methods generally expect that the extracted low-level features are representative enough to depict the expressions. However, due to the short duration and low intensity of micro-expression, the low-level features without any processing can hardly capture and reflect the critical movements in micro-expression. Moreover, the irrelevant and noisy information involved in video clips will further weaken the representation ability of the features, especially for the features extracted from inactive regions[1] with less dynamics.

In this paper, a mid-level feature learning mechanism is formulated, which processes the low-level features extracted from each facial region independently. For each region, a number of class-specific mappings are learned for projecting the original feature space to numerous subspaces, in one of which samples of the specific class are pulled closer while the samples from different classes are pushed farther. But different from [16] which intends to learn different

* Corresponding author.
E-mail addresses: hjcmail12315@gmail.com (J. He), hujianf@mail2.sysu.edu.cn (J.-F. Hu), luxialan12@gmail.com (X. Lu), wszheng@ieee.org (W.-S. Zheng).

[1] In this paper, active regions are defined as the regions with relatively large and perceptible facial movements in general, while the inactive ones are not.

http://dx.doi.org/10.1016/j.patcog.2016.11.029
Received 15 July 2016; Received in revised form 1 November 2016; Accepted 30 November 2016
Available online 05 December 2016

mappings independently, we explicitly introduce a common mapping to constrain the mapping learning. With this restriction, learnings of different feature mappings are linked together, and the common information among them can be mined for boosting our feature learning. In this way, a more discriminative mid-level feature with better generalization ability can be obtained to represent each facial region. We call our learning the *multi-task mid-level feature learning*. The mid-level features of all the local regions are then concatenated together for recognition. To further improve the system performance, two different weighting schemes are utilized for the feature concatenation.

In addition, micro-expressions collected in the existing works [17–19] are only recorded by digital video cameras. In some emergency situations, the video clips we can utilize for criminal investigation could be recorded by a mobile device with relatively low quality, which makes the micro-expression recognition problem more challenging. In this paper, we have additionally collected a mobile micro-expression dataset for evaluation. The results show that our method can be well generalized to tackle the low-quality micro-expression recognition problem.

In general, the experimental results on two widely used non-mobile micro-expression datasets and one mobile micro-expression set demonstrate the effectiveness of the proposed method.

2. Related work

For the purpose of addressing micro-expression recognition problem, some low-level features (e.g. LBP-TOP) were proposed at the early stage. LBP-TOP, which was first adopted in traditional facial expression recognition [20], is a 3D variant of LBP. By encoding the binary relationship patterns between each pixel and its neighbors on three orthogonal planes, the dynamic texture of the entire video can be represented. For more efficient computation, two more compact and lightweight representations, called LBP-SIP [21] and LBP-MOP [22], were also presented. Unlike LBP-TOP where all the adjacent points are concerned in the feature computation, LBP-SIP only considers the six neighbors on the intersecting lines. More compactly, LBP-MOP is constructed by concatenating LBP features from only three mean images, which are the pooling results of the respective stacks along three orthogonal directions. To represent micro-expressions in an intuitional way, Liong and Phan [23] also proposed an optical-strain-based feature for recognition.

To improve performance of recognition, several methods were proposed to enhance the low-level features. To emphasize the importance of active regions, Liong et al. [13] proposed to concatenate local features with different weights. Regions with higher optical strain magnitudes are thought to be more important, and will be weighted with larger values. With direction information considered on each orthogonal plane, LBP-TOP was further extended to Local Spatiotemporal Directional (LSTD) feature in [24]. Considering the close relationship between color and emotions, Wang et al. [14,25] proposed to extract LBP-TOP features from the tensor independent color space (TICS). Compared with RGB, color components in TICS are less related and thus more discriminative features can be extracted for recognition. To avoid the statistical instability of LBP-TOP, a re-parametrization technique based on the second local Gaussian jet was proposed in [26].

Aside from methods which enhance the basic low-level features, there are still numerous methods proposed to extract other robust representations. Lu et al. [27] presented a Delaunay-based temporal coding model (DTCM), which encodes local temporal variations in each subregion and represents the total variation by only preserving the ones with high saliency. Oh et al. [28] proposed a monogenic Riesz wavelet representation, where a two-layer architecture is adopted to extract magnitude, phase and orientation features of different scales. With nice facial alignment, Liu et al. [15] also proposed a Main Directional Mean Optical-flow (MDMO) feature. By removing small head movements in optical flow domain, more reliable local information can be obtained with less noisy influence. To preserve the shape properties of micro-expressions, a representation called spatiotemporal local binary pattern with integral projection (STLBP-IP) was presented by Huang et al. [29], where 1D- and 2DLBP features are extracted from integral images along two orthogonal directions. With efficient vector quantization and Fisher criterion, Huang et al. [30] proposed another spatiotemporal feature, where compact and discriminative codebooks are learned for feature extraction.

From the methods mentioned above, we can notice that most existing methods adopt a similar framework, where local features are extracted from different facial regions and then simply concatenated for recognition. However, with irrelevant and noisy information involved, the subtle dynamic patterns of micro-expression are not easy to be captured and represented by the low-level features, especially for the ones from inactive regions. Different from the normal facial expression, even a subtle movement in inactive region could serve as an important cue of micro-expression. With less discriminative information extracted from the inactive regions, features concatenated by the low-level ones may lead to questionable results and poor performance. To tackle this problem, a multi-task mid-level feature learning method is proposed to enhance the discrimination ability of the low-level features. For further improvement of the concatenated features, two weighting schemes are also presented.

3. Proposed approach

The overall framework of the proposed method consists of two parts: Firstly, enhancing the discrimination ability of local features by utilizing a multi-task mid-level feature learning mechanism, where several class-specific feature mappings are learned as illustrated in Fig. 1. Secondly, concatenating the enhanced mid-level features from the same videos by two different weighting schemes, and then making decision using SVM classifiers with RBF kernel [31].

3.1. Multi-task mid-level feature learning

Suppose that there are c micro-expression classes included in the recognition problem. For the i-th micro-expression, we have N_i video clips for training. Video clips are first spatially divided into $a \times b$ non-overlapping blocks. Then we extract low-level features from all the blocks and $t = a \times b$ low-level features are obtained to represent each video clip. Let us denote the corresponding low-level features as $\{\vec{x}_{ij}^{k}\}_{i,j,k=1,1,1}^{c,N_i,t}$, where $\vec{x}_{ij}^{k} \in R^d$ is the k-th low-level feature for the j-th training sample from the i-th micro-expression. To be concise, $\vec{x}_{ij}^{k}$ will be rewritten as $\vec{x}_{ij}$ in the following.

For each low-level feature $\vec{x}_{ij}$, we first calculate its top k_{inter} interclass nearest neighbors $N_{inter}(\vec{x}_{ij})$ from other micro-expressions and k_{intra} intraclass nearest neighbors $N_{intra}(\vec{x}_{ij})$ from the same micro-expression based on the Euclid metric. Then the inter(intra) class similarity between local feature $\vec{x}_{ij}$ and the p-th interclass nearest neighbor $\vec{x}_{ijp}$ (the q-th intraclass nearest neighbor $\vec{x}_{ijq}$) can be defined as follows:

$$A_{ijp} = \begin{cases} \exp\left(-\dfrac{\|\vec{x}_{ij} - \vec{x}_{ijp}\|^2}{\sigma^2}\right), & \text{if } \vec{x}_{ijp} \in N_{inter}(\vec{x}_{ij}) \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

$$B_{ijq} = \begin{cases} \exp\left(-\dfrac{\|\vec{x}_{ij} - \vec{x}_{ijq}\|^2}{\sigma^2}\right), & \text{if } \vec{x}_{ijq} \in N_{intra}(\vec{x}_{ij}) \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

where σ is a parameter to scale the similarities of different point pairs.

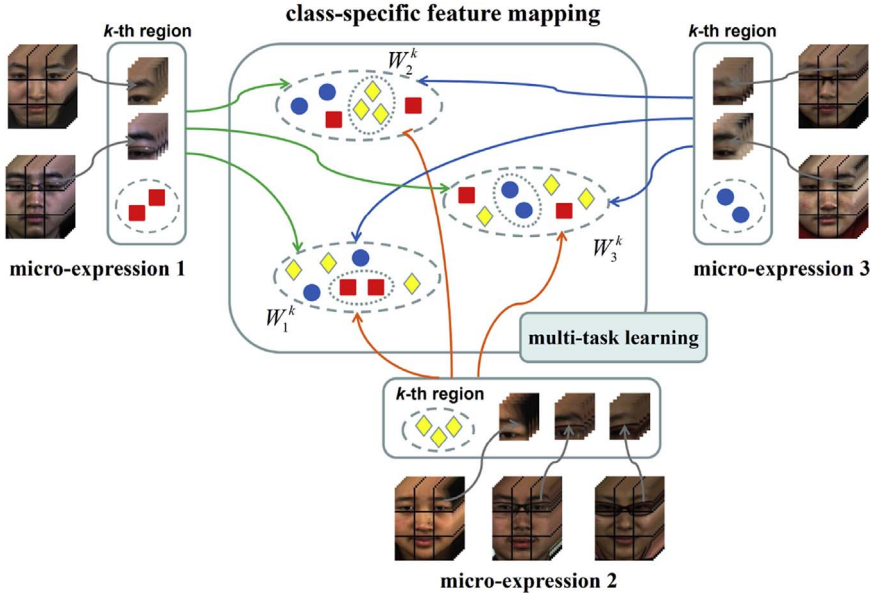

Fig. 1. A graphic illustration of the proposed feature learning framework. We first partition each video clip into smaller regions and then extract the low-level features to represent each region. For each region (e.g. the k-th region), we learn several class-specific feature mappings $\{\boldsymbol{W}_i^k\}_{i=1}^c$ under the multi-task learning mechanism, where $\boldsymbol{W}_i^k$ pulls the features of the i-th micro-expression together and pushes the features from other classes farther.

Based on the above inter(intra) class similarities, we formulate our mid-level feature learning as:

$$\begin{aligned}\min_{\boldsymbol{W}_1,\ldots,\boldsymbol{W}_c,\boldsymbol{\theta}} \quad & L = -J_1 + J_2 + R \\ = & -\sum_{i=1}^{c}\left(\frac{1}{N_i \times k_{inter}}\sum_{j=1}^{N_i}\sum_{p=1}^{k_{inter}} \|\boldsymbol{W}_i^T\vec{x}_{ij} - \boldsymbol{W}_i^T\vec{x}_{ijp}\|^2 A_{ijp}\right) \\ & + \sum_{i=1}^{c}\left(\frac{1}{N_i \times k_{intra}}\sum_{j=1}^{N_i}\sum_{q=1}^{k_{intra}} \|\boldsymbol{W}_i^T\vec{x}_{ij} - \boldsymbol{W}_i^T\vec{x}_{ijq}\|^2 B_{ijq}\right) \\ & + \alpha\sum_{i=1}^{c}\|\boldsymbol{W}_i - \boldsymbol{\theta}\|^2 \text{ subject to } \quad \boldsymbol{W}_i^T\boldsymbol{W}_i = \boldsymbol{I}, \quad i = 1, 2, \ldots, c.\end{aligned} \tag{3}$$

Here, $\boldsymbol{W}_i \in R^{d\times h}$ represents a class-specific feature mapping for the i-th micro-expression. Our objective function consists of three terms, and we will discuss each term in detail in the following:

Term J_1: J_1 is employed to quantify the distance between samples from different classes. Unlike the LDA method which separates different classes by a common subspace, we aim to learn several class-specific feature mappings simultaneously. Each feature mapping corresponds to a specific class, and the samples from other classes are expected to be separated by the mapping as shown in Fig. 2. We also note that the more similar two samples are, the higher A_{ijp} will be obtained, which leads to larger margin between them in the feature subspace. So by minimizing $-J_1$, samples from different micro-expressions will be well separated in the corresponding subspaces, thus more discriminative features can be learned for recognition.

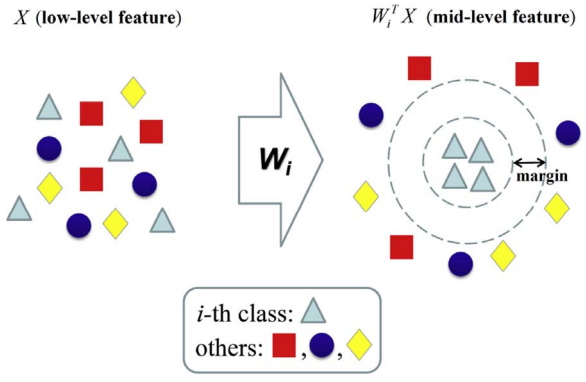

Fig. 2. Illustration of feature projection of the i-th class-specific feature mapping.

Note that J_1 can be conveniently rewritten as:

$$J_1 = \sum_{i=1}^{c}\mathrm{tr}(\boldsymbol{W}_i^T\boldsymbol{H}_{i,inter}\boldsymbol{W}_i). \tag{4}$$

where

$$\boldsymbol{H}_{i,inter} = \frac{1}{N_i \times k_{inter}}\sum_{j=1}^{N_i}\sum_{p=1}^{k_{inter}}(\vec{x}_{ij} - \vec{x}_{ijp})(\vec{x}_{ij} - \vec{x}_{ijp})^T A_{ijp}. \tag{5}$$

and $\mathrm{tr}(\cdot)$ is the trace operation.

Term J_2: Different from J_1, this term is used to quantify the distance between samples from the same class. J_2 is minimized such that samples from the same class are clustered together. Thus, the components that are consistent in each class will be preserved for recognition.

Similar to J_1, J_2 can be rewritten as:

$$J_2 = \sum_{i=1}^{c}\mathrm{tr}(\boldsymbol{W}_i^T\boldsymbol{H}_{i,intra}\boldsymbol{W}_i). \tag{6}$$

where

$$\boldsymbol{H}_{i,intra} = \frac{1}{N_i \times k_{intra}}\sum_{j=1}^{N_i}\sum_{q=1}^{k_{intra}}(\vec{x}_{ij} - \vec{x}_{ijq})(\vec{x}_{ij} - \vec{x}_{ijq})^T B_{ijq}. \tag{7}$$

Term R: This term is used to establish linkages among the learnings of different mappings. Since the movements contained in micro-expression are subtle, the information that can be used for model training is limited, and thus learning different mappings independently could easily lead to a solution with poor generalization ability. To tackle this problem, we explicitly introduce a common mapping $\boldsymbol{\theta}$ to constrain the mapping learning:

$$\boldsymbol{W}_i = \boldsymbol{U}_i + \boldsymbol{\theta}. \tag{8}$$

Here, $\boldsymbol{\theta}$ represents a common structure shared by different mappings, and $\boldsymbol{U}_i$ can be regarded as the specific component. With the restriction of $\boldsymbol{\theta}$, learnings of different $\boldsymbol{W}_i$ are linked together. By learning different mappings simultaneously, the common information among them can

be mined and supplied for each individual feature mapping learning. With more available information, features with better generalization ability can be obtained for recognition. The similarities of different mappings are measured by the term R, and its effect is controlled by α.

R can be rewritten as:

$$R = \alpha \operatorname{tr}\left(\sum_{i=1}^{c} \boldsymbol{W}_i \boldsymbol{W}_i^T\right) + \alpha c \operatorname{tr}(\boldsymbol{\theta}\boldsymbol{\theta}^T) - \alpha \operatorname{tr}\left[\left(\sum_{i=1}^{c} \boldsymbol{W}_i\right)\boldsymbol{\theta}^T + \boldsymbol{\theta}\left(\sum_{i=1}^{c} \boldsymbol{W}_i^T\right)\right]. \tag{9}$$

Then we can rewrite Eq. (3) as:

$$L = \sum_{i=1}^{c} \operatorname{tr}[\boldsymbol{W}_i^T(\boldsymbol{H}_{i,intra} - \boldsymbol{H}_{i,inter})\boldsymbol{W}_i] + \alpha \operatorname{tr}\left(\sum_{i=1}^{c} \boldsymbol{W}_i \boldsymbol{W}_i^T\right) + \alpha c \operatorname{tr}(\boldsymbol{\theta}\boldsymbol{\theta}^T) - \alpha \operatorname{tr}\left[\left(\sum_{i=1}^{c} \boldsymbol{W}_i\right)\boldsymbol{\theta}^T + \boldsymbol{\theta}\left(\sum_{i=1}^{c} \boldsymbol{W}_i^T\right)\right] \quad \text{subject to } \boldsymbol{W}_i^T\boldsymbol{W}_i = \boldsymbol{I},$$

$$i = 1, 2, \ldots, c. \tag{10}$$

3.2. Optimization

It is not easy to solve the above problem by optimizing $\{\boldsymbol{W}_i\}_{i=1}^c$ and $\boldsymbol{\theta}$ simultaneously. We have to solve it in an iterative manner as introduced below.

Updating $\boldsymbol{W}_i$ with $\boldsymbol{\theta}$ fixed: When $\boldsymbol{\theta}$ is fixed, the gradient of Eq. (10) with respect to $\boldsymbol{W}_i$ can be calculated as:

$$\frac{\partial L}{\partial \boldsymbol{W}_i} = 2(\boldsymbol{H}_{i,intra} - \boldsymbol{H}_{i,inter})\boldsymbol{W}_i + 2\alpha\boldsymbol{W}_i - 2\alpha\boldsymbol{\theta}$$
$$= 2[(\boldsymbol{H}_{i,intra} - \boldsymbol{H}_{i,inter} + \alpha\boldsymbol{I})\boldsymbol{W}_i - \alpha\boldsymbol{\theta}]. \tag{11}$$

Due to the orthogonality constraint, we cannot update $\boldsymbol{W}_i$ by simply setting $\partial L/\partial \boldsymbol{W}_i$ to be $\boldsymbol{0}$. Here, we use a generalized gradient descent method on the Grassman manifold in [32,37] to achieve the optimization.

Updating $\boldsymbol{\theta}$ with $\{\boldsymbol{W}_i\}_{i=1}^c$ fixed: When $\{\boldsymbol{W}_i\}_{i=1}^c$ are fixed, the gradient of Eq. (10) with respect to $\boldsymbol{\theta}$ can be calculated as:

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = -2\alpha\left(\sum_{i=1}^{c} \boldsymbol{W}_i\right) + 2\alpha c\boldsymbol{\theta}. \tag{12}$$

By letting $\partial L/\partial\boldsymbol{\theta} = \boldsymbol{0}$, $\boldsymbol{\theta}$ can be updated as:

$$\boldsymbol{\theta} = \frac{1}{c}\left(\sum_{i=1}^{c} \boldsymbol{W}_i\right). \tag{13}$$

In this way, we can get c class-specific feature mappings for each facial region. The detailed procedures are outlined in Algorithm 1.

Algorithm 1.

Input: local features of training samples $\{\vec{x}_{ij}^k\}_{i,j,k=1,1,1}^{c,N_i,t}$, dimension of class-specific feature subspace h, number of inter- and intra-class neighbors k_{inter}, k_{intra}, convergence error ε, max iteration time T_{max}, and two user-defined parameters α, σ.

Output: class-specific feature mappings $\{\boldsymbol{W}_i^k\}_{i,k=1,1}^{c,t}$.

For $k = 1, 2, \ldots, t$, **repeat**

Step 1 (**Initialization**)

1.1. Set $\boldsymbol{\theta}^{k,0} = \boldsymbol{I}_{d\times h}$;

1.2. For $i = 1, 2, \ldots, c$, repeat

Set $\boldsymbol{W}_i^{k,0} = randn(d, h)$;

Step 2 (**Preparation**)

2.1. For each local feature $\vec{x}_{ij}^k$, the similarities A_{ijp}, B_{ijq} between $\vec{x}_{ij}^k$ and its inter- and intra-class neighbors are calculated from Eqs. (1) and (2), respectively.

2.2. Compute auxiliary matrices $\boldsymbol{H}_{i,inter}$, $\boldsymbol{H}_{i,intra}$ from Eqs. (5) and (7), respectively.

Step 3 (**Optimization**)

For $r = 1, 2, \ldots, T_{max}$, repeat

3.1. With $\boldsymbol{\theta}^{k,r-1}$ fixed, $\{\boldsymbol{W}_i^k\}^r$ are updated using [32].

3.2. With $\{\boldsymbol{W}_i^k\}^r$ fixed, $\boldsymbol{\theta}^{k,r}$ is computed from Eq. (13).

3.3. Calculate the objective value L^r from Eq. (10) with $\{\boldsymbol{W}_i^k\}^r$ and $\boldsymbol{\theta}^{k,r}$.

3.4. If $r > 2$ and $|L^{r-1} - L^r| < \varepsilon$, go to Step 4.

Step 4 (**Output**)

Output class-specific feature mappings $\{\boldsymbol{W}_i^k\}_{i=1}^c$.

3.3. Recognition

Given a testing sample, we first partition it into t blocks and then extract low-level feature from each block. Class-specific feature mappings $\{\boldsymbol{W}_i^k\}_{i,k=1,1}^{c,t}$ obtained in the training stage are used to map low-level features into mid-level ones. Specifically, for each low-level feature $\vec{y}^k$ (extracted from training or testing sample), we first calculate the projection of each class-specific mapping as::

$$\vec{v}_i^k = (\boldsymbol{W}_i^k)^T\vec{y}^k, \quad i = 1, \ldots, c, \quad k = 1, \ldots, t, \tag{14}$$

where $\vec{v}_i^k$ is the feature projection of $\vec{y}^k$ under the i-th class-specific feature mapping.

These feature projections form our new feature representations for the corresponding facial regions. We then concatenate the projections of the same class and get \$c\$ holistic feature representations:

$$\vec{V}_i = \left[(\vec{v}_i^1)^T, \ldots, (\vec{v}_i^t)^T\right]^T, \quad i = 1, \ldots, c. \tag{15}$$

Indeed, an improved concatenated features can be obtained using the weighting schemes described in [13]:

$$\vec{V}_i^* = \left[(w_i^1 \cdot \vec{v}_i^1)^T, \ldots, (w_i^t \cdot \vec{v}_i^t)^T\right]^T, \quad i = 1, \ldots, c. \tag{16}$$

To determine parameters $\{w_i^k\}_{i,k=1,1}^{c,t}$, the mean optical strain [13] is calculated for each facial region to depict the motion intensity. To find out the general active regions, num_{most} regions with relatively large strain magnitudes are firstly recorded as candidates in each video. Their frequencies in each class are then calculated. Candidates with low frequencies will be further excluded, and the weights are computed from the remainders.

With $\{w_i^k\}_{i,k=1,1}^{c,t}$ directly used, active regions with higher weight assignment will be emphasized in the recognition. To keep a balance between active and inactive regions in the concatenated features, we can utilize weights in an inverse way, i.e. $\widehat{w}_i^k = 1 - w_i^k$. By doing this, more attention is taken to the inactive regions to eliminate the biases caused by the dynamic information gap between active and inactive regions. Both weighting schemes will be used and evaluated in experiments.

For recognition, we exactly follow the procedures described in [33], where holistic features (i.e. $\vec{V}_i$ or $\vec{V}_i^*$) of the same class (e.g. class i) were fed into a two-class SVM classifier, each SVM outputs a confidence value indicating the probability of a testing sample belonging to the considered micro-expression class (e.g. class i). And totally we can obtain \$c\$ SVM classifiers for recognition. The class with the highest SVM output is selected as our predicted label.

4. Experiments

In this section, two non-mobile and one mobile micro-expression datasets are used to evaluate the effectiveness of the proposed method.

The datasets, implementation details, and our experimental results are described as follows.

4.1. Experiments on non-mobile datasets

4.1.1. Non-mobile datasets

SMIC: The SMIC dataset [17] contains 164 micro-expression video clips. These video clips were recorded from 16 subjects and labeled into three different classes: positive (happy), negative (sad, fear and disgust) and surprise. Following [29,30], we used all the 164 samples for evaluation. To eliminate the spatial and temporal discrepancies, bicubic interpolation and temporal interpolation method (TIM) [34] were employed to normalize each video clip so that all the videos have a resolution of 150 × 120 pixels and 20 frames.

CASME2: The CASME2 dataset [19], which is an extension of the CASME dataset [18], consists of 26 subjects with 255 micro-expression video clips recorded by a 200 fps camera. These samples include seven classes: happiness, surprise, fear, sadness, disgust, repression and others. In our experiments, classes with few samples (i.e. fear and sadness) were not used for evaluation as [19] did. Thus we conducted experiments on the rest 246 samples from 5 classes. Similar to SMIC, all the video clips in this set were normalized to a uniform size of 150×120 pixels and 30 frames in the spatial and temporal dimensions.

4.1.2. Implementation details

Similar to [17,22,29], we used the leave-one-subject-out (LOSO) cross-validation to evaluate the proposed method, where samples from one certain subject were used as testing data, while the rest served as the training samples. This process repeated until all the subjects were met, and the mean recognition accuracy was used to measure the performance.

For the parameter setting, the four parameters σ, k_{inter}, k_{intra} and α used in our multi-task mid-level feature learning were set as 100, 15, 5 and 0.001, respectively. The dimension of the class-specific feature subspace h was selected as $0.6 \times d$. For computing local features on the SMIC dataset, video clips were spatially divided into 5×1 blocks, while the division grid on CASME2 was selected as 5×4. The parameter num_{most} used on SMIC and CASME2 was set as 2 and 9, respectively. Three kinds of features LBP-TOP, LBP-MOP and LBP-MOP*[2] were used as our low-level features. The radii (R_x, R_y, R_t) in axes X, Y and T used in LBP-TOP and LBP-MOP (LBP-MOP*) were set as (4, 4, 2) and (2, 2, 2) on SMIC, (3, 3, 3) and (1, 1, 3) on CASME2, respectively.

4.1.3. Results and analysis

4.1.3.1. Comparison with the original low-level features. To validate the effectiveness of the proposed mid-level feature learning method, we conduct our experiments on both SMIC and CASME2 datasets. We compare the performance of the original low-level features and the mid-level features learned by our multi-task mid-level feature learning (MMFL) method. Both features were evaluated using two different weighting schemes described in Section 3.3 (denoted as w_{active} and $w_{balance}$). For w_{active}, local features are concatenated with weights proportional to the activeness of facial regions, while weights of $w_{balance}$ are inverse (i.e. $\widehat{w}_i^k = 1 - w_i^k$). The results without weighting scheme are also reported, and we denoted it as w_{no}. By examining the comparison results presented in Tables 1 and 2, we can obtain the following observations:

- For the case of w_{no}, the results obtained by model MMFL are higher than that of original features on both SMIC and CASME2 datasets. It indicates that the proposed multi-task mid-level feature learning method can enhance the discrimination ability of the original low-level features and thus lead to better recognition performance.
- With both weighting schemes used, different demands of weighting schemes can be observed for SMIC and CASME2 datasets. As can be seen from Tables 1 and 2, no matter whether multi-task mid-level feature learning is utilized or not, results of using w_{active} are higher than that of using $w_{balance}$ on SMIC, while on CASME2 the methods using $w_{balance}$ always perform better. This reflects the inherent difference between both datasets. SMIC set is more dependent on the active regions (e.g. eyes and mouth), in which relatively large and discriminative motions can be captured and used to recognize the positive (happy) and surprise emotions, while the remainder (negative) can be filtered out from them. However, in CASME2, more diverse and detailed expressions are included. To distinguish them, more attention should be paid to the subtle discrepancies between different micro-expressions, especially for the ones from inactive regions. Due to that, compared with w_{active}, $w_{balance}$ is more adequate for the recognition of the CASME2 dataset and can lead to better recognition performance.
- It is noticed that, even some original features (LBP-MOP and LBP-MOP*) on SMIC get slightly higher results than the ones of MMFL when w_{active} is utilized, their results using $w_{balance}$ are generally lower than the ones of MMFL on both datasets. It indicates that, with plentiful dynamics in active regions, the original low-level features can depict some discriminative patterns as MMFL does, and benefits from w_{active} on the SMIC dataset. But for inactive regions with insufficient information, the original features are less robust than the ones obtained by our MMFL, which shows the effectiveness of the proposed multi-task mid-level feature learning.

Table 1
The result (%) of MMFL and the original features (Original) on the SMIC dataset.

Method	Weighting Scheme	LBP-TOP	LBP-MOP	LBP-MOP*
Original	w_{no}	45.09	44.22	38.69
	Using $w_{balance}$	47.45	43.16	4.48
	Using w_{active}	51.73	**63.95**	**63.72**
MMFL	w_{no}	**49.02**	**53.81**	**42.64**
	Using $w_{balance}$	51.03	47.09	47.58
	Using w_{active}	**55.19**	**62.33**	**63.15**

Table 2
The result (%) of MMFL and the original features (Original) on the CASME2 dataset.

Method	Weighting Scheme	LBP-TOP	LBP-MOP	LBP-MOP*
Original	w_{no}	48.90	54.24	55.20
	Using $w_{balance}$	30.15	30.83	29.77
	Using w_{active}	25.56	22.85	25.46
MMFL	w_{no}	**53.33**	**57.59**	**58.09**
	Using $w_{balance}$	**54.60**	**57.61**	**59.81**
	Using w_{active}	41.34	39.73	43.21

4.1.3.2. Comparison with traditional supervised subspace learning. To further validate the effectiveness of the proposed multi-task mid-level feature learning, we compared our method with some traditional supervised subspace learning methods, including LDA. A variant of MMFL without regularization term R was also implemented, and denoted as NR. The comparison results are presented in Tables 3 and 4. From these tables, we can obtain the following observations:

- For the case of w_{no}, LDA performs poorly on both datasets and get worse results than the original ones presented in Tables 1 and 2. Such result indicates that LDA is inadequate to tackle problems with insufficient information. Different from LDA, NR and MMFL, which

[2] LBP-MOP* is an extension of LBP-MOP based on [13].

Table 3
The result (%) of LDA, NR and MMFL on the SMIC dataset.

Method	Weighting Scheme	LBP-TOP	LBP-MOP	LBP-MOP*
LDA	w_{no}	37.65	35.26	36.28
	Using $w_{balance}$	43.48	43.22	57.39
	Using w_{active}	49.67	49.36	39.72
NR	w_{no}	33.57	41.87	41.22
	Using $w_{balance}$	41.77	43.11	44.70
	Using w_{active}	41.63	53.23	58.68
MMFL	w_{no}	**49.02**	**53.81**	**42.64**
	Using $w_{balance}$	51.03	47.09	47.58
	Using w_{active}	**55.19**	**62.33**	**63.15**

Table 4
The result (%) of LDA, NR and MMFL on the CASME2 dataset.

Method	Weighting Scheme	LBP-TOP	LBP-MOP	LBP-MOP*
LDA	w_{no}	42.63	43.39	48.19
	Using $w_{balance}$	38.16	35.90	36.69
	Using w_{active}	42.61	42.47	41.50
NR	w_{no}	48.73	56.27	57.03
	Using $w_{balance}$	42.24	56.34	56.97
	Using w_{active}	42.09	40.76	44.87
MMFL	w_{no}	**53.33**	**57.59**	**58.09**
	Using $w_{balance}$	**54.60**	**57.61**	**59.81**
	Using w_{active}	41.34	39.73	43.21

decompose problem into numerous two-class ones, can generate features with higher generalization ability and get better performance. Furthermore, by sharing common structure between different mappings, MMFL can get further improved and obtain better recognition results than the NR method.

- When using w_{active} on SMIC and $w_{balance}$ on CASME2, we can find that MMFL can always get improvement, and its results are higher than the ones of LDA and NR methods. This result demonstrates that, with higher generalization ability, feature obtained by MMFL can better represent the dynamic patterns of different regions and further benefit from weighting schemes in the feature concatenation.
- It is noticed that, no matter which supervised subspace learning method is utilized, the performance of using w_{active} is generally higher than the one of using $w_{balance}$ on SMIC, while on CASME2 the results (except LDA) are opposite. Such phenomenon once again indicates the inherent difference between SMIC and CASME2, and the outlier further shows the poor generalization ability of features extracted by LDA, especially for the ones from inactive regions.

4.1.3.3. Comparison with the state-of-the-art methods. Here, we compare our MMFL model with several state-of-the-art micro-expression recognition approaches including OS (optical strain) [23], LBP-SIP+Gp and LBP-TOP+Gp [22](Gp denotes Gaussian pyramid used in feature extraction), RW (monogenic resize wavelet) [28], STM (selective transfer machine) [35], SS (sparse sampling) [36], STLBP-IP (spatiotemporal local binary pattern with integral projection) [29] and STCLQP (spatiotemporal completed local quantization pattern) [30]. The comparison results are listed in Tables 5 and 6. From these tables, we can obtain the following three observations:

- From Tables 5 and 6, we can observe that the results of MMFL are higher than the ones of OS, RW, LBP-SIP+Gp and LBP-TOP+Gp on both datasets. It shows that, compared with the basic spatiotemporal features, MMFL can reveal more relevant information, and more discriminative features can be obtained for better recognition performance.
- It is noticed that, by eliminating redundant frames in video clips and reducing imbalanced sample distribution of different subjects, SS and STM can generally get better performance than the basic spatiotemporal features. However, they are still generally lower than the ones of MMFL. The reason is that, without any processing, an amount of irrelevant and noisy information is involved in the basic spatiotemporal features. Based on them, SS and STM can only gain limited improvement.
- As can be seen from Tables 5 and 6, MMFL cannot always outperform STLBP-IP and STCLQP. However, compared with our MMFL, STCLQP and STLBP-IP are highly dependent on the initialization of clustering centers and the neutral frame selection, which makes them less stable in real-world applications.

Table 5
The result (%) of MMFL and the state-of-the-art methods on the SMIC dataset.

Method		ACC
State-of-the-art	OS [23]	53.56
	RW [28]	34.00
	STM [35]	44.00
	SS [36]	58.00
	STLBP-IP [29]	57.93
	STCLQP [30]	**64.02**
MMFL + w_{active}	LBP-TOP	55.19
	LBP-MOP	**62.33**
	LBP-MOP*	**63.15**

Table 6
The result (%) of MMFL and the state-of-the-art methods on the CASME2 dataset.

Method		ACC
State-of-the-art	LBP-SIP+Gp [22]	38.46
	LBP-TOP+Gp [22]	37.25
	RW [28]	46.00
	STM [35]	44.00
	SS [36]	49.00
	STLBP-IP [29]	**59.51**
	STCLQP [30]	**58.39**
MMFL + $w_{balance}$	LBP-TOP	54.60
	LBP-MOP	**57.61**
	LBP-MOP*	**59.81**

4.1.3.4. Influence of parameter α. Here, we evaluated the influence of α in our model by varying it from 10^{-5} to 10^{-1} while fixing the other parameters. The experimental results on both datasets are presented in Figs. 3 and 4.

From these figures, we can observe that in most cases, results look like a para-curve and have peaks when α is near 0.001. The reason could be that the larger α is, the more similar the class-specific feature mappings will be. So when a high α is used, the discrepancies of class-specific feature mappings will decrease, as well as the discrimination ability of the learned features. Likewise, when α is too small, the learnings of different feature mappings are poorly linked. Less common information can be mined for boosting our feature learning, which leads to less discriminative features and poorer recognition performance. To obtain better recognition results, a proper α with intermediate value 0.001 is selected in this paper.

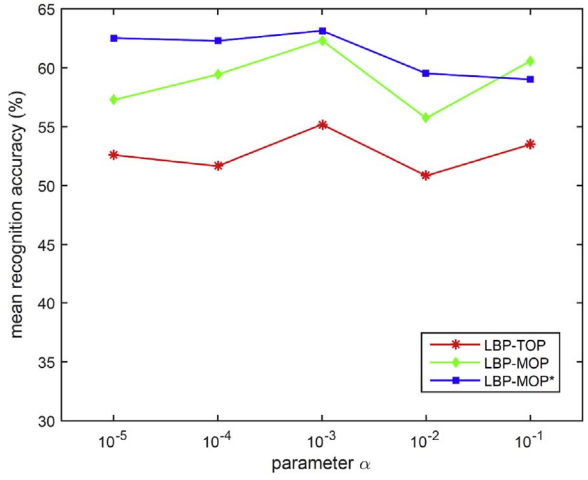

Fig. 3. Parameter analysis of α on SMIC when w_{active} is used.

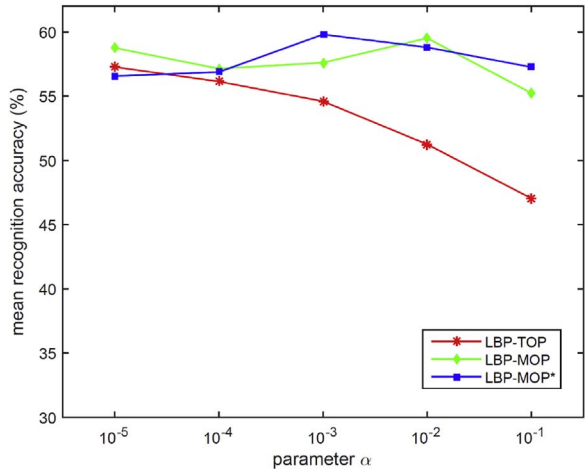

Fig. 4. Parameter analysis of α on CASME2 when $w_{balance}$ is used.

4.2. Experiments on mobile dataset

4.2.1. Mobile dataset

Nowadays, with the wide use of slow-motion mode, some mobile devices (e.g. iPhone and iPad) can record video clips in a high sampling rate (e.g. 120 fps and 240 fps), so they can serve as another source to collect micro-expression data. Indeed, mobile devices may not produce videos of the same quality with conventional cameras, but they are more widely used in daily life and can capture sufficient detailed facial movements for micro-expression recognition.

So far, there is no available mobile micro-expression dataset. To construct a mobile dataset, 17 subjects (14 male and 3 female) were invited to mimic 6 micro-expressions (i.e. happiness, surprise, fear, sadness, disgust and anger). An iPad Air 2 with slow-motion mode (120 fps) was fixed indoors to record the facial movements. Unlike that on the SMIC and CASME2 datasets, to simulate the camerawork in real daily life, we do not explicitly control the filming illumination, which makes the dataset more challenging. In total, we have 306 micro-expression video clips. We call this new collected dataset as *mobileDB*. Similar to SMIC and CASME2, each video clip in this dataset starts from a relatively neutral frame and ends when facial expression turns back to relatively neutral. These samples were then normalized to a uniform size (150×150 pixels and 30 frames) for evaluation. Some examples of the mobileDB dataset can be found in Fig. 5.

4.2.2. Implementation details

Similar to SMIC and CASME2, we extract the LBP-TOP, LBP-MOP and LBP-MOP* features to represent each sample. The parameters (R_x, R_y, R_t) for computing these features were set as (2, 2, 1), (2, 2, 2) and (2, 2, 2), respectively. Other parameters were kept the same as that used in the CASME2 dataset.

4.2.3. Results and analysis

To evaluate the proposed method, three different experiments were conducted. The detailed results are presented in Tables 7 and 8 and Fig. 6, respectively. From them, we can obtain some analogous observations.

4.2.3.1. Comparison with the original low-level features. From Table 7, we can find that the results of MMFL are higher than the ones of the original low-level features when w_{no} is utilized. This result is the same as the ones on SMIC and CASME2, indicating that the proposed multi-task mid-level feature learning method can enhance the discrimination ability of the low-level features and thus lead to better recognition performance. Moreover, similar to CASME2, the results of using $w_{balance}$ are higher than the ones of using w_{active} on both original feature and the MMFL. It once again indicates that $w_{balance}$, which further balances the relationship between active and inactive regions, is more adequate to tackle problems with diverse expressions. Furthermore, with more discriminative information extracted from inactive regions, MMFL using $w_{balance}$ can always get improvement, while the original features cannot, which further shows its effectiveness in feature learning with insufficient information.

4.2.3.2. Comparison with traditional supervised subspace learning. As can be seen from Table 8, results of MMFL using w_{no} are higher than the ones of LDA and NR methods. Moreover, different from LDA and NR in which some features get worse results than the original ones, MMFL can always get better performance. It shows that, by decomposing problem into numerous two-class ones and sharing common information among them, MMFL can generate features with higher generalization ability, which can further benefit from $w_{balance}$ and lead to better recognition results than the ones of LDA and NR methods.

4.2.3.3. Influence of parameter α. Similar to the results presented in Section 4.1.3.4, the results presented in Fig. 6 also look like a para-curve and have peaks when α is near 0.001. It once again indicates that, α with too large or small value will result in nearly homology or independence of different mappings, and lead to poor performance. To keep a balance, α with intermediate value should be utilized.

5. Conclusion

To address the micro-expression recognition problem, a multi-task mid-level feature learning method is proposed in this paper. By learning numerous class-specific feature mappings simultaneously, the potential common information among them can be mined and supplied for each individual feature mapping learning. With more available information, features with better discrimination and generalization abilities can be obtained for recognition. Moreover, by utilizing weighing schemes, concatenated features can get further improvement. Experimental results on two widely used non-mobile micro-expression datasets and one mobile micro-expression set demonstrate the effectiveness of the proposed method.

Acknowledgment

This work was supported partially by the National Key Research

Fig. 5. Some examples of the mobileDB dataset.

Table 7
The result (%) of MMFL and the original features (Original) on the mobileDB dataset.

Method	Weighting Scheme	LBP-TOP	LBP-MOP	LBP-MOP*
Original	w_{no}	38.24	40.20	42.48
	Using $w_{balance}$	39.22	39.87	41.50
	Using w_{active}	26.47	26.80	30.72
MMFL	w_{no}	**40.85**	**41.50**	**47.71**
	Using $w_{balance}$	**43.14**	**43.79**	**48.04**
	Using w_{active}	28.43	29.74	37.25

Table 8
The result (%) of LDA, NR and MMFL on the mobileDB dataset.

Method	Weighting Scheme	LBP-TOP	LBP-MOP	LBP-MOP*
LDA	w_{no}	19.93	30.72	22.88
	Using $w_{balance}$	21.90	19.93	19.93
	Using w_{active}	18.63	19.93	18.30
NR	w_{no}	32.35	38.89	46.08
	Using $w_{balance}$	26.14	38.56	45.42
	Using w_{active}	18.63	28.76	37.91
MMFL	w_{no}	**40.85**	**41.50**	**47.71**
	Using $w_{balance}$	**43.14**	**43.79**	**48.04**
	Using w_{active}	28.43	29.74	37.25

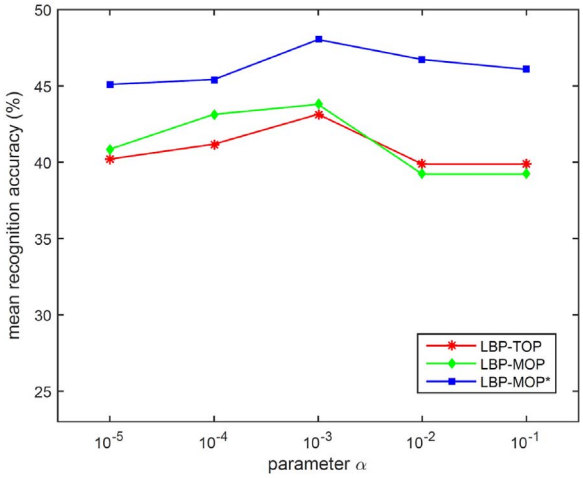

Fig. 6. Parameter analysis of α on mobileDB when $w_{balance}$ is used.

and Development Program of China (2016YFB1001002, 2016YFB1001003), NSFC (No. 61573387,61472456, 61522115, 61661130157, 61628212), Guangdong Natural Science Funds for Distinguished Young Scholar under Grant S2013050014265, the GuangDong Program (No. 2015B010105005), the Guangdong Science and Technology Planning Project (No. 2016A010102012, 2014B010118003), and Guangdong Program for Support of Top-notch Young Professionals (No. 2014TQ01X779).

References

[1] C. Shan, S. Gong, P.W. McOwan, Facial expression recognition based on local binary patterns: a comprehensive study, Image Vis. Comput. 27 (6) (2009) 803–816.
[2] J. Edwards, H.J. Jackson, P.E. Pattison, Emotion recognition via facial expression and affective prosody in schizophrenia: a methodological review, Clin. Psychol. Rev. 22 (6) (2002) 789–832.
[3] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C.M. Lee, A. Kazemzadeh, S. Lee, U. Neumann, S. Narayanan, Analysis of emotion recognition using facial expressions, speech and multimodal information, in: Proceedings of the 6th International Conference on Multimodal interfaces, ACM, Paris, France, 2004, pp. 205–211.
[4] E.A. Haggard, K.S. Isaacs, Micromomentary facial expressions as indicators of ego mechanisms in psychotherapy, in: Methods of Research in Psychotherapy, Springer, New York, 1966, pp. 154–165.
[5] P. Ekman, W.V. Friesen, Nonverbal leakage and clues to deception, Psychiatry 32 (1) (1969) 88–106.
[6] P. Ekman, M. O'Sullivan, Who can catch a liar?, Am. Psychol. 46 (9) (1991) 913.
[7] M.G. Frank, P. Ekman, The ability to detect deceit generalizes across different types of high-stake lies, J. Personality Soc. Psychol. 72 (6) (1997) 1429.
[8] P. Ekman, Darwin, deception, and facial expression, Ann. N.Y. Acad. Sci. 1000 (1) (2003) 205–221.
[9] W.-J. Yan, Q. Wu, J. Liang, Y.-H. Chen, X. Fu, How fast are the leaked facial expressions: the duration of micro-expressions, J. Nonverbal Behav. 37 (4) (2013) 217–230.
[10] D. Matsumoto, H.S. Hwang, Evidence for training the ability to read microexpressions of emotion, Motiv. Emot. 35 (2) (2011) 181–191.
[11] P. Ekman, Microexpression Training Tool, University of California, San Francisco, 2002.
[12] M.G. Frank, C.J. Maccario, V. Govindaraju, Behavior and Security, Protecting Airline Passengers in the Age of Terrorism, Greenwood Pub Group, Santa Barbara, California, 2009, pp. 86–106.
[13] S.-T. Liong, J. See, R.C.-W. Phan, A.C. Le Ngo, Y.-H. Oh, K. Wong, Subtle expression recognition using optical strain weighted features, in: Computer Vision—ACCV 2014 Workshops, Springer, Singapore, 2014, pp. 644–657.
[14] S.-J. Wang, W.-J. Yan, X. Li, G. Zhao, C.-G. Zhou, X. Fu, M. Yang, J. Tao, Micro-expression recognition using color spaces, IEEE Trans. Image Process. 24 (12) (2015) 6034–6047.
[15] Y.-J. Liu, J.-K. Zhang, W.-J. Yan, S.-J. Wang, G. Zhao, X. Fu, A main directional mean optical flow feature for spontaneous micro-expression recognition, IEEE Trans. Affect. Comput. 7 (4) (2016) 299–310.
[16] J. Lu, Y.-P. Tan, G. Wang, Discriminative multimanifold analysis for face recognition from a single training sample per person, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 39–51.
[17] X. Li, T. Pfister, X. Huang, G. Zhao, M. Pietikainen, A spontaneous micro-expression database: inducement, collection and baseline, in: IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, 2013, pp. 1–6.
[18] W.-J. Yan, Q. Wu, Y.-J. Liu, S.-J. Wang, X. Fu, Casme database: A dataset of spontaneous micro-expressions collected from neutralized faces, in: IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, 2013, pp. 1–7.
[19] W.-J. Yan, X. Li, S.-J. Wang, G. Zhao, Y.-J. Liu, Y.-H. Chen, X. Fu, Casme II: an improved spontaneous micro-expression database and the baseline evaluation, PloS One 9 (1) (2014).
[20] G. Zhao, M. Pietikainen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 915–928.
[21] Y. Wang, J. See, R.C.-W. Phan, Y.-H. Oh, Lbp with six intersection points: Reducing redundant information in lbp-top for micro-expression recognition, in: Computer Vision—ACCV 2014, Springer, Singapore, 2014, pp. 525–537.
[22] Y. Wang, J. See, R.C.-W. Phan, Y.-H. Oh, Efficient spatio-temporal local binary patterns for spontaneous facial micro-expression recognition, PloS One 10 (5) (2015).
[23] S.-T. Liong, R.C.-W. Phan, J. See, Y.-H. Oh, K. Wong, Optical strain based recognition of subtle emotions, in: International Symposium on Intelligent Signal Processing and Communication Systems, 2014, pp. 180–184.
[24] S.-J. Wang, W.-J. Yan, G. Zhao, X. Fu, C.-G. Zhou, Micro-expression recognition using robust principal component analysis and local spatiotemporal directional features, in: Computer Vision—ECCV 2014 Workshops, Springer, Zurich, 2014, pp. 325–338.
[25] S.-J. Wang, W.-J. Yan, X. Li, G. Zhao, X. Fu, Micro-expression recognition using

dynamic textures on tensor independent color space, in: International Conference on Pattern Recognition, 2014, pp. 4678–4683.
[26] J.A. Ruiz-Hernandez, M. Pietikainen, Encoding local binary patterns using the re-parametrization of the second order Gaussian jet, in: IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, 2013, pp. 1–6.
[27] Z. Lu, Z. Luo, H. Zheng, J. Chen, W. Li, A Delaunay-based temporal coding model for micro-expression recognition, in: Computer Vision—ACCV 2014 Workshops, Springer, Singapore, 2014, pp. 698–711.
[28] Y.-H. Oh, A.C. Le Ngo, J. See, S.-T. Liong, R.C.-W. Phan, H.-C. Ling, Monogenic Riesz wavelet representation for micro-expression recognition, in: IEEE International Conference on Digital Signal Processing, 2015, pp. 1237–1241.
[29] X. Huang, S.-J. Wang, G. Zhao, M. Piteikainen, Facial micro-expression recognition using spatiotemporal local binary pattern with integral projection, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 1–9.
[30] X. Huang, G. Zhao, X. Hong, W. Zheng, M. Pietikainen, Spontaneous facial micro-expression analysis using spatiotemporal completed local quantized patterns, Neurocomputing 175 (2016) 564–578.
[31] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 27:1–27:27.
[32] Z. Wen, W. Yin, A feasible method for optimization with orthogonality constraints, Math. Program. 142 (1–2) (2013) 397–434.
[33] L. Zhong, Q. Liu, P. Yang, J. Huang, D.N. Metaxas, Learning multiscale active facial patches for expression analysis, IEEE Trans. Cybern. 45 (8) (2015) 1499–1510.
[34] Z. Zhou, G. Zhao, Y. Guo, M. Pietikainen, An image-based visual speech animation system, IEEE Trans. Circuits Syst. Video Technol. 22 (10) (2012) 1420–1432.
[35] A.C. Le Ngo, R.C.-W. Phan, J. See, Spontaneous subtle expression recognition: imbalanced databases and solutions, in: Computer Vision—ACCV 2014, Springer, Singapore, 2014, pp. 33–48.
[36] A.C. Le Ngo, J. See, R.C.-W. Phan, Sparsity in dynamics of spontaneous subtle emotions: analysis & application, IEEE Trans. Affect. Comput. (2016). http://dx.doi.org/10.1109/TAFFC.2016.2523996.
[37] J.-F. Hu W.-S. Zheng J. Lai J. Zhang Jointly learning heterogeneous features for RGB-D activity recognition IEEE Trans. Pattern Anal. Mach. Intell. (Accepted)

Jiachi He received the B.S. degree from Jinan University, Guangzhou, China, in 2014. He is currently pursuing the M.S. degree from Sun Yat-Sen University, Guangzhou, China. His research interests include face and facial expression recognition.

Jian-Fang Hu received the PhD and B.S. degrees from the School of Mathematics, Sun Yat-sen University, Guangzhou, China, in 2016 and 2010, respectively. His research interests include human-object interaction modeling, 3D face modeling, and RGB-D activity recognition. He has published several scientific papers in the international journals and conferences including IEEE TPAMI, IEEE TCSVT, PR, ICCV, CVPR, and ECCV.

Xi Lu received the B.S. Degree from Jilin University. He is currently pursuing the M.S. degree under the supervision of Dr. W.-S Zheng. He is interested in computer vision and machine learning.

Wei-Shi Zheng received the Ph.D. degree in Applied Mathematics from Sun Yat-Sen University, in 2008. He is now a Professor at Sun Yat-sen University. He had been a Postdoctoral Researcher on the EU FP7 SAMURAI Project at Queen Mary University of London and an Associate Professor at Sun Yat-sen University after that. He has now published more than 80 papers, including more than 50 publications in main journals (TPAMI, TNN, TIP, TSMC-B, PR) and top conferences (ICCV, CVPR, IJCAI, AAAI). He has joined the organisation of four tutorial presentations in ACCV 2012, ICPR 2012, ICCV 2013 and CVPR 2015 along with other colleagues. His research interests include person/object association and activity understanding in visual surveillance. He has joined Microsoft Research Asia Young Faculty Visiting Programme. He is a Recipient of Excellent Young Scientists Fund of the National Natural Science Foundation Of China, and a recipient of Royal Society-Newton Advanced Fellowship.