Learning Expressionlets via Universal Manifold Model for Dynamic Facial Expression Recognition

Mengyi Liu, *Student Member, IEEE*, Shiguang Shan, *Senior Member, IEEE*,
Ruiping Wang, *Member, IEEE*, and Xilin Chen, *Fellow, IEEE*

***Abstract*— Facial expression is a temporally dynamic event which can be decomposed into a set of muscle motions occurring in different facial regions over various time intervals. For dynamic expression recognition, two key issues, temporal alignment and semantics-aware dynamic representation, must be taken into account. In this paper, we attempt to solve both problems via manifold modeling of videos based on a novel mid-level representation, i.e., expressionlet. Specifically, our method contains three key stages: 1) each expression video clip is characterized as a spatial-temporal manifold (STM) formed by dense low-level features; 2) a universal manifold model (UMM) is learned over all low-level features and represented as a set of local modes to statistically unify all the STMs; and 3) the local modes on each STM can be instantiated by fitting to the UMM, and the corresponding expressionlet is constructed by modeling the variations in each local mode. With the above strategy, expression videos are naturally aligned both spatially and temporally. To enhance the discriminative power, the expressionlet-based STM representation is further processed with discriminant embedding. Our method is evaluated on four public expression databases, CK+, MMI, Oulu-CASIA, and FERA. In all cases, our method outperforms the known state of the art by a large margin.**

***Index Terms*— Facial expression recognition, universal manifold model, Riemannian manifold, discriminant Learning, expressionlets.**

I. Introduction

AUTOMATIC facial expression recognition plays an important role in various applications, such as Human-Computer Interaction (HCI) and diagnosing mental disorders. Early research mostly focused on expression analysis from static facial images [1]. However, as facial expression can be better described as the sequential variation in a dynamic process, recognizing facial expression from video is more natural and has been proved to be more effective in recent research works [2]–[6].

Among these video-based facial expression recognition methods, one of the main concerns is how to effectively encode the dynamic information in videos. Currently, the mainstream approaches to dynamic representation are based on local spatial-temporal features like LBP-TOP (local binary patterns on three orthogonal planes) [2] and HOG 3D (histogram of oriented gradients on spatio-temporal dimensions) [7]. These local descriptors extracted in local cuboid are then pooled over the whole video or some hand-crafted segments, to obtain a representation with certain length independent of time resolution. As the low-level features possess the property of repeatability, integrating them by pooling leads to robustness to intra-class variations and deformations of different expression styles. However, this kind of technique lacks of consideration of two important issues: 1) **Temporal alignment**. Expressions are inherently dynamic events consisting of onset, apex, and offset phases. Intuitively, the recognition should conduct matching among corresponding phases, which thus requires globally temporal alignment among different sequences. The rigid pooling has inevitably dropped those sequential relations and temporal correspondences. 2) **Semantics-aware dynamic representation**. Each expression can be decomposed into a group of semantic action units, which exhibit in different facial regions with varying sizes and last for different lengths of time. Since the manually designed cuboids can only capture low-level information short of representative and discriminative ability, they are incapable of modeling the expression dynamic in higher semantic level.

In this paper, we attempt to address both issues via spatial-temporal manifold modeling based on a set of mid-level representations, i.e. **expressionlets**. The proposed mid-level expressionlet is a kind of modeling that aims to characterize the variations among a group of low-level features as shown in Figure 1. The notation "-let" means that it serves as a local (both spatially and temporally) dynamic component within a whole expression process, which shares similar spirit with "motionlet" [8] in action recognition community. Thus expressionlet bridges the gap between low-level features and high-level semantics desirably. Specifically, as shown in Figure 2, given an individual video clip, we first characterize it as a Spatial-Temporal Manifold (STM) spanned by its low-level features. To conduct spatial-temporal alignment among STMs,

Manuscript received March 31, 2016; revised August 8, 2016; accepted September 16, 2016. Date of publication October 5, 2016; date of current version October 28, 2016. This work was supported in part by the 973 Program under Contract 2015CB351802, in part by the Natural Science Foundation of China under Contract 61390511, Contract 61379083, and Contract 61272321, in part by the Strategic Priority Research Program of the CAS under Grant XDB02070004, in part by the Youth Innovation Promotion Association CAS under Grant 2015085, and in part by the FiDiPro Program of Tekes. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Peter Tay. *(Corresponding author: Shiguang Shan.)*

M. Liu, R. Wang, and X. Chen are with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: mengyi.liu@vipl.ict.ac.cn; wangruiping@ict.ac.cn; xlchen@ict.ac.cn).

S. Shan is with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China, and also with the CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai 200031, China (e-mail: sgshan@ict.ac.cn).

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

Digital Object Identifier 10.1109/TIP.2016.2615424

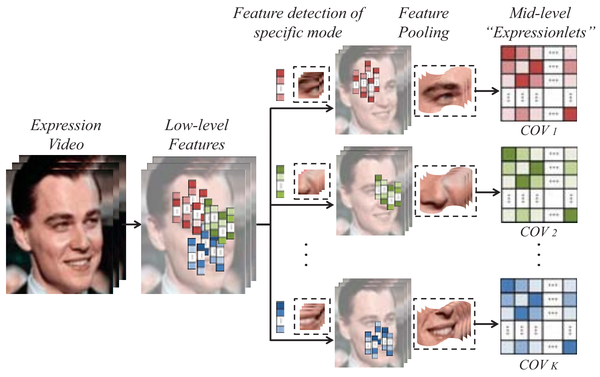


Fig. 1. A schematic illustration of constructing the mid-level representation – the proposed "expressionlets" ("COV" is short for "covariance matrix"). Each strip stands for a local feature, and the K feature modes (similar to codewords) are pre-learned and modeled via GMM.

we build a Universal Manifold Model (UMM), and represent it by a number of universal local ST modes, which can be learned by EM-like methods among the entire collection of low-level features. By fitting to UMM, the local modes on each STM can be instantiated respectively and all of the different STMs are inherently and mutually well-aligned to UMM via these corresponding modes. Finally, our expressionlet is constructed by modeling each local mode on STMs. To capture and characterize the correlations and variations among low-level features within each mode, the expressionlet comes in the form of covariance matrix of the feature set in a statistical manner, which also makes it robust to local misalignment [9]–[11].

To further enhance the discriminative ability of expressionlet, we perform a discriminant learning with these mid-level representations on all of the STMs. By considering the "margin" among corresponding expressionlets, we exploit a graph-embedding [12], [13] method by constructing partially connected graphs to keep the links between expressionlets with the same semantics. In the end, the embedded features are correspondingly concatenated into a long vector as the final manifold (video) representation for classification. Hence, the proposed expressionlet has the following characteristics: 1) **Flexible spatial-temporal range**. i.e. varying sizes of spatial regions and temporal durations. 2) **Variation modeling**. It encodes the local variations caused by expression using a covariance matrix. 3) **Discriminative ability**. It is descriptive and contains category information for recognition.

Preliminary results of the method have been published in [14]. Compared with the conference version, this paper has made three major extensions. First, we generalize the framework to be compatible for various low-level 2D/3D descriptors to construct mid-level expressionlet. Second, we provide a more detailed comparison and discussion regarding different strategies for UMM learning, including the alignment manners of local modes in UMM training stage and the low-level feature assignment manners in UMM fitting stage. Third, more extensive experiments are carried out to evaluate each component in the method and compare with other state-of-the-art algorithms.

The rest of the paper is organized as follows: Section II briefly reviews the previous related work for dynamic facial expression recognition. Section III introduces the Universal Manifold Model, i.e. a statistical model for spatial-temporal alignment among different expression manifolds (videos). Section IV presents the mid-level expressionlet learning based on UMM and conducts detailed discussions with other related works. In Section V, we provide comprehensive evaluations of the whole framework as well as each of the building block. Experiments are conducted on four public expression databases and extensively compared with the state-of-the-art methods. Finally, we conclude the work and discuss possible future efforts in Section VI.

II. Related Works

In the past several decades, facial expression recognition based on static images had aroused lots of interests among researchers. For facial feature representation, typical image descriptors including Local Binary Pattern (LBP) [15], Local Gabor Binary Pattern (LGBP) [16], Histogram of Oriented Gradient (HOG) [17], and Scale Invariant Feature Transform (SIFT) [18] have been successfully applied in this domain. Lucey et al. [19] also applied Active Appearance Model (AAM) to encode both shape (facial landmarks) and appearance variations. A comprehensive survey of some of these techniques can be found in [1] and [20].

However, as facial expressions are more naturally viewed as dynamic events involving facial motions over a time interval, recently, strong interest in modeling the temporal dynamics of facial expressions in video clips has evolved. The psychological experiments conducted in [21] have provided evidence that facial dynamics modeling is crucial for interpreting and discriminating facial expressions. Generally, the temporal modeling manners can be categorized into two groups: hard-coded and learning-based. In this paper, we review some related works of dynamic facial expression recognition based on the two schemes mentioned above.

The hard-coded modeling scheme encodes the variations among several successive frames using predefined computations. For example, optical flow is calculated between consecutive frames and has been applied in some early works for expression recognition [22], [23]. Koelstra et al. [24] used Motion History Images (MHI) to compress the motions over several frames into a single image by layering the pixel differences between consecutive frames. Another kind of typical implementation is designing spatial-temporal local descriptors to capture the dynamic information. For instance, Yang et al. [3] designed dynamic binary patterns mapping for temporally clustered Haar-like features and adopted boosting classifiers for expression recognition. Zhao et al. [5] encoded spatial-temporal information in image volumes using LBP-TOP [2] and employed SVM and sparse representation classifier for recognition. Hayat et al. [25] evaluated various dynamic descriptors including HOG/HOF [26], HOG3D [7], and 3D SIFT [27] using bag of features framework for video-based facial expression recognition. All these methods benefit from the low computational cost of local descriptors and also show favourable generalizations to different data sources and recognition tasks.

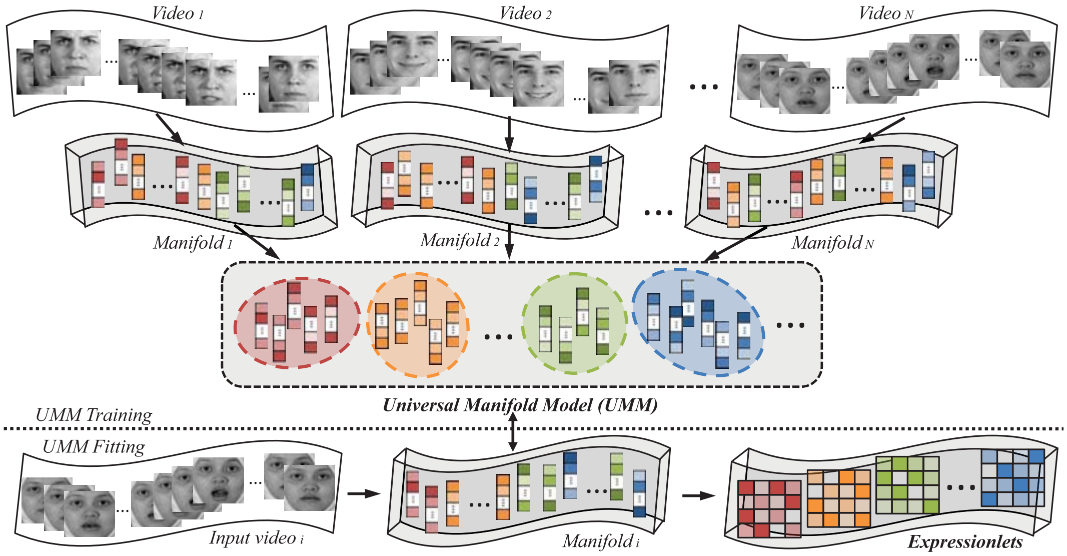


Fig. 2. The schema of the proposed method. Given an individual video clip, we intend to model it as a Spatial-Temporal Manifold (STM) spanned by local spatial-temporal features. To statistically unify and thus facilitate the alignment of STMs, we propose a Universal Manifold Model (UMM), represented as a number of universal local ST modes, which can be learned by EM-like methods. With UMM constructed, the local modes on each STM can be instantiated by fitting to UMM and thus aligned mutually, then the corresponding expressionlet is built to model the variations (via covariance matrix) in each local ST mode. Thus we obtain an expressionlet-based representation of STM. Please note that, for UMM training, we exploit both appearance and spatial-temporal location information of the local features in order to enforce some degree of locality both spatially and temporally.

To consider the specific characteristics of dynamic facial expressions, the learning-based modeling schemes attempt to explore the intrinsic correlations among facial variations using dynamic graphical models. Some representative works are briefly introduced as follows: Cohen et al. [28] used Tree-Augmented Naive Bayes (TAN) classifier to learn the dependencies among the facial motion features extracted from a continuous video. Shang and Chan [29] applied a non-parametric discriminant Hidden Markov Model (HMM) on the facial features tracked with Active Shape Model (ASM) to recognize dynamic expressions. Jain et al. [30] proposed a framework by modeling temporal variations within facial shapes using Latent-Dynamic Conditional Random Fields (LDCRFs), which obtains the entire video prediction and continuously frame labels at the same time. To further characterize the complex activities both spatially and temporally, Wang et al. [31] proposed Interval Temporal Bayesian Networks (ITBN) to represent the spatial dependencies among primary facial events and the large variety of time-constrained relations simultaneously. To summarize, the learning-based modeling can better reveal the intrinsic principles of the dynamic variations caused by facial expressions. However the construction and optimization of a such model requires lots of domain knowledge and high computational cost.

III. Universal Manifold Model (UMM)

A facial expression video depicts continuous shape or appearance variations and can be naturally modeled by a nonlinear manifold, on which each point corresponds to a certain local spatial-temporal pattern. For dynamic expression recognition, the main challenge is the large arbitrary inter-personal variance of expressing manners and execution rate for the same expression category, thus it is crucial to conduct both spatial and temporal alignment among different expression manifolds. In this section, we first introduce the manifold modeling of videos and then propose a statistic-based Universal Manifold Model (UMM) to achieve implicit alignment among different expression videos.

A. Spatial-Temporal Manifold

For clarification, we first present the spatial-temporal manifold (STM) for modeling each video clip. The STM is spanned by 3D (i.e. spatial-temporal) blocks densely sampled from the video volume, which cover a variety of local variations in both spatial and temporal space. Two kinds of common descriptors, i.e. SIFT and HOG, are employed for low-level feature extraction on each sampled block with the size of $w * h * l$, where w, h are the numbers of pixels on two spatial directions, and l is the number of frames. The extracted feature is denoted as a_{xyt}, where x, y, t are spatial-temporal index of the block on the STM.

To consider the manifold structure information, for all the blocks we augment the appearance features with their spatial-temporal coordinates, i.e. $f = \{a_{xyt}, x/w^*, y/h^*, t/l^*\}$, where a_{xyt} is the appearance feature of the block located at $\{x, y, t\}$, and w^*, h^*, l^* are the numbers of blocks on width, height and time length direction on the STM. An illustration of the local features is shown in Figure 3.

B. UMM Learning

1) Training Stage: Universal Manifold Model (UMM) is defined to statistically model the STMs from different people with different expressions. As a person-independent and expression-independent model, UMM facilitates the robust parameterized modeling of the STMs.

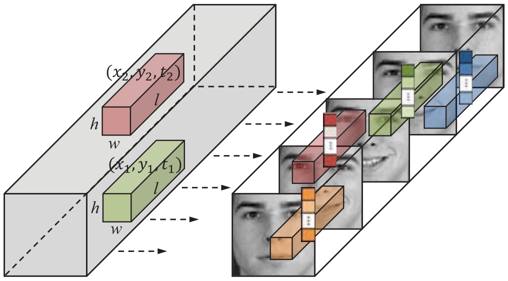


Fig. 3. An illustration of the spatial-temporal blocks for low-level feature extraction. The augmented features are then used to construct the STM.

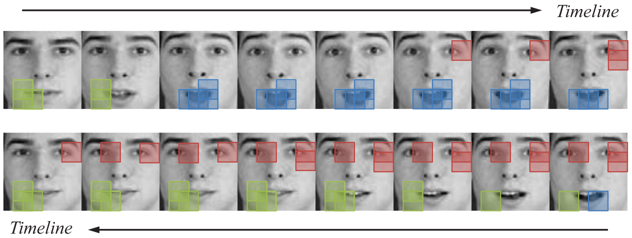


Fig. 4. Examples of typical local modes referring to the original spatial-temporal locations in videos. A local mode is consisted of a set of features with largest T probabilities to a certain Gaussian component on UMM. Note that different colors represent different local modes. **Best viewed in color.**

Algorithm 1 UMM Learning

```
Input:
    Unaligned STMs (represented by sets of low-level features): $\mathcal{F}^1, ..., \mathcal{F}^N$
Output:
    Mutually aligned STMs (represented by corresponding local modes instantiated by fitting to UMM): $\widetilde{\mathcal{F}}^1, ..., \widetilde{\mathcal{F}}^N$
    — Training —
Initialize UMM (GMM) parameter: $\Theta = \{(\omega_k, \mu_k, \sigma_k)\}$
Use EM algorithm to learn optimal UMM parameters:
    $\Theta^* = argmax_\Theta \sum_{i,b,k} \omega_k G(f_b^i|\mu_k, \sigma_k^2 I)$
    — Fitting —
for i:=1 to N do
    for k:=1 to K do
        Find top $T$ block features $F_k^i = \{f_{k_t}^i\}_{t=1}^T$ with the largest probabilities on $G_k$:
        $G(f_{k_t}^i|\mu_k^*, (\sigma_k^*)^2 I) > G(f_{k_{t+1}}^i|\mu_k^*, (\sigma_k^*)^2 I)$
    end for
    $\widetilde{\mathcal{F}}^i = \{F_1^i, F_2^i, ..., F_K^i\}$
end for
return $\Theta^*$, $\widetilde{\mathcal{F}}^1, ..., \widetilde{\mathcal{F}}^N$
```

Inspired by [32] and [33], we employ a Gaussian Mixture Model (GMM) to learn the UMM by estimating the appearance and location distribution of all the 3D block features. Thus each Gaussian component can represent a certain spatial-temporal mode modeling the variations among a set of low-level features with similar appearance and relative locations in videos.

Simply we can train a GMM with spherical Gaussian components as follows:

$$P(f|\Theta) = \sum_{k=1}^{K} w_k G(f|\mu_k, \sigma_k^2 I), \quad (1)$$

where $\Theta = (w_1, \mu_1, \sigma_1, \ldots, w_K, \mu_K, \sigma_K)$; K is the number of Gaussian mixture components; I is the identity matrix; w_k, μ_k, σ_k are the mixture weight, mean, and diagonal covariance of the k-th Gaussian component $G(f|\mu_k, \sigma_k)$. We use the typical Expectation Maximization (EM) algorithm to estimate the parameters of GMM by maximizing the likelihood of the training feature set. After training the UMM, each Gaussian component builds correspondence of a group of block features from different STMs, which constitute a local ST mode universally.

2) Fitting Stage: The UMM learned above can be regarded as a container with K-components GMM. Then, given any STM, we aim to formulate it as a parameterized instance of the UMM. For this purpose, our basic idea is to assign local features on the STM into the K Gaussian "buckets" according to their probabilities.

Formally, an expression manifold M^i can be presented as a set of local block features, i.e. $\mathcal{F}^i = \{f_1^i, \ldots, f_{B_i}^i\}$, where B_i is the number of features on M^i. For the k-th Gaussian component $G(f|\mu_k, \sigma_k)$ on the UMM, we can calculate the probabilities of each f_b^i in F^i as

$$P_k^i = \{p_k(f_b^i) \mid p_k(f_b^i) = w_k G(f_b^i|\mu_k, \sigma_k^2 I)\}_{b=1}^{B_i}. \quad (2)$$

We sort the block features f_b^i in descending order of P_k^i, and the top T features with the largest probabilities are selected for the k-th local mode construction, which can be represented as $F_k^i = \{f_{k_1}^i, \ldots, f_{k_T}^i\}$. The selected features in each set are expected to be close in space-time location and share similar appearance characteristics, which can represent the local variations occurred in a certain facial region during a small period of time. Different from the hard assignment in traditional GMM, by using such a soft manner, one feature can be assigned to multiple modes (components) for sharing, which brings favorable robustness against mis-assignment. Moreover, discarding some useless features with low probabilities to any mode can also be regarded as a "filtering" operation, which can alleviate the influence of unexpected noises irrelevant to expressions. In Figure 4, we also demonstrate some examples of the learned local modes referring to the original spatial-temporal locations in videos.

Finally, an overall procedure is summarized in Algorithm 1. Based on the input unaligned STMs $\mathcal{F}^1, \ldots, \mathcal{F}^N$, each of which is represented by a set of low-level features, the algorithm provides two kinds of outputs: a group of learned optimal UMM parameters Θ^*, and the mutually aligned STMs $\widetilde{\mathcal{F}}^1, \ldots, \widetilde{\mathcal{F}}^N$, each of which is represented by K corresponding local modes instantiated by fitting to UMM.

IV. Expressionlet Learning

A local mode on mutually aligned STM is essentially a set of local features, which jointly express the appearance or dynamic characteristics of the data. To explore such information, in the following section, we propose the mid-level "expressionlet" to integrate the power of low-level features by modeling their distribution in a statistical manner.

A. Expressionlet Modeling

Considering the correlations and variations among the features in a local model, we calculate the covariance matrix of the set F_k^i as the representation of an expressionlet:

$$C_k^i = \frac{1}{T-1}\sum_{t=1}^{T}(f_{k_t}^i - \overline{f_k^i})(f_{k_t}^i - \overline{f_k^i})^T, \tag{3}$$

where $\overline{f_k^i}$ is the mean of the block features in set F_k^i. The diagonal entries of C_k^i represent the variance of each individual feature, and the non-diagonal entries are their respective correlations. As the expressionlets are globally aligned via UMM, the covariance modeling can provide a desirable locally tolerance to spatial-temporal misalignment.

In the end, the i-th manifold M^i can be represented as a set of expressionlets, i.e. $E^i = \{C_1^i, C_2^i, \ldots, C_K^i\}$. Here the expressionlets are Symmetric Positive Definite (SPD) matrices (i.e. nonsingular covariance matrices), lying on a Riemannian manifold [34]. We exploit a Log-Euclidean Distance (LED) [35] to project these points to Euclidean vector space, where standard vector learning methods are ripely studied, as advocated in [11].

Given a covariance matrix C, the mapping to vector space is equivalent to embedding the SPD manifold $\mathcal{M}$ into its tangent space $\mathcal{T}$ at identity matrix I, i.e.:

$$\Psi : \mathcal{M} \mapsto \mathcal{T}_I, C \mapsto (log(C)). \tag{4}$$

Let $C = U\Sigma U^T$ be the eigen-decomposition of SPD matrix C, its log can be computed by

$$log(C) = Ulog(\Sigma)U^T. \tag{5}$$

As we obtain a vector mapping of covariance matrix spanned by $log(C)$, general vector learning methods, e.g. PCA, can be employed to reduce the high dimension of expressionlet. Basically, in this work, we preserve 99% energies of PCA for each expressionlets, and conduct discriminant learning for further reduction.

B. Discriminant Learning With Expressionlets

As the expressionlet possesses the property of spatial-temporal locality, an effective way of enhancing its discriminative power is to consider the "margin" among corresponding expressionlets from different STM samples. Thus we can formulate our learning scheme via the graph embedding [12] framework.

As shown in Figure 5, In the overall expressionlet set $\{E^1, \ldots, E^N\}$, given the m-th expressionlet, which corresponds to the p-th mode on M^i, denoted as C_p^i; and the n-th expressionlet, which corresponds to the q-th mode on M^j, denoted as C_q^j (Note that, if all STMs are ordered, we can denote $m = (i-1)*K+p$ and similarly $n = (j-1)*K+q$. The indices m and n are used for better illustration), with the class label l_i, l_j for M_i, M_j respectively, the intrinsic graph W_w and penalty graph W_b can be defined as follows:

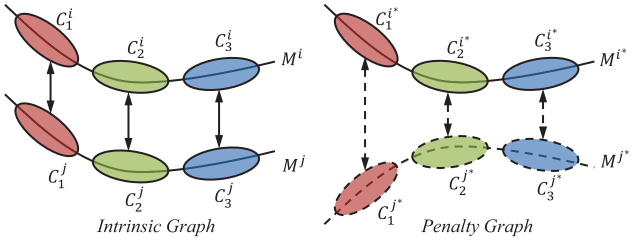


Fig. 5. The adjacency relationships of the intrinsic and penalty graphs for the discriminative learning with expressionlets (Different colors represent the different Gaussian components in UMM). M^i and M^j are two manifolds with the same class label, while M^{i^*} and M^{j^*} are with different class labels. The intrisic/penalty graph only considers the "margin" among corresponding expressionlets (C_k^i and C_k^j) generated from the same Gaussian component k.

$$W_w(m,n) = \begin{cases} 1, & if\ l_i = l_j, and\ p = q \\ 0, & otherwise \end{cases} \tag{6}$$

$$W_b(m,n) = \begin{cases} 1, & if\ l_i \neq l_j, and\ p = q \\ 0, & otherwise \end{cases} \tag{7}$$

We aim to learn an embedding function ϕ to maximize the discriminative power while simultaneously preserve the correspondence of expressionlets from the same Gaussian component. According to W_w and W_b, the within-class scatter S_w and between-class scatter S_b can be defined as:

$$S_w = \sum_{m,n} Dis(\phi(C_p^i), \phi(C_q^j))W_w(m,n), \tag{8}$$

$$S_b = \sum_{m,n} Dis(\phi(C_p^i), \phi(C_q^j))W_b(m,n), \tag{9}$$

where $Dis(\phi(C_p^i), \phi(C_q^j))$ denotes the distance between two embedded expressionlets $\phi(C_p^i)$ and $\phi(C_q^j)$.

According to Equation 5 we can obtain a vector representation x_m of the m-th expressionlet, i.e. C_p^i, where x_m is a vector spanned by $log(C_p^i)$. Simply consider a linear projection v, we can reformulate the embedded features and the distance between them in classical Euclidean space as

$$\phi(C_p^i) = v^T x_m, \phi(C_q^j) = v^T x_n, \tag{10}$$

$$Dis(\phi(C_p^i), \phi(C_q^j)) = ||v^T x_m - v^T x_n||^2. \tag{11}$$

Accordingly, we only need to learn the projection v instead of ϕ, by maximizing the between-class scatter S_b while minimizing the within-class scatter S_w:

$$v_{opt} = \arg\max \frac{v^T X(D_b - W_b)X^T v}{v^T X(D_w - W_w)X^T v}, \tag{12}$$

where D_w and D_b are diagonal matrices with diagonal elements $D_w(m,m) = \sum_n W_w(m,n)$ and $D_b(m,m) = \sum_n W_b(m,n)$. Let L_w and L_b be the Laplacian matrices of two graphs W_w and W_b. The columns of an optimal v are the generalized eigenvectors corresponding to the l largest eigenvalues in

$$XL_bX^Tv = \lambda XL_wX^Tv. \tag{13}$$

With the learned embedding function ϕ, the K expressionlets from M_i can be represented as $\{\phi(C_1^i), \ldots, \phi(C_K^i)\}$. These K features are concatenated to form a long vector as the final expression manifold (video) representation. In the end, we use multi-class linear SVM implemented by Liblinear [36] for classification.

C. Discussion

1) Expressionlet vs. AU: Action Units (AU) [37] are fundamental actions of individual or groups of facial muscles for encoding facial expression based on Facial Action Coding System (FACS). Similarly, our expressionlets are designed to model expression variations over local spatio-temporal regions in the same spirit as AUs. However, there are two differences between expressionlets and AUs: (i) AUs are manually defined concepts that are independent of person and category, while expressionlets are some mid-level representations extracted from data using learning scheme, which possess the dynamic modeling ability and discriminative power. (ii) According to FACS, each expression is encoded by the existence of a certain number of AUs. Instead of the binary coding manner, in our method, an expression can be represented by various real-valued expressionlet patterns which provide more flexible and rich information.

2) Expressionlet vs. BoVW/VLAD/FV: In our method, we extract dense local spatial-temporal features and construct a codebook (via GMM), in which each codeword can be considered as a representative of several similar local features. Both of the two operations (i.e. local feature extraction, and codebook construction) are also typical steps in Bag of Visual Words (BoVW) (or Vector of Locally Aggregated Descriptors (VLAD), and Fisher Vectors (FV)) framework.

However, in pooling stage, BoVW/VLAD/FV all perform summing/accumulating operation among the local features assigned to each certain codeword. Specifically, BoVW [38] simply estimates histogram(s) of occurrences of each codeword; VLAD accumulates the first-order difference of the vectors assigned to each codeword, which characterizes the distribution with respect to the center (codeword) [39]; Compared to VLAD, FV encodes both first-order and second-order statistics of the difference between the codewords and pooled local features and accumulates them based on the Gaussian component weights of GMM learned for codebook construction [40]. However, in our method, different from the summing operation, we make use of the second-order statistics by estimating the covariance of all the local features (augmented with location information) falling into each bucket (codeword). In this way, the local features are pooled to keep more variations, which not only encodes the relationship (difference) between the center and pooled features, but also includes the internal correlations among those pooled features which collaboratively describe some kind of motion patterns (i.e. expressionlets). In addition, in our method, by limiting the number (T in Algorithm 1) of local features falling into each bucket, not all local features are necessarily taken into account by the second-order pooling, which is also different from traditional methods. We believe such a strategy can alleviate the influence of unexpected noise or signal distortions (e.g. caused by occlusion).

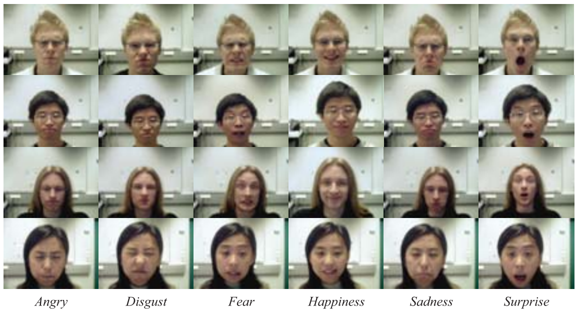


Fig. 6. The sample facial expression images extracted from the apex frames of video from Oulu-CASIA database.

V. Experiments

A. Datasets and Protocols

1) CK+ Database: The CK+ database [41] consists of 593 sequences from 123 subjects, which is an extended version of Cohn-Kanade (CK) database. The image sequence vary in duration from 10 to 60 frames and incorporate the onset (neutral face) to peak formation of the facial expression. The validated expression labels are only assigned to 327 sequences which are found to meet the criteria for 1 of 7 discrete emotions (Anger, Contempt, Disgust, Fear, Happiness, Sadness, and Surprise) based on Facial Action Coding System (FACS). We adopt leave-one-subject-out cross-validation (118 folds) following the general setup in [41].

2) Oulu-CASIA Database: The Oulu-CASIA VIS database [5] is a subset of the Oulu-CASIA NIR-VIS database, in which all the videos were taken under the visible (VIS) light condition. We evaluated our method only on the normal illumination condition (i.e. strong and good lighting). It includes 80 subjects between 23 and 58 years old, with six basic expressions (i.e. anger, disgust, fear, happiness, sadness, and surprise) of each person. Each video starts at a neutral face and ends at the apex of expression as the same settings in CK+. Similar to [5] and [42], we adopted person-independent 10-fold cross-validation scheme on the total 480 sequences. Figure 6 shows some sample facial expression images extracted from the apex frames of video from Oulu-CASIA database.

3) MMI Database: The MMI database [43] includes 30 subjects of both sexes and ages from 19 to 62. In the database, 213 sequences have been labeled with six basic expressions, in which 205 sequences were captured in frontal view. Each of the sequence reflects the whole temporal activation patterns (onset → apex → offset) of a single facial expression type. In our experiments, all of these data were used and also a person-independent 10-fold cross-validation was conducted as in several previous work [14], [42]. Compared with CK+ and Oulu-CASIA, MMI is thought to be more challenging for the subjects pose expressions non-uniformly and usually wear some accessories (e.g. glasses, moustache).

4) FERA Database: The FERA database [44] is a fraction of the GEMEP corpus [45] that has been put together to

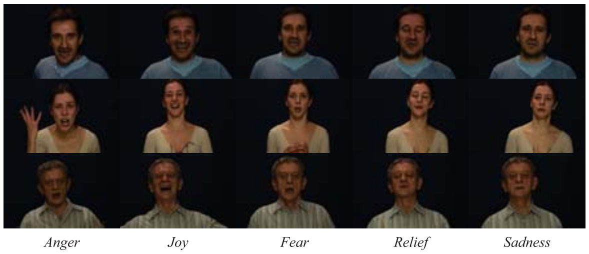


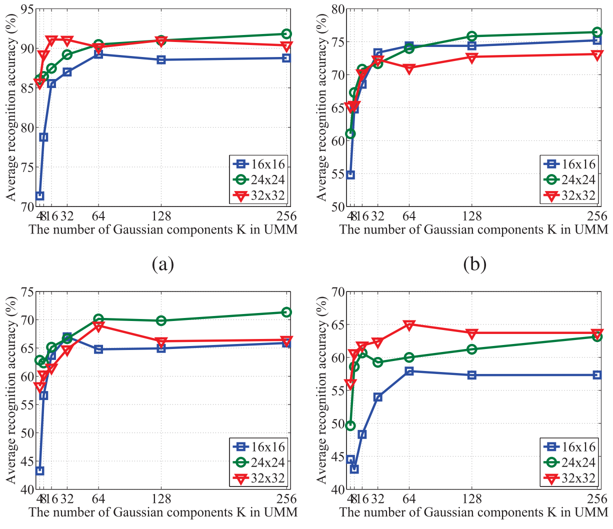
Fig. 7. The sample facial expression images extracted from the key frames of video from FERA database.

meet the criteria for a challenge on facial AUs and emotion recognition. For the emotion sub-challenge, a total of 289 portrayals were selected: 155 for training and 134 for testing. The training set included 7 (3 men) actors with 3 to 5 instances of each emotion per actor, and the test set includes 6 actors, each of whom contributed 3 to 10 instances per emotion. As the labels on the test set remain unreleased, we only use the training set and adopt leave-one-subject-out cross-validation for evaluation. The 155 sequences in the training set have been labeled with 5 expression categories: Anger (An), Fear (Fe), Joy (Jo), Sadness (Sa), and Relief (Re). FERA is more challenging than CK+, Oulu and MMI because the expressions are **spontaneous** in natural environment. Figure 7 shows some sample facial expression images extracted from the apex frames of video from FERA database.

B. Parameter Settings

For preprocessing, all the face images are normalized to 96x96 pixels based on the locations of two eyes. In the STM construction step, the low-level 3D blocks are $w * h * l$ pixels and sampled with a stride of $0.5 * w$ in spatial dimension and one frame in temporal dimension. Here w, h are tunable parameters varying over 16,24,32 (the evaluations are provided in the next subsection). Two kinds of descriptors, SIFT and HOG, are employed for low-level feature extraction. For SIFT, we apply the descriptor to the center point of each block and obtain a typical $4*4*8 = 128$ dimensions feature vector. PCA is further applied to reduce the dimension to 64. For HOG, each $w * h * l$ block is divided into $2 * 2 * 2$ grids and in each grid, the gradient orientations are quantized to 8 histogram bins, thus results in $2 * 2 * 2 * 8 = 64$ dimensions for each block. For the choice of l, we set $l = 1$ for SIFT and $l = 4$ for HOG to target both performance and efficiency.

In the following, we conduct detailed discussions on each framework component: (i) The effect of spatial scale for low-level feature extraction, which involves the parameter of patch size w, h; (ii) The effect of alignment via UMM. We compare the rigid blocking and elastic alignment manners for K local modes construction, which involves the parameter of number of modes (i.e. Gaussian components in UMM); (iii) The effect of low-level feature assignment manner in UMM fitting. Both hard-assignment and soft-assignment manners are compared and discussed regarding to the parameter of number of low-level features T to construct an expressionlet; (iv) The effect of discriminant learning with expressionlets.



(a) (b)

(c) (d)

Fig. 8. Average recognition accuracy (%) with different patch sizes for low-level feature extraction on four datasets. (a) CK+ (b) Oulu-CASIA (c) MMI (d) FERA. (**using Dense SIFT feature**).

The high-dimensions of expressionlets can be reduced simply by unsupervised PCA in vector space, or a marginal discriminant learning introduced in Section IV-B. The performance of these two schemes are compared and discussed regarding to the parameter of reduced dimension dim for an expressionlet.

C. Evaluations of Framework Components

1) The Effect of Spatial Scale for Low-Level Feature Extraction: We first evaluate the effect of spatial scale, i.e. patch size w, h, for low-level feature extraction. The w, h are varying in 16, 24, 32. Here we only take SIFT feature for example. Other parameters $T = 64$ and $dim = 256$ are fixed in the experiments on all datasets. Figure 8 illustrates the performance of different patch sizes with different numbers of Gaussian components K. As shown, on CK+, Oulu-CASIA, MMI, the green curves with $24 * 24$ perform the best. While on FERA, the results become better when adopting larger patch size. The reason may be that muscle motions induced by spontaneous expression is likely to involve larger facial regions compared to posed expression. In the following evaluations, we uniformly apply $w = h = 24$ on all datasets.

2) The Effect of Alignment via UMM: We compare the rigid blocking and elastic alignment (UMM) manners for the construction of a bank of local modes. In our experiments, the number of local modes K is varying in 16,32,64,128,256. For rigid blocking manner, the number of local modes in spatial dimension is fixed to $4 * 4 = 16$ and the blocking scheme is illustrated in Figure 9. Then the number of partitions in temporal dimension is $K/16$ (i.e. 1,2,4,8,16).

The performance comparison is shown in Figure 10. On CK+ and Oulu-CASIA, the elastic manner does not always perform better than rigid manner, especially with smaller value of K on Oulu-CASIA. It is possibly due to that the expression sequences of CK+ and Oulu-CASIA demonstrate a monotonous variation from neutral to apex status, thus the

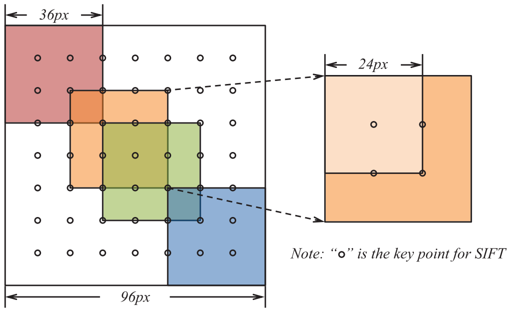
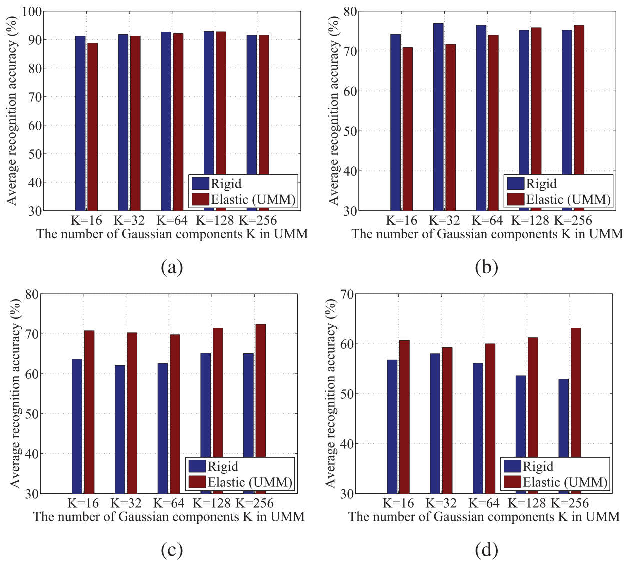


Fig. 9. An illustration of rigid blocking scheme in spatial dimension. The whole image is 96 ∗ 96 pixels and each local mode is 36 ∗ 36 pixels in spatial. For $w = h = 24$, the whole image contains $7 * 7 = 49$ key points "○" for SIFT descriptor and each local mode covers 4 as shown in the right.



(a) (b) (c) (d)

Fig. 10. Average recognition accuracy (%) with different alignment manners (rigid/elastic) on four datasets. (a) CK+ (b) Oulu-CASIA (c) MMI (d) FERA. (**using Dense SIFT feature**).

TABLE I

AVERAGE RECOGNITION ACCURACY (%) WITH DIFFERENT ASSIGNMENT MANNERS (HARD/SOFT) ON FOUR DATASETS. (a) CK+ (b) OULU-CASIA (c) MMI (d) FERA. (**USING DENSE SIFT FEATURE**)

(a)

	k=4	k=8	k=16	k=32	k=64	k=128	k=256
Hard	82.17	89.60	91.34	92.13	73.57	62.30	71.55
Soft64	86.05	86.45	87.46	89.20	90.48	90.99	91.82
Soft128	87.10	88.56	88.78	91.23	92.09	**92.75**	91.61
Soft256	87.21	87.14	88.33	86.47	88.86	87.79	87.25

(b)

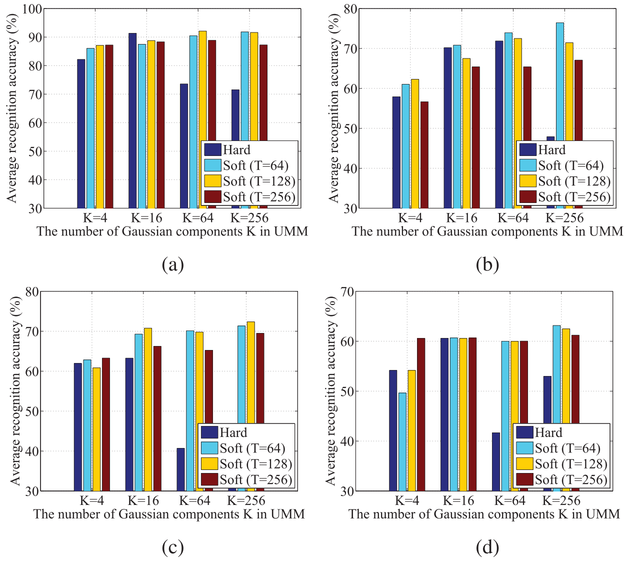
	k=4	k=8	k=16	k=32	k=64	k=128	k=256
Hard	57.92	67.08	70.21	**77.29**	71.88	47.50	47.92
Soft64	61.04	67.29	70.83	71.67	73.96	75.83	76.46
Soft128	62.29	63.75	67.50	69.79	72.50	71.25	71.46
Soft256	56.67	61.46	65.42	65.42	65.42	67.29	67.08

(c)

	k=4	k=8	k=16	k=32	k=64	k=128	k=256
Hard	61.93	63.94	63.26	64.76	40.69	37.59	43.90
Soft64	62.84	62.34	69.28	66.52	70.14	69.83	71.33
Soft128	60.84	71.07	70.77	70.27	69.78	71.42	**72.36**
Soft256	63.28	67.23	66.23	64.41	65.23	67.67	69.50

(d)

	k=4	k=8	k=16	k=32	k=64	k=128	k=256
Hard	54.16	60.59	60.59	58.09	41.66	38.04	52.95
Soft64	49.64	58.59	60.67	59.26	60.00	61.23	**63.15**
Soft128	54.16	62.69	60.57	61.21	59.98	58.67	62.48
Soft256	60.55	60.61	60.65	61.28	60.02	60.63	61.21



(a) (b) (c) (d)

Fig. 11. Average recognition accuracy (%) with different assignment manners (hard/soft) on four datasets. (a) CK+ (b) Oulu-CASIA (c) MMI (d) FERA.

temporal alignment is not the major challenge for recognition. For MMI, each of the sequence reflects the whole temporal activation from onset to apex and then to offset of a single expression in a long term; For FERA, the expression samples show much more complex temporal variations in the spontaneous manner, even with no explicit segmentation of onset, apex, or offset stages. In such situation, a temporal alignment becomes crucial for building correspondence among different sequences. As verified in our experiments, the elastic manner performs much better than the rigid manner on MMI and FERA databases. It can be observed that the improvement becomes more significant as K increases, which indicates that a larger number of local modes leads to a more elaborate alignment.

3) The Effect of Low-Level Feature Assignment Manner: In UMM fitting stage, there are also two options for low-level feature assignment to each local mode (i.e. Gaussian component). For hard assignment, each low-level feature must be assigned to only one certain component according to its largest probability (i.e. traditional GMM). For soft assignment applied in our method, each component can obtain a fixed number of features with top T probabilities. We compare these two different manners under different number of local modes (Gaussian components) $K = 4, 8, 16, 32, 64, 128, 256$ and further discuss the effect of different values of $T = 64, 128, 256$ in soft assignment. A comprehensive evaluation results are listed in Table I, with a graphical illustration in Figure 11.

As shown, the results based on hard manner can reach their peak at $K = 16$ or 32, and then suffer significant degradation as K increases. It is because that in hard manner, the larger K leads to fewer features assigned to each Gaussian component,

TABLE II

AVERAGE RECOGNITION ACCURACY (%) COMPARISON WITH EXPLET OR DIS-EXPLET ON FOUR DATASETS. (a) CK+ (b) OULU-CASIA (c) MMI (d) FERA. (**USING DENSE SIFT FEATURE**)

(a)

dim	ExpLet			Dis-ExpLet		
	k=64	k=128	k=256	k=64	k=128	k=256
64	86.19	87.16	88.57	91.01	91.10	88.03
128	89.28	89.76	89.93	92.84	**93.81**	90.56
256	90.48	90.99	91.82	92.81	93.34	93.05

(b)

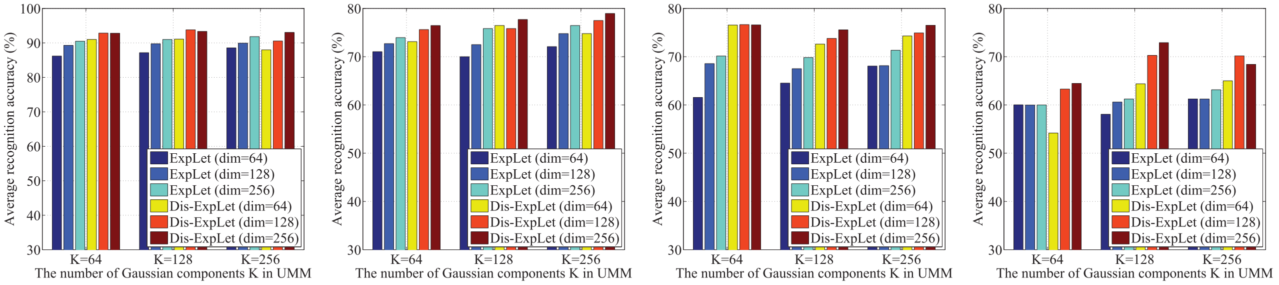
dim	ExpLet			Dis-ExpLet		
	k=64	k=128	k=256	k=64	k=128	k=256
64	71.04	70.00	72.08	73.13	76.46	74.79
128	72.71	72.50	74.79	75.63	75.83	77.50
256	73.96	75.83	76.46	76.46	77.71	**78.96**

(c)

dim	ExpLet			Dis-ExpLet		
	k=64	k=128	k=256	k=64	k=128	k=256
64	61.55	64.53	68.06	76.56	72.61	74.30
128	68.56	67.51	68.15	**76.65**	73.79	74.93
256	70.14	69.83	71.33	76.60	75.57	76.51

(d)

dim	ExpLet			Dis-ExpLet		
	k=64	k=128	k=256	k=64	k=128	k=256
64	60.03	58.07	61.25	54.18	64.38	65.00
128	59.98	60.61	61.23	63.29	70.27	70.18
256	60.00	61.23	63.15	64.48	**72.91**	68.41



(a) (b) (c) (d)

Fig. 12. Average recognition accuracy (%) comparison with ExpLet or Dis-ExpLet on four datasets. (a) CK+ (b) Oulu-CASIA (c) MMI (d) FERA.

which results in inaccurate estimation of the feature covariance for expressionlet representation. However, in soft manner, a fixed number of features assigned to each Gaussian component can guarantee a more accurate estimation of expressionlet, and hold the increasing trend as K becomes larger. On the other hand, to consider the effect of different values of T, the larger T, i.e. the more features selected in each local mode, does not always yield better performance. The reason may be that more "noise" features with low probabilities are involved when applying a larger T.

4) The Effect of Discriminant Learning: Finally we evaluate the effect of discriminant learning with expressionlets. The original dimension of expressionlets is $64 * 64 = 4096d$ as the low-level features are $64d$. For dimension reduction, we can simply apply unsupervised PCA (to $log(C)$ in Equ. 5) to obtain a low-dimensional "ExpLet", or employ the proposed discriminant learning to obtain more powerful discriminative expressionlet, which can be denoted as "Dis-ExpLet". Here we compare these two schemes by varying $dim = 64, 128, 256$ under different K, and the results are shown in Table II and Figure 12. It can be observed that "Dis-ExpLet" performs much better than "ExpLet" even using a lower dimension. The improvement is quite significant especially on MMI ($\sim$ 5.3%) and FERA ($\sim$ 9.7%), which are considered to be more challenging than CK+ and Oulu-CASIA.

D. Comprehensive Comparisons With Fisher Vector

In this section, we conduct comprehensive comparisons with the state-of-the-art encoding method Fisher Vector. The experiments are conducted based on two kinds of descriptors, i.e. SIFT (2D) and HOG (3D). And for Fisher Vector, we also tune different values of w, h to obtain the best performance. All of the results are listed in Table III.

According to the results, for $w = h = 16$ or 24, we can always observe an approximately rising trend of accuracy as the number of GMM components K increases. However, for $w = h = 32$, there usually exist an obvious degradation as K increases (except for Oulu-CASIA). It may be caused by that the patches with a larger scale encode less details which cannot provide enough local patterns for lots of partitions. Thus when K becomes larger, the cluster partitions forcibly segment some similar or related patterns, which brings confusions in pooling stage for higher-level semantics generation.

For fair comparison, in Table IV we report the performance based on original "ExpLet" (the feature dimension is reduced to dim via unsupervised PCA without discriminant learning). To simplify the discussion, we fix three inessential parameters as $w = h = 24$ and $T = 64$. As shown, the performance improves gradually with the increasing of the number of "ExpLet" K and the preserved dimension dim, and the peak values outperform the FV results significantly. Even with the same dimension of final FV representation (i.e. $2 * 64 * k = 128k$), our method (with $dim = 128$) always performs a little better, which proves that the covariance pooling scheme can capture more dynamic information for expression description thus benefits the final recognition.

Another observation is about the results based on different descriptors. For both FV and ExpLet, on CK+, Oulu-CASIA, and MMI, dense SIFT consistently performs much better than HOG, while on FERA, the HOG shows clearly superior to SIFT under all settings. The main difference of the two descriptors is whether encoding the temporal information,

TABLE III

AVERAGE RECOGNITION ACCURACY (%) BASED ON FISHER VECTOR ON FOUR DATABASES. (a) CK+ (HOG). (b) CK+ (SIFT). (c) OULU-CASIA (HOG). (d) OULU-CASIA (SIFT). (e) MMI (HOG). (f) MMI (SIFT). (g) FERA (HOG). (h) FERA (SIFT).

(a)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	58.07	65.91	68.43	78.27	77.10	80.22	83.60
24	61.17	71.10	79.05	82.42	82.92	**84.31**	82.88
32	66.36	73.16	76.68	81.52	80.83	80.56	78.38

(b)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	69.35	75.24	81.14	85.04	85.09	83.55	84.69
24	72.45	78.57	84.69	86.24	87.51	87.35	**89.18**
32	77.02	84.92	83.64	84.99	88.88	85.95	83.44

(c)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	48.54	50.83	54.79	57.08	59.38	64.17	67.92
24	51.88	57.08	62.71	61.67	67.08	67.71	**69.79**
32	46.46	51.46	56.46	61.04	64.38	68.54	67.71

(d)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	54.38	58.33	62.50	65.63	66.67	71.88	71.46
24	49.58	59.79	61.67	62.08	68.13	68.96	71.67
32	55.42	58.33	61.25	66.67	68.96	70.63	**72.92**

(e)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	42.06	47.02	48.33	55.57	55.10	54.00	58.58
24	42.76	54.81	60.39	62.56	63.41	62.70	**64.54**
32	37.99	50.94	57.37	59.24	63.99	64.03	53.77

(f)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	40.06	58.42	54.00	60.49	63.39	66.32	65.11
24	43.21	61.40	62.27	62.16	65.62	63.77	63.48
32	49.33	57.32	62.33	66.28	**68.64**	61.49	58.59

(g)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	53.00	58.88	53.06	55.02	61.45	60.83	58.19
24	57.42	55.44	62.01	63.39	63.48	**67.29**	66.58
32	59.48	62.71	60.79	59.50	67.10	60.03	54.20

(h)

w, h	k=4	k=8	k=16	k=32	k=64	k=128	k=256
16	50.47	56.25	52.38	58.88	58.31	59.54	**62.15**
24	59.69	56.06	61.33	60.17	60.83	60.17	60.67
32	53.03	55.57	52.97	56.94	60.75	60.71	48.41

TABLE IV

AVERAGE RECOGNITION ACCURACY (%) BASED ON EXPRESSIONLET ON FOUR DATABASES. (a) CK+ (HOG). (b) CK+ (SIFT). (c) OULU-CASIA (HOG). (d) OULU-CASIA (SIFT). (e) MMI (HOG). (f) MMI (SIFT). (g) FERA (HOG). (h) FERA (SIFT).

(a)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	60.05	68.25	75.35	72.18	77.81	77.92	77.14
64	65.57	70.67	73.51	80.55	79.52	78.55	80.39
128	72.66	77.03	77.04	80.26	82.27	81.68	81.19
256	76.26	77.00	81.09	81.70	85.45	82.97	82.37
512	76.38	77.17	83.19	83.46	**85.56**	83.77	82.82

(b)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	68.09	75.11	78.69	82.31	80.71	85.51	83.28
64	76.59	78.59	84.48	86.72	86.19	87.16	88.57
128	82.30	84.42	84.16	88.16	89.28	89.76	89.93
256	86.05	86.45	87.46	89.20	90.48	90.99	**91.82**
512	87.06	88.02	87.15	90.04	90.20	90.99	90.71

(c)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	33.75	41.04	50.63	57.29	60.63	61.67	63.75
64	36.88	48.96	60.21	62.08	63.96	64.79	67.08
128	45.21	54.79	66.67	65.83	66.46	70.21	69.58
256	50.63	58.54	68.13	68.75	70.21	72.08	72.50
512	54.17	62.50	69.38	71.04	72.50	73.54	**73.75**

(d)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	42.08	53.54	58.33	60.42	66.88	66.46	69.38
64	51.25	61.46	65.42	64.38	71.04	70.00	72.08
128	57.50	64.79	66.25	69.17	72.71	72.50	74.79
256	61.04	67.29	70.83	71.67	73.96	75.83	76.46
512	63.96	69.38	73.75	73.75	75.63	76.46	**76.88**

(e)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	37.45	42.75	51.88	58.70	64.04	58.02	61.25
64	42.22	53.90	56.76	62.52	66.44	62.16	64.30
128	46.47	55.58	60.41	65.05	68.64	63.64	**69.18**
256	49.38	58.92	62.35	66.53	69.02	63.91	68.92
512	49.11	58.96	63.16	67.00	67.58	66.33	68.32

(f)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	46.13	49.12	51.71	58.42	63.89	63.27	65.16
64	52.49	55.16	58.86	67.07	61.55	64.53	68.06
128	59.59	59.00	65.33	66.84	68.56	67.51	68.15
256	62.84	62.34	65.15	66.66	70.14	69.83	71.33
512	62.16	63.83	69.35	68.17	**72.00**	71.29	71.88

(g)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	51.99	43.06	52.30	61.32	59.90	63.75	61.85
64	53.37	44.41	54.84	59.34	61.88	65.06	63.79
128	57.95	49.60	58.69	61.92	63.79	67.58	68.25
256	56.53	52.18	58.05	62.50	63.77	68.20	**69.52**
512	56.53	50.91	58.01	62.46	61.83	65.62	67.58

(h)

dim	k=4	k=8	k=16	k=32	k=64	k=128	k=256
32	38.75	50.28	54.14	56.72	60.65	**64.50**	62.52
64	41.35	56.66	56.15	54.84	60.03	58.07	61.25
128	45.79	58.55	56.80	55.49	59.98	60.61	61.23
256	49.64	58.59	60.67	59.26	60.00	61.23	63.15
512	48.99	59.22	63.84	60.57	63.19	62.50	63.77

i.e. SIFT is in 2D and HOG is in 3D. We conjecture that for spontaneous samples in FERA, the variations along temporal dimension are more complex and thus require more detailed and elaborate encoding via low-level descriptors.

E. Comparisons With State-of-the-Art Methods

In this section, we compare the final results with several state-of-the-art methods. Two performance metrics, i.e. the mean recognition accuracy on each category (denoted as "mAcc") and the overall classification accuracy (denoted as "Acc") are measured for comparison. The results are listed in Table V. The comparisons on CK+, Oulu-CASIA, and MMI are under exactly the same protocols, and our "ExpLet" outperforms the existing methods significantly on both indicators (Note that, for Oulu-CASIA, "mAcc" is equal to "Acc" as the numbers of samples of all categories are the same). On FERA, by adopting cross-validation only on the training set (the same to [46]), we compare the results with 4

TABLE V

STATE-OF-THE-ART METHODS ON DIFFERENT DATABASES. ("EXPLET*" IS THE RESULTS REPORTED IN [14])

(a)

Methods	mAcc	Acc
CLM [4]	74.4	82.3
AAM [41]	83.3	88.3
ITBN [31]	86.3	88.8
MCF [48]	89.4	–
Fisher Vector	89.2	91.7
ExpLet* [14]	–	94.2
ExpLet	92.8	94.8
Dis-ExpLet	**93.8**	**95.1**

(b)

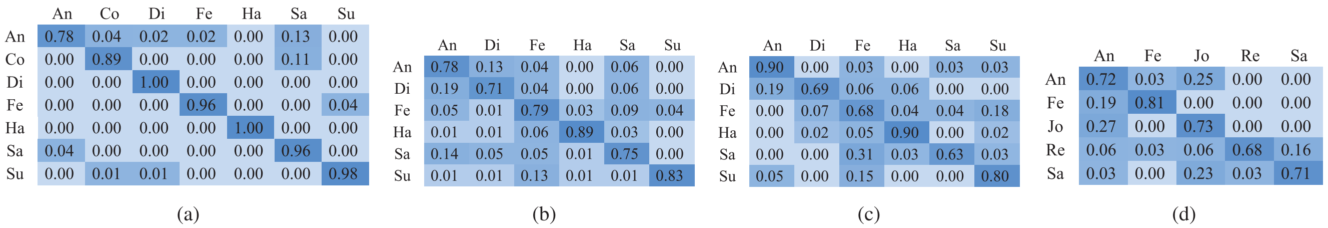
Methods	(m)Acc
AdaLBP(SVM) [5]	73.5
AdaLBP(SRC) [5]	76.2
LBP-TOP [42]	72.8
Atlases [42]	75.5
Fisher Vector	72.9
ExpLet* [14]	74.6
ExpLet	76.9
Dis-ExpLet	**79.0**

(c)

Methods	mAcc	Acc
HMM [31]	51.5	–
ITBN [31]	59.7	60.5
3DCNN [46]	50.7	53.2
3DCNNDAP [46]	62.2	63.4
Fisher Vector	68.6	70.7
ExpLet* [14]	–	75.1
ExpLet	72.4	73.7
Dis-ExpLet	**76.7**	**77.6**

(d)

Methods	mAcc	Acc
MSR [49]	56.6	–
MCF [48]	65.6	–
3DCNN [46]	46.4	46.5
3DCNNDAP [46]	56.3	56.1
Fisher Vector	67.3	67.1
ExpLet* [14]	–	–
ExpLet	69.5	69.7
Dis-ExpLet	**72.9**	**72.9**



	An	Co	Di	Fe	Ha	Sa	Su
An	0.78	0.04	0.02	0.02	0.00	0.13	0.00
Co	0.00	0.89	0.00	0.00	0.00	0.11	0.00
Di	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Fe	0.00	0.00	0.00	0.96	0.00	0.00	0.04
Ha	0.00	0.00	0.00	0.00	1.00	0.00	0.00
Sa	0.04	0.00	0.00	0.00	0.00	0.96	0.00
Su	0.00	0.01	0.01	0.00	0.00	0.00	0.98

(a)

	An	Di	Fe	Ha	Sa	Su
An	0.78	0.13	0.04	0.00	0.06	0.00
Di	0.19	0.71	0.04	0.00	0.06	0.00
Fe	0.05	0.01	0.79	0.03	0.09	0.04
Ha	0.01	0.01	0.06	0.89	0.03	0.00
Sa	0.14	0.05	0.05	0.01	0.75	0.00
Su	0.01	0.01	0.13	0.01	0.01	0.83

(b)

	An	Di	Fe	Ha	Sa	Su
An	0.90	0.00	0.03	0.00	0.03	0.03
Di	0.19	0.69	0.06	0.06	0.00	0.00
Fe	0.00	0.07	0.68	0.04	0.04	0.18
Ha	0.00	0.02	0.05	0.90	0.00	0.02
Sa	0.00	0.00	0.31	0.03	0.63	0.03
Su	0.05	0.00	0.15	0.00	0.00	0.80

(c)

	An	Fe	Jo	Re	Sa
An	0.72	0.03	0.25	0.00	0.00
Fe	0.19	0.81	0.00	0.00	0.00
Jo	0.27	0.00	0.73	0.00	0.00
Re	0.06	0.03	0.06	0.68	0.16
Sa	0.03	0.00	0.23	0.03	0.71

(d)

Fig. 13. Confusion matrices based on "Dis-ExpLet" on four datasets. (a) CK+ (b) Oulu-CASIA (c) MMI (d) FERA.

most recent methods. We also review some methods in FERA challenge [44], in person-independent setting, our result ranks in the 2nd place, only next to the "avatar" based method [47] with the accuracy of 75.2%. This may be due to that our method used fewer (6 vs. 7) subjects for training than [47]. Finally, the confusion matrices based on "Dis-ExpLet" on four datasets are illustrated in Figure 13. On all posed datasets, "happy" is always easy to be recognized, while "fear" and "sad" are more difficult and easy to be confused with each other. However, on spontaneous dataset FERA, low accuracy is obtained almost on all of the categories due to the large variations in natural and different performing manners from each subject.

VI. CONCLUSION

In this paper, we propose a new method for dynamic facial expression recognition. By considering two critical issues of the problem, i.e. temporal alignment and semantics-aware dynamic representation, a kind of variation modeling is conducted among well-aligned spatio-temporal regions to obtain a group of expresssionlets, which serve as the mid-level representations to bridge the gap between low-level features and high-level semantics. As evaluated on four state-of-the-art facial expression benchmarks, the proposed expressionlet representation has shown its superiority over traditional methods for video based facial expression recognition. As the framework is quite general and not limited to the task of expression recognition, an interesting direction in the future is to exploit its applications in other video related vision tasks, such as action recognition and object tracking.

REFERENCES

[1] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1424–1445, Dec. 2000.

[2] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 915–928, Jun. 2007.

[3] P. Yang, Q. Liu, X. Cui, and D. N. Metaxas, "Facial expression recognition using encoded dynamic features," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[4] S. W. Chew, P. Lucey, S. Lucey, J. Saragih, J. F. Cohn, and S. Sridharan, "Person-independent facial expression detection using constrained local models," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Mar. 2011, pp. 915–920.

[5] G. Zhao, X. Huang, M. Taini, S. Z. Li, and M. Pietikäinen, "Facial expression recognition from near-infrared videos," *Image Vis. Comput.*, vol. 29, no. 9, pp. 607–619, 2011.

[6] A. Sanin, C. Sanderson, M. T. Harandi, and B. C. Lovell, "Spatio-temporal covariance descriptors for action and gesture recognition," in *Proc. Int. Workshop. Appl. Comput. Vis.*, 2013, pp. 103–110.

[7] A. Klaser, M. Marszalek, and C. Schmid, "A spatio-temporal descriptor based on 3D-gradients," in *Proc. Brit. Mach. Vis. Conf.*, 2008, pp. 1–10.

[8] L. Wang, Y. Qiao, and X. Tang, "Motionlets: Mid-level 3D parts for human motion recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2674–2681.

[9] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on Riemannian manifolds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1713–1727, Oct. 2008.

[10] X. Hong, H. Chang, S. Shan, X. Chen, and W. Gao, "Sigma Set: A small second order statistical region descriptor," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1802–1809.

[11] R. Wang, H. Guo, L. S. Davis, and Q. Dai, "Covariance discriminative learning: A natural and efficient approach to image set classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2496–2503.

[12] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.

[13] R. Wang and X. Chen, "Manifold discriminant analysis," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 429–436.

[14] M. Liu, S. Shan, R. Wang, and X. Chen, " Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1749–1756.

[15] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," *Image Vis. Comput.*, vol. 27, no. 6, pp. 803–816, 2009.

[16] X. Sun, H. Xu, C. Zhao, and J. Yang, "Facial expression recognition based on histogram sequence of local Gabor binary patterns," in *Proc. IEEE Conf. Cybern. Intell. Syst.*, Sep. 2008, pp. 158–163.

[17] Y. Hu, Z. Zeng, L. Yin, X. Wei, X. Zhou, and T. S. Huang, "Multi-view facial expression recognition," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Sep. 2008, pp. 1–6.

[18] U. Tariq *et al.*, "Emotion recognition from an ensemble of features," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Mar. 2011, pp. 872–877.

[19] S. Lucey, A. B. Ashraf, and J. Cohn, "Investigating spontaneous facial action recognition through AAM representations of the face," in *Face Recognitnition*. Rijeka, Croatia: INTECH Open Access Publisher, 2007, pp. 275–286.

[20] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 1, pp. 39–58, Jan. 2009.

[21] Z. Ambadar, J. W. Schooler, and J. F. Cohn, "Deciphering the enigmatic face the importance of facial dynamics in interpreting subtle facial expressions," *Psychol. Sci.*, vol. 16, no. 5, pp. 403–410, 2005.

[22] Y. Yacoob and L. S. Davis, "Recognizing human facial expressions from long image sequences using optical flow," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 6, pp. 636–642, Jun. 1996.

[23] J. F. Cohn, A. J. Zlochower, J. J. Lien, and T. Kanade, "Feature-point tracking by optical flow discriminates subtle differences in facial expression," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Apr. 1998, pp. 396–401.

[24] S. Koelstra, M. Pantic, and I. Patras, "A dynamic texture-based approach to recognition of facial actions and their temporal models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 1940–1954, Nov. 2010.

[25] M. Hayat, M. Bennamoun, and A. El-Sallam, "Evaluation of spatiotemporal detectors and descriptors for facial expression recognition," in *Proc. Int. Conf. Human Syst. Interact.*, 2012, pp. 43–47.

[26] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[27] P. Scovanner, S. Ali, and M. Shah, "A 3-Dimensional sift descriptor and its application to action recognition," in *Proc. ACM Int. Conf. Multimedia*, 2007, pp. 357–360.

[28] L. Cohen, N. Sebe, A. Garg, L. S. Chen, and T. S. Huang, "Facial expression recognition from video sequences: Temporal and static modeling," *Comput. Vis. Image Understand.*, vol. 91, pp. 160–187, Jul. 2003.

[29] L. Shang and K.-P. Chan, "Nonparametric discriminant HMM and application to facial expression recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2090–2096.

[30] S. Jain, C. Hu, and J. K. Aggarwal, "Facial expression recognition with temporal modeling of shapes," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops.*, Nov. 2011, pp. 1642–1649.

[31] Z. Wang, S. Wang, and Q. Ji, "Capturing complex spatio-temporal relations among facial muscles for facial expression recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3422–3429.

[32] T. Hasan and J. H. Hansen, "A study on universal background model training in speaker verification," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 1890–1899, Sep. 2011.

[33] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3499–3506.

[34] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *Int. J. Comput. Vis.*, vol. 66, no. 1, pp. 41–66, 2006.

[35] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Geometric means in a novel vector space structure on symmetric positive-definite matrices," *SIAM J. Matrix Anal. Appl.*, vol. 29, no. 1, pp. 328–347, 2007.

[36] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[37] P. Ekman and W. V. Friesen, "Facial action coding system: A technique for the measurement of facial movement," San Francisco, CA, USA: Consulting Psychologists Press, 1978.

[38] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Int. Workshop Statist. Learning Comput. Vis.*, 2004, pp. 1–22.

[39] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3304–3311.

[40] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 143–156.

[41] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended Cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 94–101.

[42] Y. Guo, G. Zhao, and M. Pietikäinen, "Dynamic facial expression recognition using longitudinal facial expression atlases," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 631–644.

[43] M. F. Valstar and M. Pantic, "Induced disgust, happiness and surprise: An addition to the MMI facial expression database," in *Proc. Int. Conf. Language Resour. Eval. Workshop Emotion*, 2010, pp. 65–70.

[44] M. Valstar, M. Mehu, B. Jiang, M. Pantic, and S. Scherer, "Meta-analysis of the first facial expression recognition challenge," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 4, pp. 966–979, Aug. 2012.

[45] T. Bänziger and K. R. Scherer, "Introducing the geneva multimodal emotion portrayal (gemep) corpus," in *Blueprint for Affective Computing: A Sourcebook*. Oxford, England, U.K.: Oxford University Press, 2010, pp. 271–294.

[46] M. Liu, S. Li, S. Shan, R. Wang, and X. Chen, "Deeply learning deformable facial action parts model for dynamic expression analysis," in *Proc. Asian Conf. Comput. Vis.*, 2014, pp. 143–157.

[47] S. Yang and B. Bhanu, "Facial expression recognition using emotion avatar image," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Mar. 2011, pp. 866–871.

[48] S. W. Chew, S. Lucey, P. Lucey, S. Sridharan, and J. F. Conn, "Improved facial expression recognition via uni-hyperplane classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2554–2561.

[49] R. Ptucha, G. Tsagkatakis, and A. Savakis, "Manifold based sparse representation for robust expression recognition without neutral subtraction," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops.*, Nov. 2011, pp. 2136–2143.

Mengyi Liu (S'13) received the B.S. degree in computer science and technology from Wuhan University, Hubei, China, in 2012. She is currently pursuing the Ph.D. degree in computer science with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. She was a Research Associate with the Center for Machine Vision Research, University of Oulu, Finland, in 2014. She was a Research Associate with the Language Technologies Institute, Carnegie Mellon University, USA, from 2015 to 2016. Her research interests include computer vision, pattern recognition, machine learning, human–computer interaction, and especially focus on video-based facial expression and human behavior recognition.

Shiguang Shan (M'04–SM'15) received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2004. He joined ICT, CAS, in 2002, where he has been a Professor since 2010. He is currently the Deputy Director with the Key Laboratory of Intelligent Information Processing, CAS. He has authored over 200 papers in refereed journals and proceedings in the areas of computer vision and pattern recognition. His research interests cover computer vision, pattern recognition, and machine learning. He especially focuses on face recognition related research topics. He was a recipient of the China's State Natural Science Award in 2015, and the China's State S&T Progress Award in 2005 for his research work. He has served as the Area Chair for many international conferences, including ICCV'11, ICPR'12, ACCV'12, FG'13, ICPR'14, ICASSP'14, and ACCV'16. He is an Associate Editor of several journals, including the IEEE TRANSACTIONS ON IMAGE PROCESSING, the *Computer Vision and Image Understanding*, the *Neurocomputing*, and the *Pattern Recognition Letters*.

Ruiping Wang (S'08–M'11) received the B.S. degree in applied mathematics from Beijing Jiaotong University, Beijing, China, in 2003, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2010. He was a Post-Doctoral Researcher with the Department of Automation, Tsinghua University, Beijing, from 2010 to 2012. He was a Research Associate with the Computer Vision Laboratory, Institute for Advanced Computer Studies, University of Maryland, College Park, from 2010 to 2011. He has been a Faculty Member with the Institute of Computing Technology, Chinese Academy of Sciences, since 2012, where he is currently an Associate Professor. His research interests include computer vision, pattern recognition, and machine learning.

Xilin Chen (M'00–SM'09–F'16) received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1988, 1991, and 1994, respectively. He was a Professor with the Harbin Institute of Technology from 1999 to 2005. He has been a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, since 2004. He has authored one book and over 200 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He is a fellow of the China Computer Federation. He served as an Organizing Committee/Program Committee Member for over 50 conferences. He was a recipient of several awards, including the China's State Natural Science Award in 2015, the China's State Scientific and Technological Progress Award in 2000, 2003, 2005, and 2012 for his research work. He is currently an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, a Leading Editor of the *Journal of Computer Science and Technology*, and an Associate Editor-in-Chief of the *Chinese Journal of Computers*.