Neurocomputing 159 (2015) 126–136

Contents lists available at ScienceDirect

Neurocomputing

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

AU-inspired Deep Networks for Facial Expression Feature Learning

Mengyi Liu, Shaoxin Li, Shiguang Shan *, Xilin Chen

Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China

ARTICLE INFO

Article history:
Received 11 June 2014
Received in revised form
8 November 2014
Accepted 7 February 2015
Communicated by Qingshan Liu
Available online 27 February 2015

Keywords:
Facial expression recognition
AU-inspired Deep Networks (AUDN)
Micro-Action-Pattern
Receptive field
Group-wise sub-network learning

1、提出AU-inspired Deep Network(AUDN)
2、充分利用AU的信息但是避免检测AU
- 信息局部性变化
- 结合局部性变化的优化方法
- 最后表情识别的高级别表征

ABSTRACT

Most existing technologies for facial expression recognition utilize off-the-shelf feature extraction methods for classification. In this paper, aiming at learning better features specific for expression representation, we propose to construct a deep architecture, AU-inspired Deep Networks (AUDN), inspired by the psychological theory that expressions can be decomposed into multiple facial Action Units (AUs). To fully exploit this inspiration but avoid detecting AUs, we propose to automatically learn: (1) informative local appearance variation; (2) optimal way to combining local variation and (3) high level representation for final expression recognition. Accordingly, the proposed AUDN is composed of three sequential modules. Firstly, we build a convolutional layer and a max-pooling layer to learn the Micro-Action-Pattern (MAP) representation, which can explicitly depict local appearance variations caused by facial expressions. Secondly, feature grouping is applied to simulate larger receptive fields by combining correlated MAPs adaptively, aiming to generate more abstract mid-level semantics. Finally, a multi-layer learning process is employed in each receptive field respectively to construct group-wise sub-networks for higher-level representations. Experiments on three expression databases CK+, MMI and SFEW demonstrate that, by simply applying linear classifiers on the learned features, our method can achieve state-of-the-art results on all the databases, which validates the effectiveness of AUDN in both lab-controlled and wild environments.

AUDN：
1.MAP表征：卷积和池化层学习微动作模式(MAP)表征--明确表示由表情带来的局部外观变化
2.感受野构建：多层特征组模拟更大的感受野--自适应结合MAP，生成更加抽象的中级语义
3.分组子网学习：对每个感受野分别多层处理为为更高级的表示构造分组的子网络

1. Introduction

In the recent years, automatic facial expression recognition has attracted much attention due to its potential applications, such as human–computer interaction, multimedia, and surveillance. In the literature, many works have been done, especially to classify six basic expressions [1–3]. However it remains a great challenge to achieve accurate expression recognition in real-world applications due to the large inter-personal differences in facial expression appearance, as well as those caused by other imaging conditions. The key to cope with the challenge is to develop more robust feature representations for facial expression. According to the representation exploited, previous expression recognition methods can be roughly categorized into two groups: Action Units (AU) based methods [4,5] and appearance-based methods [2,6], which will be briefly reviewed respectively in the following. For a more comprehensive survey of facial expression recognition, readers are referred to [7].

The *AU-based methods* are designed based on strong support from physiology and psychology, as facial expression is the result of the motions of facial muscles. In the Facial Action Coding System (FACS) [8], an pioneering work in expression recognition, each expression is decomposed into several facial Action Units (AUs) with correspondence to the muscle motions. Following the FACS, many methods recognize expressions by first detecting these pre-defined AUs and then decoding specific expression from them. For instance, [4] utilized the positions of facial landmarks and geometrical modeling for AU recognition and then expression analysis, and [5] used a dynamic Bayesian network to model the relationships among various AUs to achieve the recognition of single AU or AU combinations. However, since the AUs are essentially defined based on invisible muscle motions, they are difficult to detect from the skin appearance in the face image.

On the contrary, given the input images, *appearance-based methods* recognize facial expressions via classifying with some off-the-shelf features extracted directly from the image, without explicitly taking muscle motion into account. Typical features include Local Binary Pattern (LBP) as in [2], Gabor as in [9], Local Gabor Binary Patterns (LGBP) as in [10], Scale Invariant Feature Transform (SIFT) as in [11], and Histogram of Oriented Gradient (HOG) as in [12]. As the variations caused by expression are mainly located around certain facial parts, extracting the features from the whole face might undermine the contribution of the most expressive regions. To avoid undesirable influence caused by irrelevant appearance variations, [3,13] applied feature selection to automatically search for descriptive features extracted from local patches.

* Corresponding author.
E-mail address: sgshan@ict.ac.cn (S. Shan).

http://dx.doi.org/10.1016/j.neucom.2015.02.011
0925-2312/

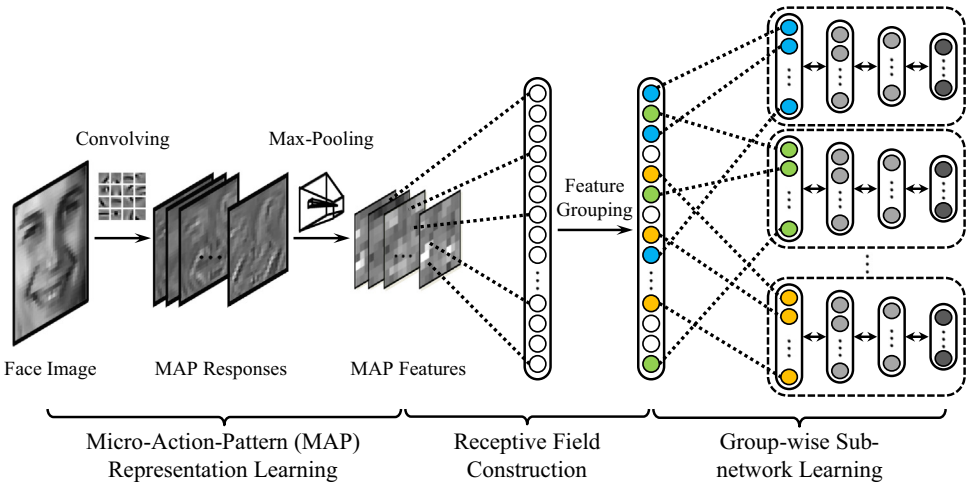


Fig. 1. The pipeline of the proposed method. There are three sequential modules, i.e. Micro-Action-Pattern (MAP) representation learning, receptive field construction, and group-wise sub-network learning.

Then the selected local features are directly concatenated into a vector for expression classification.

To briefly summarize, the appearance-based methods attempt to infer high-level expression category directly from low-level features, which is difficult and also known as the semantic gap. Thus we need something in the middle to bridge them. For facial expression, AU is a good choice to play the role. Unfortunately, AU seems not computationally feasible due to its rigid definition and hard encoding manner. To handle the above problems, in this paper, we construct a deep network to learn such mid-level representation automatically. Inspired by the principle of AUs, we propose a computational concept, Micro-Action-Pattern (MAP), which is learned from data for capturing local appearance variations caused by motion of facial muscles, such as frown, grin, and glare. Furthermore, similar to the decomposition of expression into AUs, we also designed mechanism to combine MAP into groups to simulate the abstract of higher-level concepts. As MAPs can be treated as units in a certain layer of our deep networks, each group of MAPs constitutes a receptive field of a certain unit in the next layer to specify the network connections between adjacent layers. For each receptive field, i.e. a subset of MAPs, a multi-layer learning process is further employed, which constructs a group-wise sub-network for learning incrementally higher-level features. An overview of the proposed method is presented in Fig. 1. There are three sequential modules, i.e. Micro-Action-Pattern (MAP) representation learning, receptive field construction, and group-wise sub-network learning. As the proposed deep architecture performs a hierarchical feature learning process inspired by the interpretation of facial AUs, we call it AU-inspired Deep Networks (AUDN).

This paper is an extended version of our previous conference paper [14]. There are three major differences in this extended version: (1) different strategies for receptive field construction are extensively evaluated for comprehensive understanding about optimal feature combination in deep networks; (2) For group-wise feature learning in each receptive field, two methods, i.e. Multi-Layer Perceptron (MLP) [15] and Deep Belief Networks (DBN) [16], are compared to clarify the necessity of additional unsupervised pertaining in DBN and (3) cross-database validation experiments are added to validate the generalization ability of proposed AUDN method for expression recognition.

The main contributions of this paper are summarized as follows: (1) We propose a novel AU-inspired feature learning framework to extract features specifically for expression recognition, which have strong descriptive power and are more physiologically and psychologically appealing. (2) Different receptive field construction and subnetwork learning schemes are investigated to explore what kind of compositional local features is good for mid-level expression description. (3) With only linear classifier, the learned features achieve state-of-the-art performance on all three databases under strict person-independent protocol, i.e. lab-controlled databases CK+ [17], MMI [18], and a wild scenario database SFEW [19].

The remainder of this paper is structured as follows. We present the details of our proposed AUDN in Section 2. Section 3 gives a brief introduction of the databases and settings in our experiments. Section 4 provides comprehensive evaluations on our whole framework and three sequential modules. Section 5 concludes the paper.

2. AU-inspired Deep Networks

As shown in Fig. 1, the proposed AUDN is constituted of three functional modules, i.e. MAP representation learning, receptive field construction and group-wise subnetwork learning. In this section, we present how these three well-designed modules in AUDN can help learn expression specific features automatically.

2.1. Micro-Action-Pattern Representation

One of the key ingredients of well known FACS [8] theory is that an observed expression can be decomposed into a set of local appearance variations produced by specific AUs. Inspired by this observation, in order to learn high level expression specific features, we should first encode these local variations for subsequent usage. Considering the locality of AU, we densely (with step of 1 pixel) sampled a large number of small patches from all the training expression images. And by employing a clustering algorithm, we can obtain a bank of typical patterns, the so-called Micro-Action-Pattern (MAP) prototypes in our framework, to jointly represent all kinds of local variations caused by facial expressions. Specifically, suppose the patch size is u-by-u pixels, to obtain an over-complete representation, we set $K > u^2$ in K-means clustering and learn K centroids $c^{(k)}(k = 1, 2, \ldots, K)$ from all patches after normalizing and whitening, which are considered as the MAP prototypes mentioned above. Then each MAP prototype is used as a filter to convolve with the patches in a whole facial images, for calculating "responses" to this MAP (filter). For an l-by-l-pixel input image with t-by-t patches (where $t = l + 1 - u$),

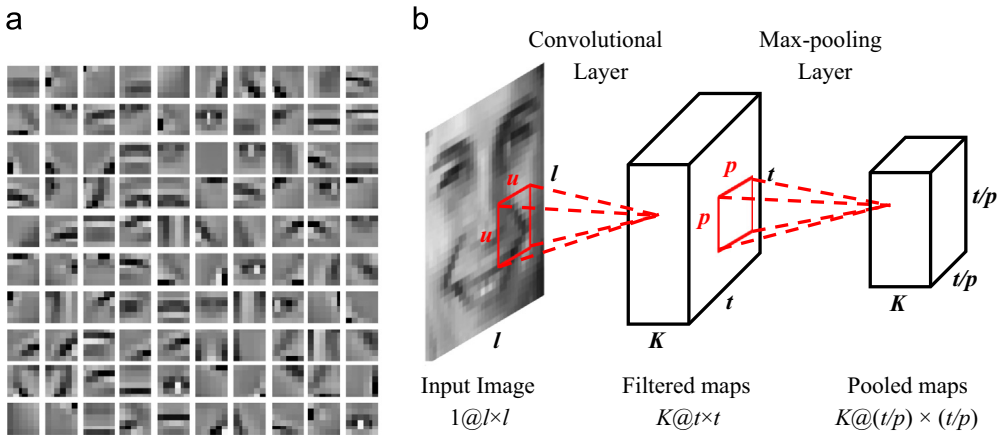


Fig. 2. Implementation details: (a) MAP prototypes (i.e. convolutional filters) learned by K-means. (b) Process of convolving and pooling.

each 2D grid of t-by-t responses for a single filter is generally called a "feature map". In the end we will get a t-by-t-by-K dimensions representation after the convolutional layer. To achieve translation invariance, we further apply max-pooling over adjacent, disjoint p-by-p patches on each t-by-t map to obtain the final MAP representation for each expression image (see Fig. 2).

2.2. Receptive field construction

In this module, we focus on constructing receptive fields from the outputs of max-pooling layer (i.e. MAP representation), each of which corresponds to a complex combinations of local appearance variations depicted by MAPs.

2.2.1. Discussion

Due to the preconceived biological notions of receptive field, in the traditional deep networks [20–23], the receptive fields are often manually defined as local spatial regions. However, in various recognition scenarios, the size and shape of receptive fields may be task-dependent which makes it difficult to predefine without prior knowledge. As expression often involves a set of synergetic movements of disconnected facial regions, such task-dependent receptive field architecture may be especially crucial for deep networks aiming at facial expression modeling. To address this problem, some previous works did have been proposed. For example, [24] proposed to select receptive fields with arbitrary shapes from a pre-generated over-complete set, which is constituted of all possible rectangle candidates of receptive field in spatial locality. Coates and Ng [25] proposed to construct receptive fields automatically by grouping sets of most similar features in the current deep network layer. In this work, we propose to exploit more adaptive receptive fields in network layers. Two major issues have been considered on grouping MAPs into receptive field: feature redundancy within each receptive field and feature relevance to the expression categories. First, if features are highly redundant, single receptive field may not be informative enough for subsequent feature learning. Second, the relevance between features and expression categories should be considered to improve the description and discrimination of each receptive field.

2.2.2. Formulation and algorithm

In this paper, a uniform formulation is proposed to discuss the effect of two factors: usage of supervised information (Supervision (S)/No Supervision (NS)), i.e. the relevance between features and expression labels, and within-receptive-field feature redundancy (Redundancy (R)/No Redundancy (NR)).

Before presenting the uniform formulation of receptive field construction, we first introduce the measurement of supervision and redundancy. In our approach, the supervision is measured by the relevance between the category label and specific MAP features, and the redundancy is measured by the relevance among MAP features within the same receptive field. Here mutual information is utilized to measure relevance of two variables. Formally, given two random variables x and y, their mutual information is defined in accordance with their probabilistic density functions $p(x)$, $p(y)$, and $p(x,y)$:

$$I(x;y)=\iint p(x,y)\log\frac{p(x,y)}{p(x)p(y)}\,dx\,dy. \tag{1}$$

Suppose a MAP subset S (that forms the receptive field) has m features $S=\{x_i|i=1,2,\ldots,m\}$. Given expression label c, the supervised information can be represented by measuring the overall label-relevance:

$$D(S,c)=\frac{1}{|S|}\sum_{x_i\in S}I(x_i;c). \tag{2}$$

If there is no supervision, the self-information entropy can be used instead:

$$H(S)=\frac{1}{|S|}\sum_{x_i\in S}I(x_i;x_i). \tag{3}$$

Intuitively, features of higher label-relevance or larger entropy are more descriptive for classification, as the former implies more discriminability while the latter means larger variance. So, we attempt to maximize $D(S,c)$ in case of supervision or $H(S)$ in unsupervised scenario. Note that, in implementation, the distribution of continuous x_i is approximated via discretizing it to discrete variables.

Similarly, the overall redundancy between every pair of MAP features within a receptive field is defined as follows:

$$R(S)=\frac{1}{|S|^2}\sum_{x_i,x_j\in S}I(x_i;x_j). \tag{4}$$

In previous deep networks, the receptive fields are often manually designed as local spatial regions, in which the features are highly redundant. We argue that this kind of receptive field may not be informative enough for subsequent feature learning. To explore whether the features within each receptive field should be more redundant or not, we evaluate two conflicting criteria: maximize $R(S)$ and minimize $R(S)$. By combining it with the above information theory terms, four criteria are designed respectively for comparison: $\max(H+R)$, $\max(H-R)$, $\max(D+R)$, and $\max(D-R)$. The four schemes and formulations are listed in Table 1.

Table 1
The correspondence of our schemes for MAP grouping and their formulations.

Schemes	NS + R	NS + NR	S + R	S + NR
Formulations	$\max(H+R)$	$\max(H-R)$	$\max(D+R)$	$\max(D-R)$

Inspired by [26], we apply a greedy search to find the local optimal features defined by our objective. Suppose we already have a set S^{k-1} containing $k-1$ features. Our goal is to find the k-th feature from the rest $P=X-S^{k-1}$. The incremental algorithm optimizes the following condition:

$$\max_{x_j \in P}[OBJ(x_j) \pm \frac{1}{k-1}\sum_{x_i \in S^{k-1}} I(x_j;x_i)]. \quad (5)$$

where $OBJ(x_j)$ represents $I(x_j;c)$ or $I(x_j;x_j)$. The algorithm is shown in Algorithm 1.

Algorithm 1. Receptive Field Construction.

Input
MAP representations: $X=\{x_i|i=1,2,\ldots,M\}$;
Expression labels: c;
The number of MAPs in each receptive field: m;
The number of receptive fields: N;
Output
Receptive fields $RF^{(n)}=\{x_i^{(n)}|i=1,2,\ldots,m\}$;
Algorithm
1: Initialize $S=\phi$, $P=X-S$, $RF^{(n)}=\phi$;
2: **for** $k=1,\ldots,m$ **do**
3: **for** $n=1,\ldots,N$ **do**
4: Search for feature x_s from P based on Eq. (5):

$$x_s=\begin{cases}\underset{x_s\in P}{\operatorname{argmax}} OBJ(x_s), & k=1\\ \underset{x_s\in P}{\operatorname{argmax}}\left[OBJ(x_s)\pm\frac{1}{k-1}\sum_{x_i^{(n)}\in RF^{(n)}} I(x_s;x_i^{(n)})\right], & k>1\end{cases} \quad (6)$$

5: Update $RF^{(n)}=RF^{(n)}\bigcup\{x_s\}$, $S=S\bigcup\{x_s\}$, $P=X-S$;
6: **end for**
7: **end for**

To show the difference of the selected features in each receptive field under different schemes, some examples of local patches corresponding to MAPs are visualized in Fig. 3. We can clearly find that the "R" schemes tend to group the MAPs in local spatial region, while "NR" schemes can group some disconnected patches together for co-occurrence consideration. "S" seems to be prone to selecting features around eyes or mouth, which are more informative for characterizing expressions.

2.3. Group-wise Sub-network Learning (GSL)

After the groups of MAPs in each Receptive Field (RF) are determined, we then learn for each RF a sub-network taking the MAPs in this RF as the input. In this module, we apply two addition layers for higher-level features learning in each sub-network. After training of each layer, the features in all previous layers can be concatenated to a long vector for the sub-classifier. Simply, "GSL 1" denotes the usage of only first layer, i.e. groups of MAPs learning by receptive field construction module; "GSL 2" denotes the usage of first layer concatenated with the first hidden layer; and "GSL 3" denotes the usage of all three layers in the sub-network.

For the multi-layer group-wise sub-network learning, we investigate two mainstream algorithm: Multi-Layer Perceptron (MLP) [15], which is trained via fully supervised gradient descent, and Deep Belief Networks (DBN) [16], which involves an unsupervised pre-training step and a supervised fine-tuning step.

2.3.1. Multi-Layer Perceptron (MLP)

A Multi-Layer Perceptron (MLP) is a feedforward artificial neural network model consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called back-propagation for training the network [15,27]. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable [28]. Formally, a one-hidden layer MLP constitutes a function f: $R^D\rightarrow R^L$,

$$f(x)=G(b^{(2)}+W^{(2)}(s(b^{(1)}+W^{(1)}x))), \quad (7)$$

where D is the size of input vector x and L is the size of the output vector $f(x)$, with bias vectors $b^{(1)},b^{(2)}$; weight matrices $W^{(1)},W^{(2)}$ and activation functions G and s. The vector $h(x)=s(b^{(1)}+W^{(1)}x)$ is the output of the hidden layer. $W^{(1)}\in R^{D\times D_h}$ is the weight matrix connecting the input vector to the hidden layer. Each column $W^{(1)}_{\cdot i}$ represents the weights from the input units to the i-th hidden unit. Typical choices for s include *tanh* or the logistic *sigmoid* function. More generally, one can build a deep MLP by stacking more such hidden layers, each of which may have a different dimension.

To train an MLP, we learn all parameters of the model using Stochastic Gradient Descent with minibatches. The gradients can be calculated using the back-propagation algorithm.

2.3.2. Deep Belief Networks (DBN)

Deep Belief Networks (DBN) are graphical models which learn to extract a deep hierarchical representation of the training data. They model the joint distribution between observed vector x and the ℓ hidden layers h^k $(k=1,2,\ldots,l)$ as follows:

$$P(x,h^1,\ldots,h^\ell)=\left(\prod_{k=0}^{\ell-2}P(h^k|h^{k+1})\right)P(h^{\ell-1},h^\ell), \quad (8)$$

where $x=h^0$, $P(h^{k-1}|h^k)$ is a conditional distribution for the visible units conditioned on the hidden units at level k, and $P(h^{\ell-1},h^\ell)$ is the visible-hidden joint distribution in the top-level.

It has been shown that Restricted Boltzmann Machines (RBM) can be stacked and trained in a greedy manner to build a DBN [16,29]. The single RBM is a two-layer (i.e. visible and hidden layer) undirected graphic model without lateral connections. The nodes in visible layer and hidden layer are represented as v_i, h_i respectively. If the visible units are real values, the configurations of v_i, h_i are characterized by an energy function as follows:

$$E(v,h)=\frac{1}{2}\sum_i v_i^2-\sum_{i,j}v_iW_{ij}h_j-\sum_j b_jh_j-\sum_i c_iv_i, \quad (9)$$

where W_{ij} characterizes the association between visible and hidden nodes and c_i, b_j are the biases of visible layer and hidden layer respectively. Probabilistically, this is interpreted as

$$P(v,h)=\frac{\exp(-E(v,h))}{Z},\quad Z=\sum_{v,h}\exp(-E(v,h)). \quad (10)$$

The hidden nodes are conditionally independent given the visible layer nodes, and vice versa. The parameters of RBM can be optimized by performing stochastic gradient descent on maximizing the log-likelihood of training data. As computing the exact gradient of log-likelihood is intractable, Contrastive Divergence (CD) approximation is used [30] which works fairly well in practice.

As RBM is usually served as an unsupervised "pre-training" tool, we also performed supervised "fine-tuning" after stacked

Fig. 3. Examples of patches corresponding to grouped features under different schemes: (a) NS + R, (b) NS + NR, (c) S + R, (d) S + NR.

RBMs to refine the parameters. This procedure is equivalent to initializing the parameters of a MLP with the weights and hidden layer biases obtained with the stacked RBMs.

3. Databases and evaluation protocols

In this section, we introduce the three databases and the corresponding evaluation protocols in our experiments. Two of them, CK+ [17] and MMI [18], are lab-controlled data. Another one, SFEW [19], is wild environment data.

3.1. CK+ database

The CK+ database [17] consists of 593 sequences from 123 subjects, which is an extended version of Cohn–Kanade (CK) [31] database (some examples are shown in Fig. 4). The validating emotion labels are only assigned to 327 sequences which were found to meet criteria for one of the 7 discrete emotion (anger, contempt, disgust, fear, happiness, sadness, and surprise) based on FACS. In our experiments, to compare with other methods [3,2] those focus on 6 basic expression classes, we make use of 309 of these sequences (remove 18 "contempt" sequences out). For each sequence, the first image (neutral face) and three peak frames are used for prototypic expression recognition which are the same as the settings in [2,3]. Based on the subject ID given in the dataset, we construct 10 person-independent subsets by sampling in ID ascending order with step size equals 10 and adopt 10-fold cross-validation as in [3]. What is more, to avoid parameter sensitivity, only linear SVM classifier [32] is used in all of our experiments.

3.2. MMI database

The MMI database [18] includes 30 subjects of both sexes and ages from 19 to 62. In the datasets, 213 sequences have been labeled with six basic expressions, in which 205 sequences are captured in frontal view. We use the data from all these 205 sequences as in [3]. Same as the settings on CK+, the neutral face and three peak frames in each sequence have been used and 10-fold cross-validation is conducted in the same way (construct 10 person-independent subsets by sampling in ID ascending order with step size equals 10). Compared to CK+, MMI have more challenging conditions: The subjects posed expressions non-uniformly, and many of them wear accessories (e.g. glasses, moustache).

3.3. SFEW database

For further validation, we also evaluate our method in a much more difficult scenario: facial expressions in the wild. The Static Facial Expression in the Wild (SFEW) database [19], which has been extracted from movies (see examples in Fig. 5), is different from the available facial expression datasets generated in highly controlled lab environments. It is the first database that depicts real-world or simulated real-world conditions for expression recognition. The database is divided into two sets. Each set contains seven subfolders corresponding to seven expression categories (anger, disgust, fear, neutral, happy, sad, and surprise). The sets were created in strict person independent manner that there is no overlap between training and testing set. In total, there are 700 images (346 in Set1, 354 in Set2) and 95 subjects. According to Strictly Person Independent (SPI) Protocol for SFEW, two-fold experiments need to be conducted, i.e. train on Set1/Set2 and test on Set2/Set1. The average accuracy of two-fold experiments is used as final performance measurement.

4. Experiments

In this section, we provide a comprehensive evaluation of the building modules and the whole framework of our method. All comparisons are performed on the data introduced in Section 3. We discuss all the important parameters and alternative network structures in our experiments. Before feeding to our AUDN, all the faces are detected automatically by Viola–Jones face detector [33] and then normalized to 32×32 based on the locations of eyes detected by method in [34].

Fig. 4. The sample facial expression images from CK+ database.

Fig. 5. The sample facial expression images from SFEW database.

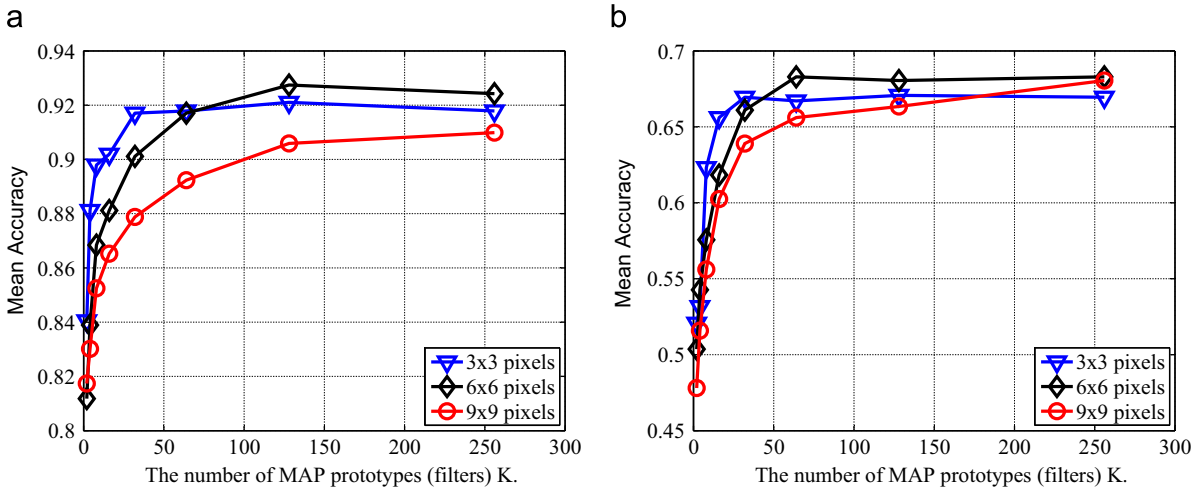


Fig. 6. Comparison of different MAP parameters. (a) CK+ database. (b) MMI database.

4.1. Evaluation on MAP representation

In MAP representation learning module, there are two important parameters, i.e. the size of densely sampled patches u, and the number of MAP prototypes (filters) K. We evaluate the effect of two parameters by ranging $u=3,6,9$ and $K=2^1,2^2,\ldots,2^8$. Here max-pooling is applied over 3-by-3 regions on each convolutional feature map for the final MAP representation. Using a linear SVM classifier, we can obtain the recognition results as shown in Fig. 6.

As shown, for all different sizes of patches, the recognition accuracy is improved as the number of MAP prototypes K increases. The "6×6 patches" worked best when there are enough MAP prototypes (e.g. $K>100$). Considering the computational time and space, we choose $u=6$ and $K=100$ in our method. Accordingly, the dimension of final MAP features is $9\times 9\times 100=8100$.

4.2. Evaluations on receptive field construction

Recalling Algorithm 1, two parameters are very important for the receptive field construction: i.e. the number of MAPs in each receptive field (i.e. the size of receptive field) m, and the number of receptive field N. We comprehensively evaluate the effect of the two parameters by

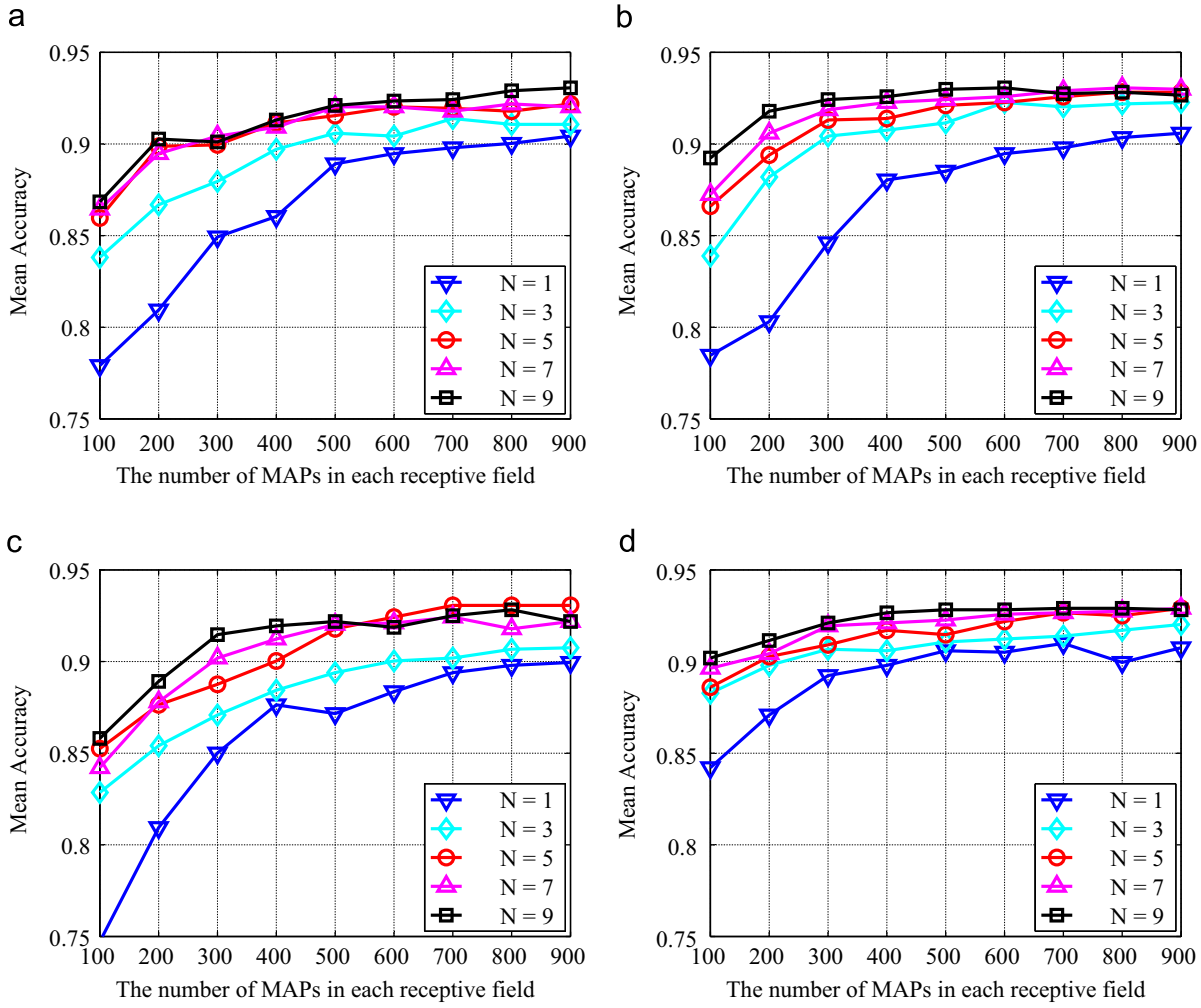


Fig. 7. Performance comparison on CK+ with different receptive field construction schemes: (a) No Supervision + Redundancy (NS + R). (b) No Supervision + No Redundancy (NS + NR). (c) Supervision + Redundancy (S + R). (d) Supervision + No Redundancy (S + NR).

ranging $m = 100, 200, \ldots, 900$ and $N = 1, 2, \ldots, 9$. To find the best way for constructing receptive fields, we also evaluate the four criteria for feature grouping as described in Section 2.2: NS + R, NS + NR, S + R, and S + NR. Under different parameters in Algorithm 1, the results are demonstrated in Fig. 7 for CK+ and Fig. 8 for MMI respectively.

As can be seen from the figures, we can get consistent observation on both of the databases: the sorting of the methods in terms of mean accuracy can be given as $NS+R < NS+NR$, $S+R < S+NR$. And the sorting of the methods in terms of sensitivity to parameters can be given as $NS+R > NS+NR > S+R > S+NR$. The observations imply that (i) redundancy controlling is good for constructing a batch of descriptive receptive fields, and further improves the fusion performance. (ii) With supervision, the algorithm can achieve comparable accuracy even using low dimensional features.

We also conduct comparisons with two general schemes, "random" (which means constructing each receptive field by randomly selecting MAP features) and "spatial" (which means constructing receptive fields according to the spatial locations of features). The same parameters are evaluated for "random" and only $N=9$ is performed for "spatial". To illustrate the overall trends of each scheme, we average the results obtained under different numbers of receptive fields. Including "random" and "spatial" schemes, we show the comparisons in Fig. 9. We can see that "spatial" scheme cannot provide good performance due to the large redundancy within each local receptive field. "random" scheme gets the middle place for its ability to reduce redundancy by scattering the features.

4.3. Evaluations on group-wise sub-network learning

In each sub-network, The number of units in visible layer equals to the feature dimensions in receptive field, i.e. $m = 100, 200, \ldots, 900$. The number of units in hidden layers is 100 and 50 respectively. Two kinds of multi-layer feature learning methods, i.e. Multi-Layer Perceptron (MLP) and Deep Belief Networks (DBN), are evaluated in this paper. After multi-layer learning, for each receptive field, the features in all the layers are concatenated to construct final hierarchical features for sub-classifier. In the end, the sub-classifiers from each receptive field are averaged for fusion result as demonstrated in Fig. 10.

As redundancy controlling is proved to be effective, in this module, we conduct our experiment only based on $NS+NR$ and $S+NR$. We demonstrate the overall trends of $NS+NR$ and $S+NR$ before and after group-wise sub-network learning in Fig. 11. As shown, DBN achieves better performance compared to MLP due to unsupervised pre-training, which initializes the model to a point in parameter space that renders the optimization process more effective, in the sense of achieving a lower minimum of the empirical cost function.

4.4. Overall results

Under the optimal combinations of the three modules, the best performance we achieve are listed in Table 2 (where the "GSL x"

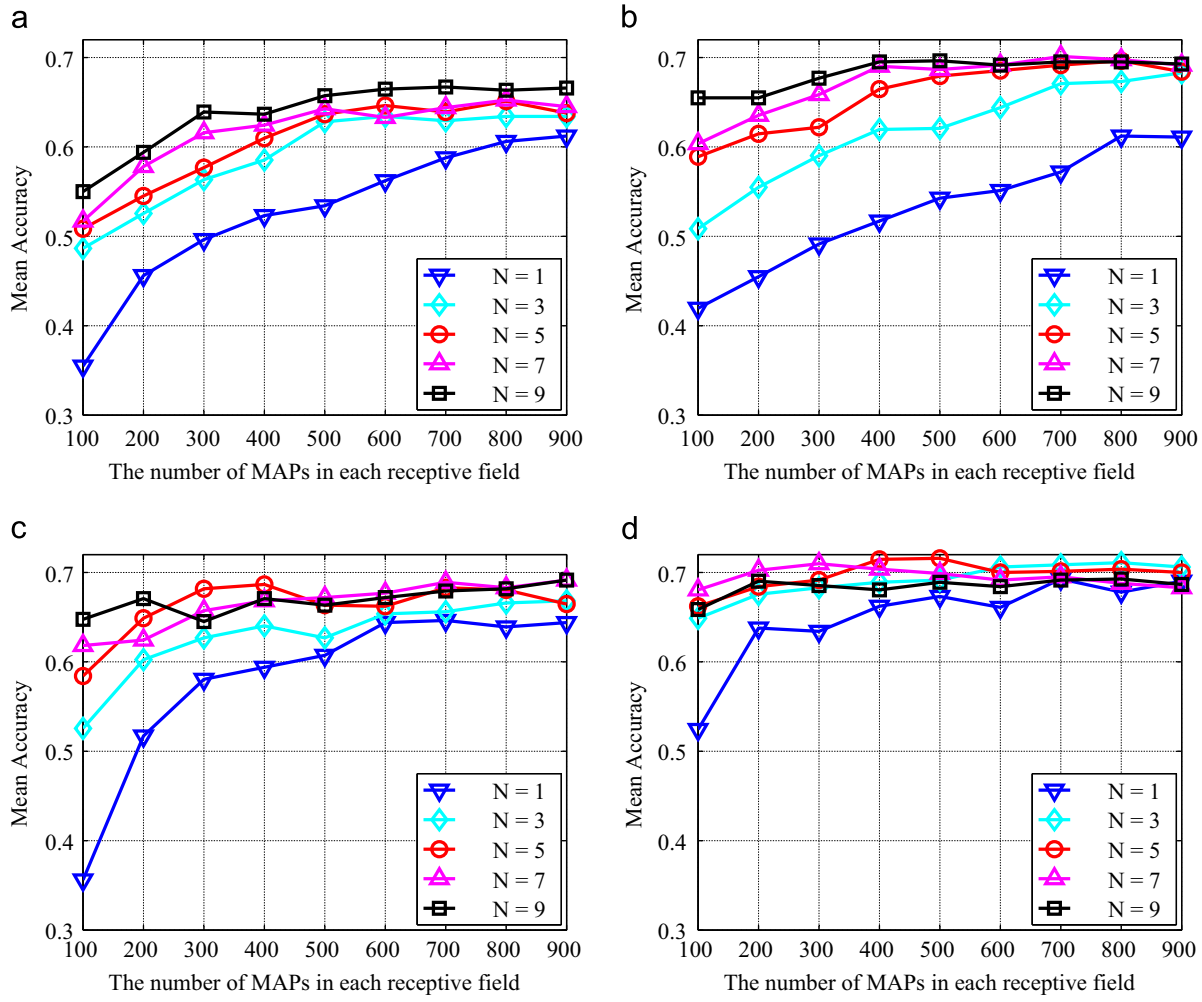


Fig. 8. Performance comparison on MMI with different receptive field construction schemes: (a) No Supervision + Redundancy (NS + R). (b) No Supervision + No Redundancy (NS + NR). (c) Supervision + Redundancy (S + R). (d) Supervision + No Redundancy (S + NR).

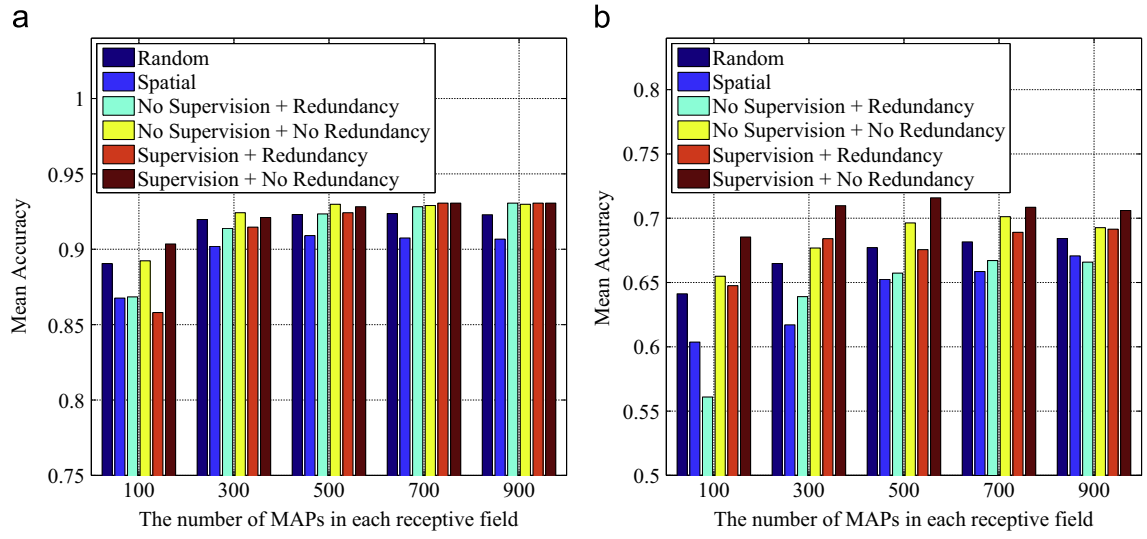


Fig. 9. Performance comparison of different strategies for receptive field construction. (a) CK+ database. (b) MMI database.

represents different evaluation manners in Group-wise Sub-network Learning, as mentioned in Section 2.3). We compared our "learned feature" with the hand-crafted features as listed in Table 2. All the experimental results are based on the features achieved by running the descriptor code released by the authors. The experiment settings are Image size is 32×32 pixels as the same as before. LBP (944 dimensions): 16 patches with the size of 8×8 pixels and 59 dimensions uniformed feature on each patch; SIFT (1152 dimensions): 9 lattice points with the step of 8 and 128 dimensions feature for each point; HOG (1568 dimensions): 7×7

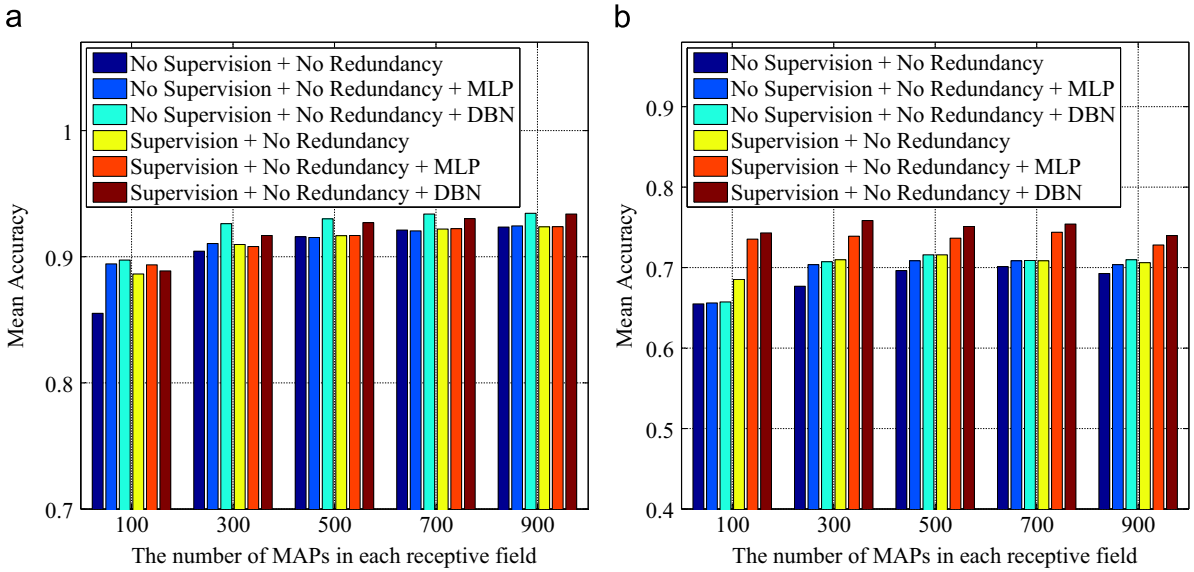


Fig. 10. Performance comparison of different group-wise sub-networks learning schemes. (a) CK+ database. (b) MMI database.

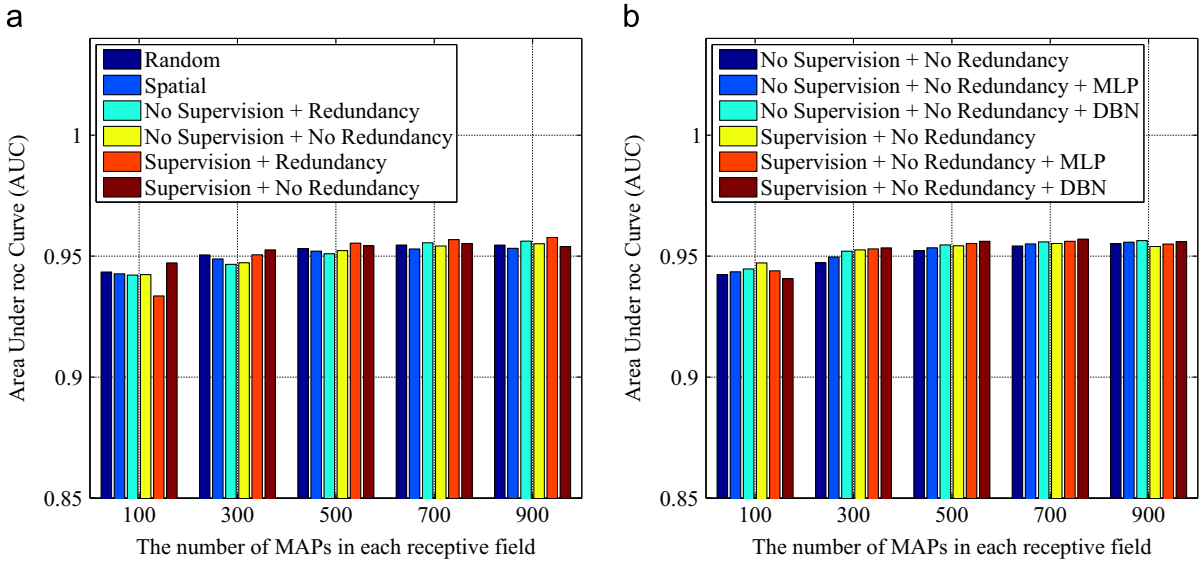


Fig. 11. AU detection performance comparison on CK+ database. (a) Different strategies for receptive field construction. (b) Different group-wise sub-networks learning schemes.

Table 2
Expression recognition performance on CK+, MMI, and SFEW databases (compared with hand-crafted features).

Methods	Accuracy (%)		
	CK+	MMI	SFEW
LBP	83.87 81.89(RBF)	52.93 50.37(RBF)	21.29 23.71 (RBF)
SIFT	86.39 87.31 (RBF)	57.80 61.46 (RBF)	20.45 21.14 (RBF)
HOG	89.53 88.61(RBF)	63.17 65.24 (RBF)	19.52 22.71 (RBF)
Gabor	88.61 85.09(RBF)	56.10 57.56 (RBF)	19.29 19.14(RBF)
AUDN			
MAP	**92.19**	**68.66**	**23.14**
GSL 1	**93.06**	**71.59**	**26.43**
GSL 2	**93.70**	**73.05**	**28.86**
GSL 3	**93.62**	**75.85**	**30.14**

The bold results are obtained using our method. The underlined results are the better ones in their table cells or columns.

Table 3
Expression recognition performance comparison with state-of-the-art methods.

Methods	Shan09 [2]	Zhong12 [3]	Dhall11 [19]	Liu12 [35]	Liu13 [14]	**AUDN**
CK+	86.94	89.89	–	–	92.22	**93.70**
MMI	47.74	73.53	–	–	74.76	**75.85**
SFEW	–	–	19.00	29.30	26.14	**30.14**

overlapped blocks with the size of 8×8 pixels, 2×2 cells and 8 histogram bins for each block; Gabor (10 240 dimensions): convolutional images with 5 spatial scales and 8 orientations with 2×2 downsampling. We performed both linear SVM and RBF SVM on all these hand-crafted features for the best performance. The parameters of RBF are tuned empirically by grid searching over $c = 2^k, k = 0, 1, \ldots, 9; g = 2^l, l = -5, -4, \ldots, 0$.

Table 4
Cross-database performance on CK+, MMI, and SFEW databases.

Train/Test	CK+/MMI	CK+/SFEW	MMI/CK+	MMI/SFEW
Accuracy (%)	72.20	29.43	93.46	25.00

Table 5
AU detection performance on CK+ database (compared with hand-crafted features).

Method	LBP	SIFT	HOG	Gabor	**AUDN-MAP**	**AUDN-GSL**
AUC (%)	92.67	93.86	94.66	92.34	**95.27**	**95.78**

The bold results are obtained using our method. The underlined results are the better ones in their table cells or columns.

Table 6
AU detection performance comparison with state-of-the-art methods.

Method	SPTS [17]	CAPP [17]	SPTS+CAPP [17]	**AUDN-MAP**	**AUDN-GSL**
AUC (%)	90.0	91.4	94.5	**95.27**	**95.78**

The bold results are obtained using our method. The underlined results are the better ones in their table cells or columns.

We also conduct comparisons with the state-of-the-art methods (see Table 3). For CK+ and MMI databases, we cite two results AFL [2] and CSPL [3] which were reported in [3]. These two results are achieved under the same 10-fold person-independent protocols as ours. However, our experiments are performed on seven expression categories including neutral, which is more challenging than the six categories problem in [3]. For SFEW database, we cite the baseline results reported in [19] and the best known result in [35].

Compared to the results on CK+ and MMI, the recognition performance degrade significantly on SFEW due to its challenging image conditions, e.g. large variations of pose and illumination. The faces normalized by automatically detected eyes location suffer from severe misalignment, so that none of the algorithms can work well as on CK+ and MMI. However, it shows that gradually better results have been achieved during our learning process, and finally significant improvements are achieved on all the three databases.

4.5. Cross-database evaluation

As a learning-based methods, there is pervasive worry about its generalization ability. To evaluate this point, we also perform cross-database experiments, that is, training feature models on one database and testing on the other two data. The results are reported in Table 4, which shows that our method can also achieve very promising results even under cross-database testing. Specifically, the models trained on lab data CK+ can get similar performance on the wild data SFEW compared with training on itself. These results convincingly prove the strength of the proposed method.

4.6. Extended discussion on AU detection

AUDN is an AU-inspired framework for expression recognition without the expensive computation of AU, however in this section, we will show that our method can simultaneously work well on AU detection. The evaluation is performed on CK+ database, which full FACS coding of peak frames is provided for all 593 sequences. Following [17], we use all neutral and peak frames for training and linear one-vs-all two-class SVM for each AU detector. The Area Underneath the Receiver-Operator Characteristic (ROC) Curve (AUC) is served as the evaluation measurement. In Fig. 11, we report the average AUC results of all AUs, which is the same as in [17]. And the performance is also compared with the hand-crafted features in Table 5 and state-of-the-art method in Table 6. From these results, we can see that the proposed method is also valid for action unit detection.

5. Conclusion

In this paper, we propose to construct a deep architecture to learn features especially for facial expression recognition, which is called "AUDN". Inspired by the interpretation of AU, we propose a computational representation MAP to capture the local appearance variations caused by facial expression, and construct adaptive receptive fields to simulate the grouping of different MAPs. Additional multi-layer receptive field specific sub-network learning process can further generate high-level features which benefit expression recognition specifically. The proposed AUDN achieves the best published performance on three facial expression databases including both lab control and wild scenarios.

In the future, we will try to explore more descriptive mid-level representation rather than simple non-overlapping receptive fields and extend this method to other facial image analysis tasks related to local facial actions, such as expression synthesis, facial animation, and facial feature tracking.

Acknowledgment

This work is partially supported by National Basic Research Program of China 973 Program under Contract no. 2015CB351802, and Natural Science Foundation of China under Contract nos. 61222211, 61272319, and 61390510.

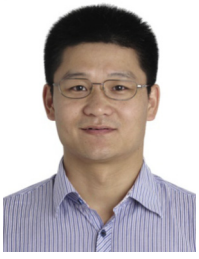
References

[1] C.E. Izard, The face of emotion, Appleton-Century-Crofts 1, New York, 1971.
[2] 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.
[3] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, D.N. Metaxas, Learning active facial patches for expression analysis, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 2562–2569.
[4] Y.-I. Tian, T. Kanade, J.F. Cohn, Recognizing action units for facial expression analysis, IEEE Trans. Pattern Anal. Mach. Intell. 23 (2) (2001) 97–115.
[5] Y. Tong, W. Liao, Q. Ji, Facial action unit recognition by exploiting their dynamic and semantic relationships, IEEE Trans. Pattern Anal. Mach. Intell. 29 (10) (2007) 1683–1699.
[6] M.J. Lyons, J. Budynek, S. Akamatsu, Automatic classification of single facial images, IEEE Trans. Pattern Anal. Mach. Intell. 21 (12) (1999) 1357–1362.
[7] Z. Zeng, M. Pantic, G.I. Roisman, T.S. Huang, A survey of affect recognition methods: audio, visual, and spontaneous expressions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (1) (2009) 39–58.
[8] P. Ekman, W.V. Friesen, Facial Action Coding System: A Technique for the Measurement of Facial Movement, Consulting Psychologists Press, Palo Alto, 1978.
[9] G. Littlewort, M.S. Bartlett, I. Fasel, J. Susskind, J. Movellan, Dynamics of facial expression extracted automatically from video, Image Vis. Comput. 24 (6) (2006) 615–625.
[10] X. Sun, H. Xu, C. Zhao, J. Yang, Facial expression recognition based on histogram sequence of local Gabor binary patterns, in: IEEE Conference on Cybernetics and Intelligent Systems, 2008, pp. 158–163.
[11] U. Tariq, K.-H. Lin, Z. Li, X. Zhou, Z. Wang, V. Le, T.S. Huang, X. Lv, T.X. Han, Emotion recognition from an ensemble of features, in: IEEE International Conference on Automatic Face and Gesture Recognition and Workshops (FG), 2011, pp. 872–877.
[12] Y. Hu, Z. Zeng, L. Yin, X. Wei, X. Zhou, T.S. Huang, Multi-view facial expression recognition, in: IEEE International Conference on Automatic Face and Gesture Recognition and Workshops (FG), 2008, pp. 1–6.
[13] P. Yang, Q. Liu, D.N. Metaxas, Exploring facial expressions with compositional features, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2638–2644.
[14] M. Liu, S. Li, S. Shan, X. Chen, Au-aware deep networks for facial expression recognition, in: IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2013.

[15] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature 323 (6088) (1986) 533–536.
[16] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[17] P. Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, I. Matthews, The extended Cohn–Kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2010, pp. 94–101.
[18] M. Valstar, M. Pantic, Induced disgust, happiness and surprise: an addition to the mmi facial expression database, in: International Conference on Language Resources and Evaluation Workshops (LRECW), 2010, pp. 65–70.
[19] A. Dhall, R. Goecke, S. Lucey, T. Gedeon, Static facial expression analysis in tough conditions: data, evaluation protocol and benchmark, in: IEEE International Conference on Computer Vision Workshops (ICCVW), 2011, pp. 2106–2112.
[20] Y. LeCun, F. J. Huang, L. Bottou, Learning methods for generic object recognition with invariance to pose and lighting, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, 2004, pp. II-97.
[21] H. Lee, R. Grosse, R. Ranganath, A.Y. Ng, Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations, in: International Conference on Machine Learning (ICML), 2009, pp. 609–616.
[22] D. Scherer, A. Müller, S. Behnke, Evaluation of pooling operations in convolutional architectures for object recognition, in: International Conference on Artificial Neural Networks (ICANN), 2010, pp. 92–101.
[23] A. Coates, A.Y. Ng, The importance of encoding versus training with sparse coding and vector quantization, in: International Conference on Machine Learning (ICML), vol. 8, 2011, p. 10.
[24] Y. Jia, C. Huang, T. Darrell, Beyond spatial pyramids: Receptive field learning for pooled image features, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 3370–3377.
[25] A. Coates, A.Y. Ng, Selecting receptive fields in deep networks, in: Advances in Neural Information Processing Systems (NIPS), vol. 24, 2011, pp. 2528–2536.
[26] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[27] F. Rosenblatt, Principles of Neurodynamics. Perceptrons and the Theory of Brain Mechanisms, Technical Report, DTIC Document, 1961.
[28] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Syst. 2 (4) (1989) 303–314.
[29] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, Greedy layer-wise training of deep networks, in: Advances in Neural Information Processing Systems, vol. 19, 2007, p. 153.
[30] G.E. Hinton, Training products of experts by minimizing contrastive divergence, Neural Comput. 14 (8) (2002) 1771–1800.
[31] T. Kanade, J.F. Cohn, Y. Tian, Comprehensive database for facial expression analysis, in: IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2000, pp. 46–53.
[32] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, Liblinear: a library for large linear classification, J. Mach. Learn. Res. 9 (2008) 1871–1874.
[33] P. Viola, M.J. Jones, Robust real-time face detection, Int. J. Comput. Vis. 57 (2) (2004) 137–154.
[34] X. Zhao, X. Chai, Z. Niu, C. Heng, S. Shan, Context modeling for facial landmark detection based on non-adjacent rectangle (nar) Haar-like feature, Image Vis. Comput. 30 (3) (2012) 136–146.
[35] M. Liu, S. Li, S. Shan, X. Chen, Enhancing expression recognition in the wild with unlabeled reference data, in: Asian Conference on Computer Vision (ACCV), 2012.

Mengyi Liu received the B.S. degree in Computer Science and Technology from Wuhan University in 2012. Currently, she is pursuing the Ph.D. degree at the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. Her research interests include computer vision, pattern recognition, human–computer interaction, and especially focus on facial expression recognition.

Shaoxin Li received the B.S. degree in computer science and technology from the Huazhong University of Science and Technology, Wuhan, China, in 2009. He is currently pursuing the Ph.D. degree with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His research interests include pattern recognition, image processing, and, in particular, face recognition and facial attribute prediction in the wild environments.



Shiguang Shan 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, Beijing, in 2004. He has been with ICT, CAS since 2002 and has been a Professor since 2010. He is also the Vice Director of the Key Lab of Intelligent Information Processing of CAS. His research interests cover image analysis, pattern recognition, and computer vision. He is focusing especially on face recognition related research topics. He received the China's State Scientific and Technological Progress Awards in 2005 for his work on face recognition technologies.

Xilin Chen received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology (HIT), Harbin, China, in 1988, 1991, and 1994 respectively. He was a Professor with the HIT from 1999 to 2005 and was a Visiting Scholar with Carnegie Mellon University from 2001 to 2004. He has been a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, since August 2004. His research interests include image processing, pattern recognition, computer vision, and multimodal interface. He has received several awards, including the China's State Scientific and Technological Progress Award in 2000, 2003, 2005, and 2012 for his research work.