Multi-channel Pose-aware Convolution Neural Networks for Multi-view Facial Expression Recognition

Yuanyuan Liu [1], Jiabei Zeng [2], Shiguang Shan* [2] and Zhuo Zheng[1]
[1]*Faculty of information engineering, China University of Geosciences, Wuhan, China*
[2] *Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing, China*
liuyy@cug.edu.cn, jiabei.zeng@vipl.ict.ac.cn, sgshan@ict.ac.cn and zhuozheng_2017@163.com

***Abstract*—Although tremendous strides have been made in facial expression recognition(FER), recognizing facial expressions in non-frontal views remains an open challenge due to the limited access to large scale training data with various poses. To make full use of the limited data, we propose a novel multi-channel pose-aware convolution neural network (MPCNN) that consists of three parts: the multi-channel feature extraction, jointly multi-scale feature fusion, and the pose-aware recognition. The feature extraction part has 3 sub-CNNs and it learns convolutional features from different features. The joint fusion part fuses multi-scale features to enhance high-level feature representation in a hierarchical way. The fused features are fed to the pose-aware recognition part that includes pose-specific recognition branches and a pose estimation sub-network. According to the estimated pose, MPCNN finally classifies the facial expression through a conditional weighted combination of the pose-specific recognition branches. MPCNN is end-to-end trainable by minimizing the joint loss of pose and expression recognition. We evaluated the proposed method on two public multi-view FER datasets (BU-3DFE and KDEF) and a FER dataset in the wild (SFEW). The experimental results demonstrate that MPCNN outperforms the state-of-the-art FER methods with both within-dataset and cross-dataset settings.**

***Keywords*-Multi-view facial expression recognition; MPCNN; Pose-aware recognition network; Jointly multi-scale fusion**

I. INTRODUCTION

Facial expressions convey cues about the emotional state of human beings and they serve as import affect signals. Hence, facial expression recognition (FER) has become a hot research topic of human-computer interaction. Automated FER is crucial to applications such as digital entertainment, customer service, driver monitoring, emotion robot, etc. [1], [2], [3]. Extensive studies and methods have been developed. A majority of the proposed methods were evaluated with constrained frontal FER, and their performance degenerates when dealing with cases of non-frontal FER [4], [5], [6]. To address this issue, we propose a jointly multi-channel pose-aware convolution neural network (MPCNN) to recognize multi-view facial expressions with great efficiency and robustness.

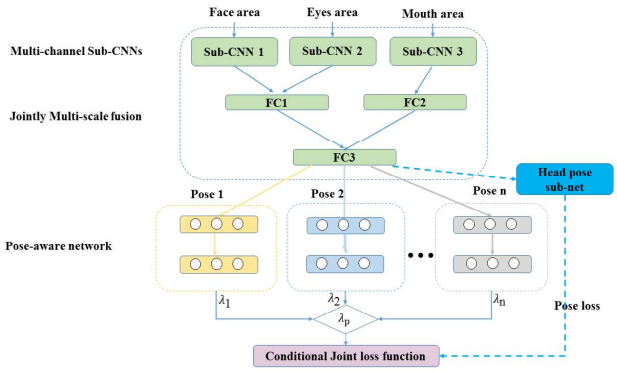

Figure 1. An overview of the proposed 3-step MPCNN for multi-view FER. Three sub-CNNs are used to extract multi-channel convolutional features, firstly. Then, the jointly multi-scale fusion layers attempt to learn a joint fusion feature in a hierarchical way, shared for multiple views. Finally, the pose-aware recognition network includes several pose-specific recognition branches and an additional head pose sub-network to overcome pose variances. λ_p is the probability of head pose that is calculated to guide the network learning through a conditional joint loss. The dash lines are processes merely in the training phase. The solid lines denote processes in both training and test phase.

A general FER framework consists of two major steps: feature extraction and classifier construction. For facial feature extraction, methods are developed based on both local and global facial feature. The accuracy of the local feature based methods relies on the detection accuracy of eyes, eyebrows, nose, and lips [7], [2], [8]; the global feature based methods usually use whole-face features to recognize expressions [9], [10], [11], [12]. Therefore, they are applicable for low-resolution images but are sensitive to occlusion and illumination variance. For multi-view FER, head pose variation and partial occlusion make robust feature extraction a challenging task. For classifier construction, the convolution neural network (CNN) has recently gained great popularity for the classification task because of its superior performance and robustness [13]. Encouraged by these advancements, many deep CNNs have been developed and trained for FER [1], [10], [7], [14]. In [14], Tang replaced the Softmax layer in the AlexNet [13] with a

978-1-5386-2335-0/18/$31.00 ©2018 IEEE

linear support vector machine (SVM), achieved a recognition accuracy of 71.2% on the FER2013 dataset. Jung *et al.* [7] trained a joint fine-tuning method on a deep temporal-geometry network and a deep appearance network for FER, which achieved accuracies of 95.22% and 81.46% in the CK+ and Oulu-CASIA datasets.

Deep CNNs automatically learn high-level feature representation from images but demand a large training dataset and high-performance computing [15], [16]. Training a deep network with limited data might even give rise to inferior performance, owing to the over-fitting problem [17]. To solver the problem, Zhou *et al.* [18] transfered feature maps to action units by a pre-trained deep CNN on the ImageNet dataset recently. The accuracy on multi-view KFED facial expression dataset reached 86.43%. Zheng *et al.* [19] proposed the deep neural network (DNN) with the SIFT feature, whose accuracy was 78.9% on the multi-view BU-3DFE dataset. In [4], a combination of CNN and special image pre-processing steps were proposed to recognize six facial expressions and an averaged accuracy of 90.96% under head pose at 0° on BU-3DFE dataset. Despite the methods were proposed for multi-view FER under various poses, it remains a large gap between the performance of the current algorithms with large pose variation using limited amount of training data.

To make full use of the limited data, we propose the MPCNN method for multi-view FER, which consists of three parts: the multiple-channel feature extraction, jointly multi-scale feature fusion and the pose-aware recognition network (see Figure 1). Our method aims at improving both accuracy and efficiency. The multi-channel convolutional features are extracted from different facial regions, and fused into a jointly multi-scale high-layer feature representation to suppress the influence of over-fitting problem. Head poses are estimated to overcome large head pose variances, and multi-view facial expressions are classified by pose-specific recognition branches in the pose-aware recognition part.

Our contributions include the following:

1) a novel MPCNN method is proposed for multi-view FER using limited amount of training data. MPCNN is end-to-end trainable with three components: multi-channel feature extractors, multi-scale high-layer feature fusion, and pose-aware recognizer.
2) the approach was evaluated on three typical multi-view facial expression datasets and showed it advantages over the state-of-the-art methods.

The rest of this paper is organized as follows: Section II presents our MPCNN method for multi-view FER. Section III discusses the experimental results using publicly available datasets. Section IV concludes this paper.

II. MPCNN FOR MULTI-VIEW FACIAL EXPRESSION RECOGNITION

A. The training architecture of the MPCNN

Figure 2 gives the training pipeline. As can be seen, first, three sub-CNNs are used to extract multi-channel convolutional features from three different facial regions (i.e., the facial region R_{face}, the mouth region R_{mouth} and the eye regions R_{eyes}). Then, in the step of jointly multi-scale feature fusion, MPCNN integrates the convolutional features into a high-level representation. The discriminative representation is shared by the sub-networks in the following pose-aware recognition network. Finally, to suppress the errors from head pose variation, the pose-aware recognition network consists of multiple pose-specific recognition branches and an additional pose estimation sub-network. MPCNN is optimized by minimizing a joint loss of pose conditioned expression recognition loss and pose estimation loss.

1) Multi-channel convolutional feature extraction: Rather than extract features from the whole face as the conventional CNNs do, MPCNN uses three sub-CNNs to extract multi-channel convolutional features from different facial regions and scales. Much recent work makes use of scale-normalized extractors for the task, e.g., VGG-face in [20] and CNN in [4]. The input images usually are resized to a canonical template size. When facing to multi-scale input images, what should the size of the template be? On one hand, we want a small scale that can extract local features from facial local regions; on the other hand, we want a large scale that can exploit detailed global features from the whole face to increase accuracy. Instead of a "one-size-fits-all" approach, we train separate multi-channel convolutional feature extractors tuned for different scales. We define $(\mathrm{M}_1;\mathrm{M}_2;\mathrm{M}_3)$ to represent three channel sub-CNNs, respectively. The input data of each channel sub-CNN comes from different regions and scales, which makes the feature maps more robust and efficient with the limited training data. During training, we use the uniform back-propagation to optimize the multi-channel sub-networks.

The first channel M_1 for the whole facial region: M_1 is used to extract and learn the global convolutional feature from the whole facial region. It first averagely normalizes the input data R_{face} (facial region) to the size of 200×200. M_1 contains three convolution layers followed by 3 max-pooling layers and 3 max-pooling layers. Each filter in a convolution layer is of size 5×5 and there are 32, 64, and 64 such filters in the first three layers, respectively. Then the final convolutional layer before output layer is of size $64 \times 50 \times 50$. For the two output layers, one outputs a probability distribution (per feature map) of head pose. The other outputs a probability distribution of facial expression.

The second channel M_2 for the eyes region: To extract facial local features from eyes regions, we use a small

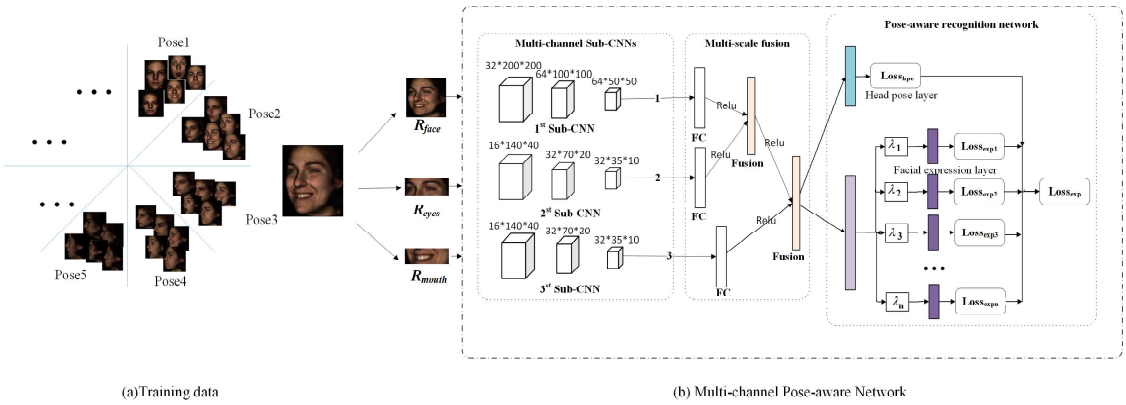

Figure 2. The MPCNN architecture. (a) The training data with different poses and expressions, (b) The training pipeline. During training, the network takes different facial regions of size 200×200, 140×40 and 140×40 as inputs, and outputs the joint losses of the pose and expression. The multi-channel sub-CNNs extract multi-scale features that are fused into the joint fusion feature representation. The pose-aware recognition network uses several pose-specific branches to learn expression losses under different views. The whole network is optimized through back-propagation in an end-to-end way.

template in the channel M_2. M_2 first crops the above about one-third of facial regions as the eyes' region, then averagely pools the eyes regions to size 140×40. As shown in Figure 2, the filter in each convolution layer of M_2 is of size 5×5 and there are 16, 32, and 32 such filters in the first three layers, respectively. The final convolutional layer before output layer is of size $32 \times 35 \times 10$. M_2 also outputs pose classification loss and expression classification loss.

The third channel M_3 for the mouth region: M_3 extracts the local mouth convolutional features. It first crops the below about one-third of facial region as the mouth region, and resizes the mouth region as 140×40. The filter in the M_3 is of size 5×5 and there are 16, 32, and 32 such filters in the first three layers, respectively. The final convolutional layer before output layer is of size $32 \times 35 \times 10$. M_3 also outputs pose classification loss and expression classification loss.

Table I
THE PSEUDOCODE OF THE JOINTLY MULTI-SCALE FUSION.

Algorithm: Jointly multi-scale fusion

Input: Convolutional feature matrixs $\mathbf{M_k}$, with $k = \{1, 2, 3\}$,
initialized weight matrixs $\mathbf{W_i^k}$,
initialized bias vectors $\mathbf{b_i^k}$.
Output: Fusion feature vector $\mathbf{f^2}$

1. Flatten feature matrixs in each sub-CNN:
$\mathbf{y_i^k} = \mathrm{Reshape}(\mathbf{M_k})$,
2. Compute and activate the connected layer parameters:
$\mathbf{v_i^k} = \mathrm{ReLu}(\mathbf{y_i^k W_i^k} + \mathbf{b_i^k})$,
3. Fuse high-level features and activation in the first fusion layer:
$\mathbf{p^1} = \mathrm{Contat}(\mathbf{v_i^1}, \mathbf{v_i^2})$,
$\mathbf{f^1} = \mathrm{ReLu}(\mathbf{p^1 W^1} + \mathbf{b^1})$,
4. Fuse and enhance high-level features in the second fusion layer:
$\mathbf{p^2} = \mathrm{Contat}(\mathbf{f^1}, \mathbf{v_i^3})$,
$\mathbf{f^2} = \mathrm{ReLu}(\mathbf{p^2 W^2} + \mathbf{b^2})$,
5. Return $\mathbf{f^2}$.

2) Jointly multi-scale feature fusion: To enhance the representation ability of the features trained from limited amount of training data, jointly multi-scale feature fusion uses two feature fusion layers to fuse multi-scale and multi-channel convolutional features in a hierarchical way. It can reduce the number of neuron nodes with general fully-connected layers through joint fusion. The jointly multi-scale fusion pseudocode is given in Table. I. We first compute and active the feature vector $\mathbf{v_i^k}$ with multi-channel convolutional features, as Step 1, 2 in Table I. $\mathbf{v_i^k}$ denotes the feature vector from the k^{th} sub-CNN channel, where i is the dimension of the vector. Then, in the first fusion layer, $\mathbf{p^1}$ and $\mathbf{f^1}$ are computed for high-layer feature representation from $\mathbf{v_i^1}$ and $\mathbf{v_i^2}$ by the Contact operation and activation, respectively. Finally, the second fusion layer enhances the high-layer fusion representation from the third feature vector $\mathbf{v_i^3}$ and the first fusion feature vector $\mathbf{f^1}$. $\mathbf{f^2}$ is the enhanced high-layer fusion feature for output.

Beside, to suppress the problem of over-fitting, we use the "Dropout" method to drop randomly some neurons after each fusion layer. In our work, the dropout parameters are set as 0.7 and 0.8 in the two fusion layers. The enhanced high-level fusion feature is fed into the pose-aware recognition network.

3) Pose-aware recognition network: The pose-aware recognition network consists of multiple pose-specific recognition branches and a pose estimation sub-network. The pose-aware recognition network attempts to deal with expression recognition across views by joint loss of pose conditioned expression recognition loss and pose estimation loss.

Give the pose p, we compute the pose conditioned expression recognition loss $L(e|p)$ as the discrepancy between the estimated facial expression and the ground truth facial expression of all the training samples from pose p. Mathe-

matically speaking, $L(e|p)$ is formulated as:

$$L(e|p) = \frac{1}{N_{\mathrm{B}}} \sum_{j=1}^{N_{\mathrm{B}}} (\hat{\mathbf{y}}_p - \mathbf{y}_p)^2, \tag{1}$$

where N_{B} denotes the number of mini-batch feeds in training. $\mathbf{y}_p \in \{0,1\}^E$ denotes facial expression labels, and E is the number of expression categories. $\hat{\mathbf{y}} \in \mathbb{R}^E$ denotes the facial expression prediction of the network under the pose p.

The pose estimation loss can be calculated by the following function in the pose estimation sub-network:

$$L_p = \frac{1}{N_{\mathrm{B}}} \sum_{j=1}^{N_{\mathrm{B}}} (\hat{\mathbf{p}} - \mathbf{p})^2, \tag{2}$$

$\hat{\mathbf{p}} \in \{0,1\}^P$ denotes head pose labels, and P is the number of pose categories. $\hat{\mathbf{p}} \in \mathbb{R}^P$ is the head pose prediction computed by the pose estimation sub-network.

MPCNN is optimized by minimizing the joint loss of several pose conditioned expression recognition loss and a pose estimation loss. The pose estimation loss L_p is used to guide the pose-specific branch learning through a conditional joint loss. The conditional joint loss function of the pose-aware recognition network for an image is defined as:

$$Loss_{\mathrm{exp}} = \sum_{p=1}^{P} \lambda_{\mathrm{p}} L(e|p) + \lambda_{\mathrm{hpe}} L_p, \tag{3}$$

where $\lambda_{\mathrm{p}} \in [0,1]$ denotes the probability of the head pose p, which is used to balance contributions of poses, calculated by the pose estimation sub-network. Different from a empirical value in multi-task CNN [21], different views contain different pose scores in pose-specific branches, which can suppress the influence of pose variances. λ_{hpe} is set as 0.5 by empirical value.

For the full MPCNN training, the structures of all channels are similar, and the gradient back propagation processes of all channels are also similar. Each channel has a head pose classification loss and a expression recognition loss. Adding them with conditional loss weights, we can calculate the joint loss function by Equation 3.

B. Multi-view facial expression recognition

The test pipeline of the MPCNN method is shown in Figure 3, which includes three stages: multi-channel convolutional feature extraction, joint fusion feature representation and pose-aware expression recognition. Given an input test image, it is first built into image pyramids to handle different facial regions and scales. In the first stage, three parallel sub-CNNs obtain the multi-channel feature maps from different facial regions, as shown in Figure 4. In the second stage, the multi-channel feature maps are fed into the joint fusion layers that output enhanced 1536-dimensional high-layer fusion feature representation in a hierarchical way. The fused

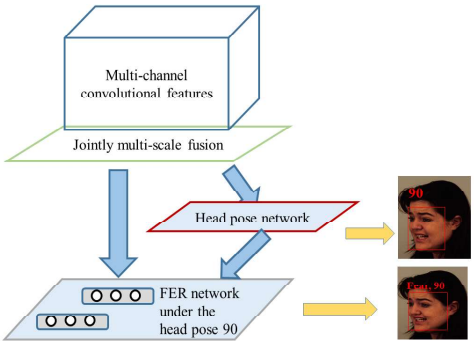

Figure 3. The test pipeline of the MPCNN for multi-view facial expression recognition.

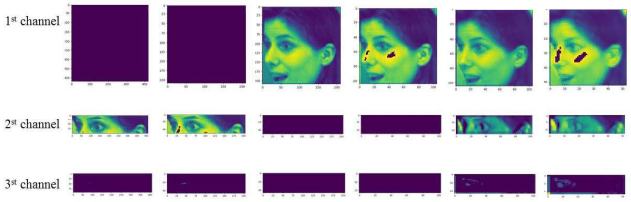

Figure 4. Multi-channel convolutional feature maps by the MPCNN.

features are used to estimate the head poses p in the yaw rotation and, meanwhile, are passed to the pose-aware recognition part. Finally, according to the estimation of each pose, the recognition part classifies the facial expression through a conditional weighted combination of the pose-specific recognition branches, which can suppress the influence of view variation.

III. Experimental Results

A. Datasets and Settings

To evaluate our approach, three face expression datasets were used: KDEF multi-view emotion dataset [22], BU-3DFE multi-view facial expression dataset [23], and SFEW in-the-wild emotion dataset [24]. The KDEF contains 35 females and 35 males displaying 7 different emotional expressions (Anger, Disgust, Fear, Happiness, Sadness, Surprise, and Neutral). Each expression is viewed from 5 different yaw angles. The BU-3DFE contains 100 people of different ethnicities, including 56 females and 44 males. Six facial expressions (Anger, Disgust, Fear, Happiness, Sadness, and Surprise) are elicited by various manners and head poses, and each of them includes 4 levels of intensities. These models are described by both 3D geometrical shapes and color textures with 83 Feature Points identified on each model. The SFEW dataset is a subset of EmotiW2015 in-the-wild emotion dataset, which contains static images based spontaneous facial expression collected in real-world

Table II
AVERAGE ACCURACIES OF HEAD POSE ESTIMATION ON BU-3DFE.

Head Pose	-90°	-60°	-45°	-30°	0°	30°	45°	60°	90°
Accuracy	1.0	1.0	0.996	0.926	1.0	0.998	0.973	0.94	1.0

conditions. SFEW has been divided into three sets: Train (880 samples), Val (383 samples) and Test (372 samples).

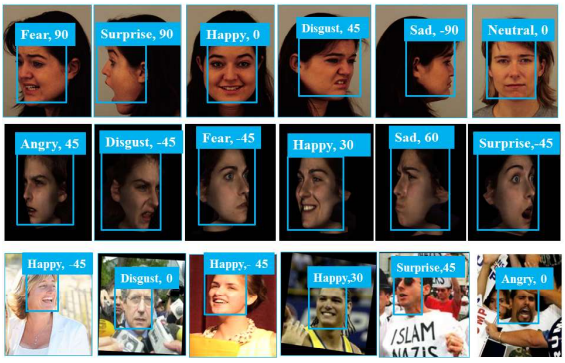

Figure 5. Examples of muti-view FER using KDEF, BU-3DFE and SFEW datasets. Top row: results of KDEF dataset. Middle row: results of the BU-3DFE dataset. Bottom row: results of SFEW dataset.

Examples of multi-view FER of KDEF, BU-3DFE, and SFEW datasets are shown in Figure 5. In our experiments, the training and validation data sets include 16515 images with 75 persons from BU-3DFE dataset, 3920 images from with 56 persons from KDEF dataset. A 5-fold cross-validation was conducted for parameter adjustment. The MPCNN model for each head pose was trained with 784 images from KDEF dataset, 1835 images from the BU-3DFE dataset. In the FER evaluation, we used other 980 images from KDEF dataset, 5166 images from the BU-3DFE dataset, and 755 images from SFEW dataset. We used the Tensorflow framework [25] for implementing CNN. The important training parameters in the experiments include learning rate (0.01), epochs (25000), mini-batch size(32), and convolution kernel size (5×5). The experiments were conducted on a PC with Intel (R) Core(TM) i7-6700 CPU at 4.00GHz and 32GB memory, and NVIDA GeForce GTX 1080.

B. Experiments with Multi-view BU-3DFE Dataset

When dealing with multi-view images, the head pose is estimated for correction of pose-induced inconsistency in expression recognition.

1) Head pose estimation: For head pose estimation, we use the same settings as the FER. Each image in the BU-3DFE dataset is automatically annotated with one out of the nine head pose labels ({ -90°, -60°, -45°, -30°, 0°, +30°, +60°, +75°, 90° }). Table II shows the average accuracies of head pose estimation in each class. The MPCNN estimated 9 head pose classes in the yaw rotation and achieved the average accuracy of 98.15%. Our method aligned head poses for expression recognition.

2) Multi-view facial expression recognition: Table III lists the confusion matrix from the BU-3DFE dataset. The average accuracy of expression recognition is 91.22% under overall head poses. The highest accuracy is 99.1% of Happy followed by that of Sadness, Surprise, and Anger, which are above 90%. The lowest accuracy is 76.9% for Fear.

Table III
CONFUSION MATRIX OF EXPRESSION RECOGNITION ON BU-3DFE.

	Anger	Disgust	Fear	Happy	Sadness	Surprise
Anger	**0.936**	0.009	0.002	0.001	0.052	0.0
Disgust	0.081	**0.815**	0.012	0.053	0.029	0.01
Fear	0.015	0.014	**0.769**	0.142	0.036	0.024
Happy	0.002	0.0	0.005	**0.991**	0.001	0.001
Sadness	0.011	0.001	0.0	0.0	**0.988**	0.0
Surprise	0.0	0.009	0.013	0.008	0.012	**0.958**

Table IV
COMPARISON OF ACCURACY (%) AND STD. USING DIFFERENT METHODS ON BU-3DFE.

Methods	Features	Poses	Accuracy(STD.)
CNN	face image	9	68.9 (1.5)
SVM [9]	LBP and LGBP	5	71.1 (1.2)
PC-RF [5]	Heterogeneity	5	76.1 (1.0)
JFDNN [7]	Image and landmarks	5	72.5 (1.3)
CGPR [8]	facial landmarks	5	76.5 (0.8)
GSRRR [19]	Sparse SIFT	9	78.9 (1.0)
DNN-D [1]	SIFT	9	80.1 (0.8)
MPCNN	**Multi-channel features**	**9**	**91.22 (0.5)**

The average accuracy of our MPCNN method is compared with that of CNN, SVM [9], Pair Conditional Random Forests (PC-RF) [5], Joint fine-tuning in deep neural networks (JFDNN) [7], Coupled gaussian process regression (CGPR) [8], Group sparse reduced-rank regression (GSRRR) [19], and Deep neural network-driven SIFT (DNN-D) [1] in Table IV. The CNN in our experiment contains three convolution layers followed by three max-pooling layers and two fully connected layers. Each filter is of size 5×5 and there are 32, 64, and 128 such filters in the first three layers, respectively. The input images are rescaled to 224 by 224.

The accuracy of the CNN on BU-3DFE dataset is 68.9% as presented in Table IV. The accuracy of multi-class SVM with LBP and LGBP in [9] is 71.1%. Dapogny et. al. [5] proposed PC-RF to capture low-level expression transition patterns on the condition of head pose estimation for multi-view dynamic facial expression recognition. On the pose various BU-3DEF dataset, the average accuracy reached 76.1%. JFDNN achieves 72.5% which contains three convolution layers and two hidden layers, where the filters in the three convolution layers are in size 5×5, the numbers

of the hidden nodes are set to be 100 and 600. The higher accuracies are achieved with SIFT feature using GSRRR and DNN-D methods proposed in [19] and [1], which are 78.9% and 80.1%, respectively. Our method achieves 91.22% which is competitive to the methods above. The lowest STD. of 0.5% using our method also proved the robustness of the proposed MPCNN.

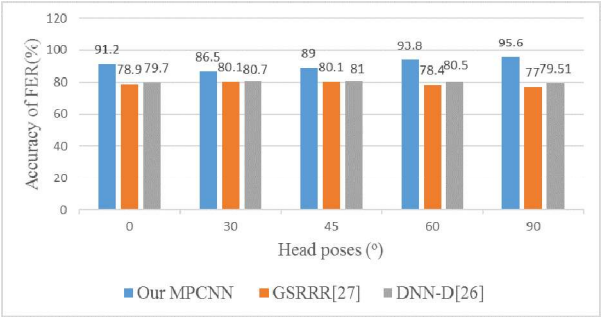

Figure 6. Average accuracy (%) comparison of multi-view FER under each head pose on BU-3DFE .

Figure 6 describes the average accuracy comparison of multi-view FER under each head pose using our method, GSRRR [19] and DNN-D [1]. The accuracies of our MPCNN under five right head poses are all higher than the accuracies of GSRRR [19] and DNN-D [1]. The highest accuracies achieved by the methods [1] and [19] are 81.0% and 80.1% at 45°, respectively. The highest accuracy of our method is achieved under the head pose 90°, which is 95.6%. And the lowest accuracy of 86.5% appear both under the head pose 30°. One can see that our method can effectively suppress the occlusion and facial deformation from large pose various.

C. Experiments with multi-view KDEF dataset

Table V
AVERAGE ACCURACIES OF HEAD POSE ESTIMATION ON KDEF.

Head Pose Class	-90°	-45°	0°	45°	90°
Accuracy	1.0	0.995	1.0	1.0	1.0

1) Head pose estimation: For head pose estimation, each image in the KDEF dataset is annotated with one out of the five head pose labels (-90°, -45°, 0°, +45°, 90°). Table V shows the accuracies of head pose estimation on KDEF dataset. The MPCNN estimated 5 head pose classes in the horizontal direction and achieved the average accuracy of 99.8% in the limited number dataset.

Table VI
CONFUSION MATRIX OF FACIAL EXPRESSION RECOGNITION ON KDEF.

	Anger	Disgust	Fear	Happy	Sadness	Surprise	Neutral
Anger	**0.881**	0.009	0.001	0.001	0.064	0.0	0.037
Disgust	0.091	**0.822**	0.012	0.053	0.012	0.01	0.0
Fear	0.043	0.058	**0.766**	0.007	0.072	0.007	0.047
Happy	0.0	0.0	0.0	**0.977**	0.023	0.0	0.0
Sadness	0.009	0.0	0.09	0.028	**0.755**	0.0	0.118
Surprise	0.0	0.05	0.013	0.0	0.05	**0.848**	0.039
Neutral	0.0	0.0	0.0	0.0	0.056	0.007	**0.938**

2) Multi-view facial expression recognition: Table VI shows the confusion matrix of multi-view facial expression recognition from the KDEF dataset. The average accuracy of 7 facial expressions is 86.9% under overall head poses. Relatively low accuracies appear between these expressions because the KDEF dataset has less training images than the BU-3DFE Dataset.

Table VII
COMPARISON OF ACCURACY (%) AND STD USING DIFFERENT METHODS ON KDEF.

Methods	Features	Poses	Accuracy
CNN	Image	5	55.8 (0.9)
SVM [9]	LBP and LGBP	5	70.5 (1.2)
SURF boosting [26]	SURF	5	74.05 (0.9)
TLCNN [18]	Action Unit Selective Feature	5	86.43 (1.0)
MPCNN	**Multi-channel features**	**5**	**86.9 (0.6)**

Table VII shows the recognition accuracies on KDEF dataset using the MPCNN, Transfer learning based CNN (TLCNN) [18], SURF boosting [26], SVM [9] and CNN. One can see that our MPCNN can obtain the best performance and robustness. Compared to TLCNN [18] using a pre-training model, our method did not depend on any pre-trained CNN models, eg., VGG-16 or AlexNet, and achieved the accuracy of 86.9% with 0.6 of STD.

D. Evaluate on cross-dataset performance

To verify the generalization of the proposed MPCNN method, cross-dataset experiments were carried out, as shown in Table VIII. When doing cross-dataset evaluation, only BU-3DFE and KDEF datasets can be used as training data. We didn't train on SFEW dataset because it has no specific annotations for the views. The proposed method achieved 70.25% and 53.16% accuracies on the KDEF and SFEW datasets when trained on BU-3DFE dataset, and 54.6% and 44.35% on BU-3DFE and SFEW datasets when

trained on KDEF dataset; it all outperformed the CNN and the pre-CNN methods. The cross-data performance on more challenging SFEW has shown to provide a substantial, about 17.21% improvement over baseline results.

Table VIII
AVERAGE ACCURACY AND STD. ACROSS DATASETS.

Training dataset	BU-3DFE dataset.		KDEF dataset	
Methods	Testing on KDEF	Testing on SFEW	Testing on BU-3DFE	Testing on SFEW
CNN	44.6(1.6)	38.85(2.5)	39.9(1.8)	32.5(2.0)
pre-CNN [18]	48.13(1.9)	42.7 (2.2)	46.55(1.8)	40.36(2.2)
MPCNN	**70.25(1.6)**	**53.16(2.0)**	**54.6(1.5)**	**44.35(2.5)**

E. Performance of different training models and training data

Tabel IX compares the performance of FER systems with different models and training data. The efficiency is shown that MPCNN has the fewest reference time, training samples and numbers of layers. From the table, MPCNN's numbers of layers, training time and model size (9, 24ms/image, 113MB), are all less than VGG-16 model (13, 58ms/image, 518MB) and ResNet-101 model (101, 79ms/image, 170MB). To discuss the influence of training data, each image of the BU-3DFE dataset is amplified 12 times using different data augmentation methods, which includes horizontal flip, rotation, skewing and zooming with cropping. The MPCNN achieved 91.22% of the accuracy without date augmentation and 94.22% with date augmentation, which outperforms other models using the same training date. One can see that the MPCNN performs well even when there are only a small amount of training data with a shallow network.

IV. CONCLUSIONS AND FUTURE WORKS

This paper proposes a novel jointly multi-channel pose-aware convolutional neural network (MPCNN) method for multi-view FER on limited availability of training data. In MPCNN, multi-channel sub-CNNs learn face, mouth and eyes' regions for multi-scale convolutional feature extraction, firstly. Then, jointly multi-scale feature fusion layers are devised to unify high-level fusion feature representation for different views and scales in a hierarchical way. Finally, the pose-aware network classifies final facial expressions under the estimated head pose by a conditional joint loss function, which can suppress the influence of pose variances.
.

Experiments were conducted using public multi-view KDEF, BU-3DFE and SFEW datasets. The proposed method achieved much improved performance and great robustness with an average accuracy of 91.22% on BU-3DFE dataset and 86.9% on KDEF dataset. The average time for performing a FER is about 11ms on two GTX 1080 GPUs. In contrast to deep CNNs which require large-scale training data, MPCNN performs well even when there is only limited availability of training data. In future work, we will introduce RNN into MPCNN construction by incorporation temporal information on video sequences.

V. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No.61602429, No.61702481, No.41701446).

REFERENCES

[1] T. Zhang, W. Zheng, Z. Cui, Y. Zong, J. Yan, and K. Yan, "A deep neural network-driven feature learning method for multi-view facial expression recognition," *IEEE Transactions on Multimedia*, vol. 18, no. 12, pp. 2528–2536, 2016.

[2] A. Tawari and M. M. Trivedi, "Face expression recognition by cross modal data association," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1543–1552, 2013.

[3] J. Wu, Z. Lin, W. Zheng, and H. Zha, "Locality-constrained linear coding based bi-layer model for multi-view facial expression recognition," *Neurocomputing*, vol. 239, pp. 143–152, 2017.

[4] A. T. Lopes, E. de Aguiar, A. F. De Souza, and T. Oliveira-Santos, "Facial expression recognition with convolutional neural networks: Coping with few data and the training sample order," *Pattern Recognition*, vol. 61, pp. 610–628, 2017.

[5] A. Dapogny, K. Bailly, and S. Dubuisson, "Dynamic pose-robust facial expression recognition by multi-view pairwise conditional random forests," *arXiv preprint arXiv:1607.06250*, 2016.

[6] M. Jampour, V. Lepetit, T. Mauthner, and H. Bischof, "Pose-specific non-linear mappings in feature space towards multi-view facial expression recognition," *Image and Vision Computing*, vol. 58, pp. 38–46, 2017.

[7] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, "Joint fine-tuning in deep neural networks for facial expression recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2983–2991.

[8] O. Rudovic, I. Patras, and M. Pantic, "Coupled gaussian process regression for pose-invariant facial expression recognition," *Computer Vision–ECCV 2010*, pp. 350–363, 2010.

Table IX
THE PERFORMANCE OF FER SYSTEMS WITH DIFFERENT MODELS AND TRAINING DATA.

Convolutional Models	# Training data	#Convs layers	Model Size	Training times /img(ms)	Test Times /img (ms)	Accuracy(%)
VGG-16 [20]	12×16515	13	518MB	58ms	21ms	92.6
VGG-16 [20]	16515	13	518MB	58ms	21ms	89.6
ResNet101 [27]	12×16515	101	170MB	79ms	33ms	94.05
ResNet101 [27]	16515	101	170MB	79ms	33ms	90.5
Multi-channel convolutional model	12×16515	3×3	113MB	24ms	11ms	94.22
Multi-channel convolutional model	16515	3×3	113MB	24ms	11ms	91.22

[9] S. Moore and R. Bowden, "Local binary patterns for multi-view facial expression recognition," *Computer Vision and Image Understanding*, vol. 115, no. 4, pp. 541–558, 2011.

[10] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*. IEEE, 2016, pp. 1–10.

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

[12] G. Fanelli, A. Yao, P.-L. Noel, J. Gall, and L. Van Gool, "Hough forest-based facial expression recognition from video sequences," in *European Conference on Computer Vision*. Springer, 2010, pp. 195–206.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[14] Y. Tang, "Deep learning using linear support vector machines," *arXiv preprint arXiv:1306.0239*, 2013.

[15] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[16] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition." in *Icml*, vol. 32, 2014, pp. 647–655.

[17] W. Ge and Y. Yu, "Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning," *arXiv preprint arXiv:1702.08690*, 2017.

[18] Y. Zhou and B. E. Shi, "Action unit selective feature maps in deep networks for facial expression recognition," in *Neural Networks (IJCNN), 2017 International Joint Conference on*. IEEE, 2017, pp. 2031–2038.

[19] W. Zheng, "Multi-view facial expression recognition based on group sparse reduced-rank regression," *IEEE Transactions on Affective Computing*, vol. 5, no. 1, pp. 71–85, 2014.

[20] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition." in *BMVC*, vol. 1, no. 3, 2015, p. 6.

[21] R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *arXiv preprint arXiv:1603.01249*, 2016.

[22] D. Lundqvist, A. Flykt, and A. Öhman, "The karolinska directed emotional faces (kdef)," *CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet*, no. 1998, 1998.

[23] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato, "A 3d facial expression database for facial behavior research," in *Automatic face and gesture recognition, 2006. FGR 2006. 7th international conference on*. IEEE, 2006, pp. 211–216.

[24] A. Dhall, O. Ramana Murthy, R. Goecke, J. Joshi, and T. Gedeon, "Video and image based emotion recognition challenges in the wild: Emotiw 2015," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. ACM, 2015, pp. 423–426.

[25] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[26] Q. Rao, X. Qu, Q. Mao, and Y. Zhan, "Multi-pose facial expression recognition based on surf boosting," in *Affective Computing and Intelligent Interaction (ACII), 2015 International Conference on*. IEEE, 2015, pp. 630–635.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," pp. 770–778, 2015.