Peak-Piloted Deep Network for Facial Expression Recognition

Xiangyun Zhao[1(✉)], Xiaodan Liang[2], Luoqi Liu[3,4], Teng Li[5], Yugang Han[3],
Nuno Vasconcelos[1], and Shuicheng Yan[3,4]

[1] University of California, San Diego, USA
{xiz019,nvasconcelos}@ucsd.edu
[2] Carnegie Mellon University, Pittsburgh, USA
xdliang328@gmail.com
[3] 360 AI Institute, Beijing, China
{liuluoqi,hanyugang}@360.cn
[4] National University of Singapore, Singapore, Singapore
eleyans@nus.edu.sg
[5] Institute of Automation, Chinese Academy of Sciences, Beijing, China
tenglwy@gmail.com

Abstract. Objective functions for training of deep networks for face-related recognition tasks, such as facial expression recognition (FER), usually consider each sample independently. In this work, we present a novel peak-piloted deep network (PPDN) that uses a sample with peak expression (easy sample) to supervise the intermediate feature responses for a sample of non-peak expression (hard sample) of the same type and from the same subject. The expression evolving process from non-peak expression to peak expression can thus be implicitly embedded in the network to achieve the invariance to expression intensities. A special-purpose back-propagation procedure, peak gradient suppression (PGS), is proposed for network training. It drives the intermediate-layer feature responses of non-peak expression samples towards those of the corresponding peak expression samples, while avoiding the inverse. This avoids degrading the recognition capability for samples of peak expression due to interference from their non-peak expression counterparts. Extensive comparisons on two popular FER datasets, Oulu-CASIA and CK+, demonstrate the superiority of the PPDN over state-of-the-art FER methods, as well as the advantages of both the network structure and the optimization strategy. Moreover, it is shown that PPDN is a general architecture, extensible to other tasks by proper definition of peak and non-peak samples. This is validated by experiments that show state-of-the-art performance on pose-invariant face recognition, using the Multi-PIE dataset.

Keywords: Facial expression recognition · Peak-piloted · Deep network · Peak gradient suppression

1 Introduction

Facial Expression Recognition (FER) aims to predict the basic facial expressions (e.g. happy, sad, surprise, angry, fear, disgust) from a human face image, as
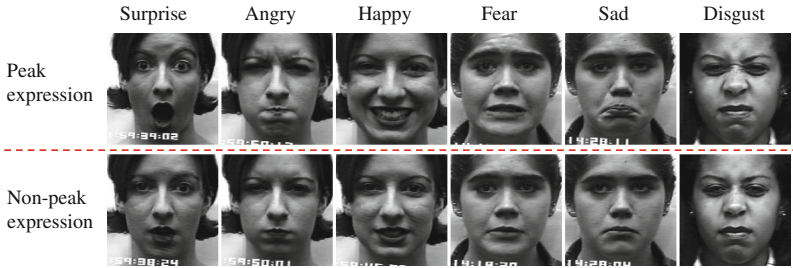
Fig. 1. Examples of six facial expression samples, including surprise, angry, happy, fear, sad and disgust. For each subject, the peak and non-peak expressions are shown.

illustrated in Fig. 1.[1] Recently, FER has attracted much research attention [1–7]. It can facilitate other face-related tasks, such as face recognition [8] and alignment [9]. Despite significant recent progress [4,10–12], FER is still a challenging problem, due to the following difficulties. First, as illustrated in Fig. 1, different subjects often display the same expression with diverse intensities and visual appearances. In a videostream, an expression will first appear in a subtle form and then grow into a strong display of the underlying feelings. We refer to the former as a non-peak and to the latter as a peak expression. Second, peak and non-peak expressions by the same subject can have significant variation in terms of attributes such as mouth corner radian, facial wrinkles, etc. Third, non-peak expressions are more commonly displayed than peak expressions. It is usually difficult to capture critical and subtle expression details from non-peak expression images, which can be hard to distinguish across expressions. For example, the non-peak expressions for fear and sadness are quite similar in Fig. 1.

Recently, deep neural network architectures have shown excellent performance in face-related recognition tasks [13–15]. The has led to the introduction of FER network architectures [4,16]. There are, nevertheless, some important limitations. First, most methods consider each sample independently during learning, ignoring the intrinsic correlations between each pair of samples (e.g., easy and hard samples). This limits the discriminative capabilities of the learned models. Second, they focus on recognizing the clearly separable peak expressions and ignore the most common non-peak expression samples, whose discrimination can be extremely challenging.

In this paper, we propose a novel peak-piloted deep network (PPDN) architecture, which implicitly embeds the natural evolution of expressions from non-peak to peak expression in the learning process, so as to zoom in on the subtle differences between weak expressions and achieve invariance to expression intensity. Intuitively, as illustrated in Fig. 2, peak and non-peak expressions from the same subject often exhibit very strong visual correlations (e.g., similar face parts) and can mutually help the recognition of each other. The proposed PPDN uses the feature responses to samples of peak expression (easy samples) to supervise

---

[1] This work was performed when Xiaoyun Zhao was an intern at 360 AI Institute.
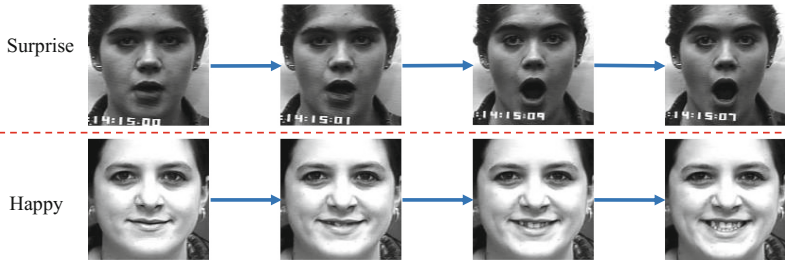
Fig. 2. Expression evolving process from non-peak expression to peak expression.

the responses to samples of non-peak expression (hard samples) of the same type and from the same subject. The resulting mapping of non-peak expressions into their corresponding peak expressions magnifies their critical and subtle details, facilitating their recognition.

In principle, an explicit mapping from non-peak to peak expression could significantly improve recognition. However, such a mapping is challenging to generate, since the detailed changes of face features (e.g., mouth corner radian and wrinkles) can be quite difficult to predict. We avoid this problem by focusing on the high-level feature representation of the facial expressions, which is both more abstract and directly related to facial expression recognition. In particular, the proposed PPDN optimizes the tasks of (1) feature transformation from non-peak to peak expression and (2) recognition of facial expressions in a unified manner. It is, in fact, a general approach, applicable to many other recognition tasks (e.g. face recognition) by proper definition of peak and non-peak samples (e.g. frontal and profile faces). By implicitly learning the evolution from hard poses (e.g., profile faces) to easy poses (e.g., near-frontal faces), it can improve the recognition accuracy of prior solutions to these problems, making them more robust to pose variation.

During training, the PPDN takes an image pair with a peak and a non-peak expression of the same type and from the same subject. This image pair is passed through several intermediate layers to generate feature maps for each expression image. The L2-norm of the difference between the feature maps of non-peak and peak expression images is then minimized, to embed the evolution of expressions into the PPDN framework. In this way, the PPDN incorporates the peak-piloted feature transformation and facial expression recognition into a unified architecture. The PPDN is learned with a new back-propagation algorithm, denotes peak gradient suppression (PGS), which drives the feature responses to non-peak expression instances towards those of the corresponding peak expression images, but not the contrary. This is unlike the traditional optimization of Siamese networks [13], which encourages the feature pairs to be close to each other, treating the feature maps of the two images equally. Instead, the PPDN focuses on transforming the features of non-peak expressions towards those of peak expressions. This is implemented by, during each back-propagation iteration, ignoring the gradient information due to the peak expression image in

the L2-norm minimization of feature differences, while keeping that due to the non-peak expression. The gradients of the recognition loss, for both peak and non-peak expression images, are the same as in traditional back-propagation. This avoids the degradation of the recognition capability of the network for samples of peak expression due to the influence of non-peak expression samples.

Overall, this work has four main contributions. (1) The PPDN architecture is proposed, using the responses to samples of peak expression (easy samples) to supervise the responses to samples of non-peak expression (hard samples) of the same type and from the same subject. The targets of peak-piloted feature transformation and facial expression recognition, for peak and non-peak expressions, are optimized simultaneously. (2) A tailored back-propagation procedure, PGS, is proposed to drive the responses to non-peak expressions towards those of the corresponding peak expressions, while avoiding the inverse. (3) The PPDN is shown to perform intensity-invariant facial expression recognition, by effectively recognizing the most common non-peak expressions. (4) Comprehensive evaluations on several FER datasets, namely CK+ [17] and Oulu-CASIA [18], demonstrate the superiority of the PPDN over previous methods. Its generalization to other tasks is also demonstrated through state-of-the-art robust face recognition performance on the public Multi-PIE dataset [19].

2   Related Work

There have been several recent attempts to solve the facial expression recognition problem. These methods can be grouped into two categories: sequence-based and still image approaches. In the first category, sequence-based approaches [1, 7,18,20,21] exploit both the appearance and motion information from video sequences. In the second category, still image approaches [4,10,12] recognize expressions uniquely from image appearance patterns. Since still image methods are more generic, recognizing expressions in both still images and sequences, we focus on models for still image expression recognition. Among these, both hand-crafted pipelines and deep learning methods have been explored for FER. Hand-crafted approaches [10,11,22] perform three steps sequentially: feature extraction, feature selection and classification. This can lead to suboptimal recognition, due to the combination of different optimization targets.

Convolutional Neural Network (CNN) architectures [23–25] have recently shown excellent performance on face-related recognition tasks [26–28]. Methods that resort to the CNN architecture have also been proposed for FER. For example, Yu et al. [5] used an ensemble of multiple deep CNNs. Mollahosseini et al. [16] used three inception structures [24] in convolution for FER. All these methods treat expression instances of different intensities of the same subject independently. Hence, the correlations between peak and non-peak expressions are overlooked during learning. In contrast, the proposed PPDN learns to embed the evolution from non-peak to peak expressions, so as to facilitate image-based FER.

3 The Peak-Piloted Deep Network (PPDN)

In this work we introduce the PPDN framework, which implicitly learns the evolution from non-peak to peak expressions, in the FER context. As illustrated in Fig. 3, during training the PPDN takes an image pair as input. This consists of a peak and a non-peak expression of the same type and from the same subject. This image pair is passed through several convolutional and fully-connected layers, generating pairs of feature maps for each expression image. To drive the feature responses to the non-peak expression image towards those of the peak expression image, the L2-norm of the feature differences is minimized. The learning algorithm optimizes a combination of this L2-norm loss and two recognition losses, one per expression image. Due to its excellent performance on several face-related recognition tasks [29,30], the popular GoogLeNet [24] is adopted as the basic network architecture. The incarnations of the inception architecture in GoogLeNet are restricted to filters sizes 1×1, 3×3 and 5×5. In total, the GoogLeNet implements nine inception structures after two convolutional layers and two max pooling layers. After that, the first fully-connected layer produces the intermediate features with 1024 dimensions, and the second fully-connected
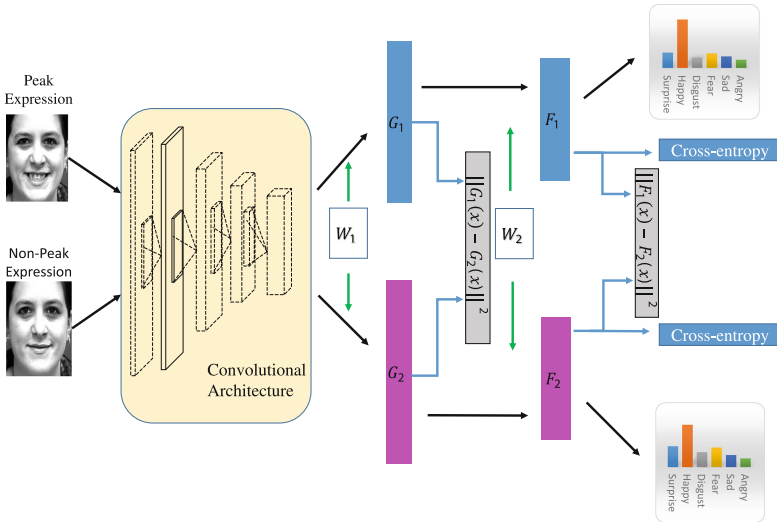
Fig. 3. Illustration of the training stage of PPDN. During training, PPDN takes the pair of peak and non-peak expression images as input. After passing the pair through several convolutional and fully-connected layers, the intermediate feature maps can be obtained for peak and non-peak expression images, respectively. The L2-norm loss between these feature maps is optimized for driving the features of the non-peak expression image towards those of the peak expression image. The network parameters can thus be updated by jointly optimizing the L2-norm losses and the losses of recognizing two expression images. During the back-propagation process, the Peak Gradient Suppression (PGS) is utilized.

layer generates the label predictions for six expression labels. During testing, the PPDN takes one still image as input, outputting the predicted probabilities for all six expression labels.

3.1   Network Optimization

The goal of the PPDN is to learn the evolution from non-peak to peak expressions, as well as recognize the basic facial expressions. We denote the training set as $S = \{x_i^p, x_i^n, y_i^p, y_i^n, i = 1, ..., N\}$, where sample x_i^n denotes a face with non-peak expression, x_i^p a face with the corresponding peak expression, and y_i^n and y_i^p are the corresponding expression labels. To supervise the feature responses to the non-peak expression instance with those of the peak expression instance, the network is learned with a loss function that includes the L2-norm of the difference between the feature responses to peak and non-peak expression instances. Cross-entropy losses are also used to optimize the recognition of the two expression images. Overall, the loss of the PPDN is

$$J = \frac{1}{N}(J_1 + J_2 + J_3 + \lambda \sum_{i=1}^{N} ||W||^2)$$

$$= \frac{1}{N} \sum_{i=1}^{N} \sum_{j \in \Omega} ||f_j(x_i^p, W) - f_j(x_i^n, W)||^2 + \frac{1}{N} \sum_{i=1}^{N} L(y_i^p, f(x_i^p; W)) \quad (1)$$

$$+ \frac{1}{N} \sum_{i=1}^{N} L(y_i^n, f(x_i^n; W)) + \lambda ||W||^2,$$

where J_1, J_2 and J_3 indicate the L2-norm of the feature differences and the two cross-entropy losses for recognition, respectively. Note that the peak-piloted feature transformation is quite generic and could be applied to the features produced by any layers. We denote Ω as the set of layers that employ the peak-piloted transformation, and $f_j, j \in \Omega$ as the feature maps in the j-th layer. To reduce the effects caused by scale variability of the training data, the features f_j are L2 normalized before the L2-norm of the difference is computed. More specifically, the feature maps f_j are concatenated into one vector, which is L2 normalized. In the second and third terms, L represents the cross-entropy loss between the ground-truth labels and the predicted probabilities of all labels. The final regularization term is used to penalize the complexity of network parameters W. Since the evolution from non-peak to peak expression is embedded into the network, the latter learns a more robust expression recognizer.

3.2   Peak Gradient Suppression (PGS)

To train the PPDN, we propose a special-purpose back-propagation algorithm for the optimization of (1). Rather than the traditional straightforward application of stochastic gradient descent [13,29], the goal is to drive the intermediate-layer responses of non-peak expression instances towards those of the corresponding

peak expression instances, while avoiding the reverse. Under traditional stochastic gradient decent (SGD) [31], the network parameters would be updated with

$$
\begin{aligned}
W^+ &= W - \gamma \nabla_W J(W; x_i^p, x_i^p, y_i^n, y_i^p) \\
&= W - \frac{\gamma}{N} \frac{\partial J_1(W; x_i^n, x_i^p)}{\partial f_j(W; x_i^n)} \times \frac{\partial f_j(W; x_i^n)}{\partial W} - \frac{\gamma}{N} \frac{\partial J_1(W; x_i^n, x_i^p)}{\partial f_j(W; x_i^p)} \times \frac{\partial f_j(W; x_i^p)}{\partial W} \\
&\quad - \frac{1}{N} \gamma \nabla_W J_2(W; x_i^p, y_i^p) - \frac{1}{N} \gamma \nabla_W J_3(W; x_i^n, y_i^n) - 2\gamma W,
\end{aligned}
\tag{2}
$$

where γ is the learning rate. The proposed peak gradient suppression (PGS) learning algorithm uses instead the updates

$$
\begin{aligned}
W^+ &= W - \frac{\gamma}{N} \frac{\partial J_1(W; x_i^n, x_i^p)}{\partial f_j(W; x_i^n)} \times \frac{\partial f_j(W; x_i^n)}{\partial W} \\
&\quad - \frac{1}{N} \gamma \nabla_W J_2(W; x_i^p, y_i^p) - \frac{1}{N} \gamma \nabla_W J_3(W; x_i^n, y_i^n) - 2\gamma W.
\end{aligned}
\tag{3}
$$

The difference between (3) and (2) is that the gradients due to the feature responses of the peak expression image, $-\frac{\gamma}{N} \frac{\partial J_1(W; x_i^n, x_i^p)}{\partial f_j(W; x_i^p)} \times \frac{\partial f_j(W; x_i^p)}{\partial W}$ are suppressed in (3). In this way, PGS drives the feature responses of non-peak expressions towards those of peak expressions, but not the contrary. In the appendix, we show that this does not prevent learning, since the weight update direction of PGS is a descent direction of the overall loss, although not a steepest descent direction.

4   Experiments

To evaluate the PPDN, we conduct extensive experiments on two popular FER datasets: CK+ [17] and Oulu-CASIA [18]. To further demonstrate that the PPDN generalizes to other recognition tasks, we also evaluate its performance on face recognition over the public Multi-PIE dataset [19].

4.1   Facial Expression Recognition

Training. The PPDN uses the GoogLeNet [24] as basic network structure. The peak-piloted feature transformation is only employed in the last two fully-connected layers. Other configurations, using the peak-piloted feature transformation on various convolutional layers are also reported. Since it is not feasible to train the deep network on the small FER datasets available, we pre-trained GoogLeNet [24] on a large-scale face recognition dataset, the CASIA Webface dataset [32]. This network was then fine-tuned for FER. The CASIA Webface dataset contains 494,414 training images from 10,575 subjects, which were used to pre-train the network for 60 epochs with an initial learning rate of 0.01. For fine-tuning, the face region was first aligned with the detected eyes and mouth positions. The face regions were then resized to 128×128. The PPDN takes a pair of peak and non-peak expression images as input. The convolutional layer

weights were initialized with those of the pre-trained model. The weights of the fully connected layer were initialized randomly using the "xaiver" procedure [33]. The learning rate of the fully connected layers was set to 0.0001 and that of pre-trained convolutional layers to 0.000001. ALL models were trained using a batch size of 128 image pairs and a weight decay of 0.0002. The final trained model was obtained after 20,000 iterations. For fair comparison with previous methods [4,10,11], we did not use any data augmentation in our experiments.

Testing and Evaluation Metric. In the testing phase, the PPDN takes one testing image as the input and produces its predicted facial expression label. Following the standard setting of [10,11], 10-fold subject-independent cross-validation was adopted for evaluation in all experiments.

Datasets. FER datasets usually provide video sequences for training and testing the facial expression recognizers. We conducted all experiments on two popular datasets, CK+ [17] and Oulu-CASIA dataset [18]. For each sequence, the face often gradually evolves from a neutral to a peak facial expression. CK+ includes six basic facial expressions (angry, happy, surprise, sad, disgust, fear) and one non basic expression (contempt). It contains 593 sequences from 123 subjects, of which only 327 are annotated with expression labels. Oulu-CASIA contains 480 sequences of six facial expressions under normal illumination, including 80 subjects between 23 and 58 years old.

Comparisons with Still Image-Based Approaches. Table 1 compares the PPDN to still image-based approaches on CK+, under the standard setting in which only the last one to three frames (i.e., nearly peak expressions) per sequence are considered for training and testing. Four state-of-the-art methods are considered: common and specific patches learning (CSPL) [10], which employs multi-task learning for feature selection, AdaGabor [34] and LBPSVM [11], which are based on AdaBoost [36], and Boosted Deep Belief Network (BDBN) [4], which jointly optimizes feature extraction and feature

Table 1. Performance comparisons on six facial expressions with four state-of-the-art methods and the baseline using GoogLeNet in terms of average classification accuracy by the 10-fold cross-validation evaluation on CK+ database.

Method	Average accuracy
CSPL [10]	89.9 %
AdaGabor [34]	93.3 %
LBPSVM [11]	95.1 %
BDBN [4]	96.7 %
GoogLeNet(baseline)	95.0 %
PPDN	**97.3 %**

Table 2. Performance comparisons on six facial expressions with UDCS method and the baseline using GoogLeNet in terms of average classification accuracy under same setting as UDCS.

Method	Average accuracy
UDCS [35]	49.5 %
GoogLeNet(baseline)	66.6 %
PPDN	**72.4 %**

selection. In addition, we also compare the PPDN to the baseline "GoogLeNet (baseline)," which optimizes the standard GoogLeNet with SGD. Similarly to previous methods [4,10,11], the PPDN is evaluated on the last three frames of each sequence. Table 2 compares the PPDN with UDCS [35] on Oulu-CASIA, under a similar setting where the first 9 images of each sequence are ignored, the first 40 individuals are taken as training samples and the rest as testing. In all cases, the PPDN input is the pair of one of the non-peak frames (all frames other than the last one) and the corresponding peak frame (the last frame) in a sequence. The PPDN significantly outperforms all other, achieving 97.3 % vs a previous best of 96.7 % on CK+ and 72.4 % vs 66.6 % on Oulu-CASIA. This demonstrates the superiority of embedding the expression evolution in the network learning.

Training and Testing with More Non-peak Expressions. The main advantage of the PPDN is its improved ability to recognize non-peak expressions. To test this, we compared how performance varies with the number of non-peak expressions. Note that for each video sequence, the face expression evolves from neutral to a peak expression. The first six frames within a sequence are usually neutral, with the peak expression appearing in the final frames. Empirically, we determined that the 7th to 9th frame often show non-peak expressions with very weak intensities, which we denote as "weak expressions." In addition to the training images used in the standard setting, we used all frames beyond the 7th for training.

Since the previous methods did not publish their codes, we only compare the PPDN to the baseline "GoogLeNet (baseline)". Table 3 reports results for CK+ and Table 4 for Oulu-CASIA. Three different test sets were considered: "weak expression" indicates that the test set only contains the non-peak expression images from the 7th to the 9th frames; "peak expression" only includes the last frame; and "combined" uses all frames from the 7th to the last. "PPDN (standard SGD)" is the version of PPDN trained with standard SGD optimization, and "GoogLeNet (baseline)" the basic GoogLeNet, taking each expression image as input and trained with SGD. The most substantial improvements are obtained on the "weak expression" test set, 83.36 % and 67.95 % of "PPDN" vs. 78.10 % and 64.64 % of "GoogLeNet (baseline)" on CK+ and Oulu-CASIA, respectively. This is evidence in support of the advantage of explicitly learning the evolution

Table 3. Performance comparison on CK+ database in terms of average classification accuracy of the 10-fold cross-validation when evaluating on three different test sets, including "weak expression", "peak expression" and "combined", respectively.

Method	Weak expression	Peak expression	combined
PPDN(standard SGD)	81.34 %	99.12 %	94.18 %
GoogLeNet (baseline)	78.10 %	98.96 %	92.19 %
PPDN	**83.36 %**	**99.30 %**	**95.33 %**

Table 4. Performance comparison on Oulu-CASIA database in terms of average classification accuracy of the 10-fold cross-validation when evaluating on three different test sets, including "weak expression", "peak expression" and "combined", respectively.

Method	Weak expression	Peak expression	combined
PPDN(standard SGD)	67.05 %	82.91 %	73.54 %
GoogLeNet (baseline)	64.64 %	79.21 %	71.32 %
PPDN	**67.95 %**	**84.59 %**	**74.99 %**

from non-peak to peak expressions. In addition, the PPDN outperforms "PPDN (standard SGD)" and "GoogLeNet (baseline)" on the combined sets, where both peak and non-peak expressions are evaluated.

Comparisons with Sequence-Based Approaches. Unlike the still-image recognition setting, which evaluates the predictions of frames from a sequence, the sequence-based setting requires a prediction for the whole sequence. Previous sequence-based approaches take the whole sequence as input and use motion information during inference. Instead, the PPDN regards each pair of non-peak and peak frame as input, and only outputs the label of the peak frame as prediction for the whole sequence, in the testing phase. Tables 5 and 6 compare the PPDN to several sequence-based approaches plus "GoogLeNet(baseline)" on CK+ and Oulu-CASIA. Compared with [1,7,37], which leverage motion information, the PPDN, which only relies on appearance information, achieves significantly better prediction performance. On CK+, it has gains of 5.1 % and 2 % over 'STM-ExpLet' [1] and "DTAGN(Joint)" [7]. On Oulu-CASIA it achieves 84.59 % vs. the 75.52 % of "Atlases" [20] and the 81.46 % of "DTAGN(Joint)" [7]. In addition, we evaluate this experiment without peak information, i.e. selecting image with highest classification scores for all categories as peak frame in testing. PPDN achieves 99.2 % on CK+ and 83.67 % on Oulu-CASIA.

PGS vs. Standard SGD. As discussed above, PGS suppresses gradients from peak expressions, so as to drive the features of non-peak expression samples towards those of peak expression samples, but not the contrary. Standard SGD uses all gradients, due to both non-peak and peak expression samples. We

Table 5. Performance comparisons with three sequence-based approaches and the baseline "GoogLeNet (baseline)" in terms of average classification accuracy of the 10-fold cross-validation on CK+ database.

Method	Experimental settings	Average accuracy
3DCNN-DAP [37]	sequence-based	92.4 %
STM-ExpLet [1]	sequence-based	94.2 %
DTAGN(Joint) [7]	sequence-based	97.3 %
GoogLeNet (baseline)	image-based	99.0 %
PPDN (standard SGD)	image-based	99.1 %
PPDN w/o peak	image-based	**99.2 %**
PPDN	image-based	**99.3 %**

Table 6. Performance comparisons with five sequence-based approaches and the baseline "GoogLeNet (baseline)" in terms of average classification accuracy of the 10-fold cross-validation on Oulu-CASIA.

Method	Experimental settings	Average accuracy
HOG 3D [21]	sequence-based	70.63 %
AdaLBP [18]	sequence-based	73.54 %
Atlases [20]	sequence-based	75.52 %
STM-ExpLet [1]	sequence-based	74.59 %
DTAGN(Joint) [7]	sequence-based	81.46 %
GoogLeNet (baseline)	image-based	79.21 %
PPDN (standard SGD)	image-based	82.91 %
PPDN w/o peak	image-based	**83.67 %**
PPDN	image-based	**84.59 %**

hypothesized that this will degrade recognition for samples of peak expressions, due to interference from non-peak expression samples. This hypothesis is confirmed by the results of Tables 3 and 4. PGS outperforms standard SGD on all three test sets.

Ablative Studies on Peak-Piloted Feature Transformation. The peak-piloted feature transformation, which is the key innovation of the PPDN, can be used on all layers of the network. Employing the transformation on different convolutional and fully-connected layers can result in different levels of supervision of non-peak responses by peak responses. For example, early convolutional layers extract fine-grained details (e.g., local boundaries or illuminations) of faces, while later layers capture more semantic information, e.g., the appearance pattens of mouths and eyes. Table 7 presents an extensive comparison, by adding peak-piloted feature supervision on various layers. Note that we employ GoogLeNet [24], which includes 9 inception layers, as basic network. Four dif-

Table 7. Performance comparisons by adding the peak-piloted feature transformation on different convolutional layers when evaluated on Oulu-CASIA dataset.

Method	inception layers	the last FC layer	the first FC layer	both FC layers
Inception-3a	✓	✕	✕	✕
Inception-3b	✓	✕	✕	✕
Inception-4a	✓	✕	✕	✕
Inception-4b	✓	✕	✕	✕
Inception-4c	✓	✕	✕	✕
Inception-4d	✓	✕	✕	✕
Inception-4e	✓	✕	✕	✕
Inception-5a	✓	✕	✕	✕
Inception-5b	✓	✕	✕	✕
Fc1	✓	✕	✓	✓
Fc2	✓	✓	✕	✓
Average accuracy	74.49 %	73.33 %	73.48 %	74.99 %

Table 8. Comparisons of the version with and without using peak information on Oulu-CASIA database in terms of average classification accuracy of the 10-fold cross-validation.

Method	Weak expression	Peak expression	Combined
PPDN w/o peak	67.52 %	83.79 %	74.01 %
PPDN	**67.95 %**	**84.59 %**	**74.99 %**

Table 9. Face recognition rates for various poses under "setting 1".

Method	$-45°$	$-30°$	$-15°$	$+15°$	$+30°$	$+45°$	Average
GoogLeNet (baseline)	86.57 %	99.3 %	100 %	100 %	100 %	90.06 %	95.99 %
PPDN	**93.96 %**	**100 %**	**100 %**	**100 %**	**100 %**	**93.96 %**	**97.98 %**

Table 10. Face recognition rates for various poses under "setting 2".

Method	$-45°$	$-30°$	$-15°$	$+15°$	$+30°$	$+45°$	Average
Li et al. [38]	56.62 %	77.22 %	89.12 %	88.81 %	79.12 %	58.14 %	74.84 %
Zhu et al. [27]	67.10 %	74.60 %	86.10 %	83.30 %	75.30 %	61.80 %	74.70 %
CPI [28]	66.60 %	78.00 %	87.30 %	85.50 %	75.80 %	62.30 %	75.90 %
CPF [28]	73.00 %	81.70 %	89.40 %	89.50 %	80.50 %	70.30 %	80.70 %
GoogLeNet (baseline)	56.62 %	77.22 %	89.12 %	88.81 %	79.12 %	58.14 %	74.84 %
PPDN	**72.06 %**	**85.41 %**	**92.44 %**	**91.38 %**	**87.07 %**	**70.97 %**	**83.22 %**

ferent settings are tested: "inception layers" indicates that the loss of the peak-piloted feature transformation is appended for all inception layers plus the two fully-connected layers; "the first FC layer," "the last FC layer" and "both FC layers" append the loss to the first, last, and both fully-connected layers, respectively.

It can be seen that using the peak-piloted feature transformation only on the two fully connected layers achieves the best performance. Using additional losses on all inception layers has roughly the same performance. Eliminating the loss of a fully-connected layer decreases performance by more than 1 %. These results show that the peak-piloted feature transformation is more useful for supervising the highly semantic feature representations (two fully-connected layers) than the early convolutional layers.

Absence of Peak Information. Table 8 demonstrates that the PPDN can also be used when the peak frame is not known a priori, which is usually the case for real-world videos. Given all video sequences, we trained the basic "GoogLeNet (baseline)" with 10-fold cross validation. The models were trained with 9-folds and then used to predict the ground-truth expression label in the remaining fold. The frame with the highest prediction score in each sequence was treated as the peak expression image. The PPDN was finally trained using the strategy of the previous experiments. This training procedure is more applicable to videos where the information of the peak expression is not available. The PPDN can still obtain results comparable to those of the model trained with the ground-truth peak frame information.

4.2   Generalization Ability of the PPDN

The learning of the evolution from a hard sample to an easy sample is applicable to other face-related recognition tasks. We demonstrate this by evaluating the PPDN on face recognition. One challenge to this task is learning robust features, invariant to pose and view. In this case, near-frontal faces can be treated easy examples, similar to peak expressions in FER, while profile faces can be viewed as hard samples, similar to non-peak expressions. The effectiveness of PPDN in learning pose-invariant features is demonstrated by comparing PPDN features to the "GoogLeNet(baseline)" features on the popular Multi-PIE dataset [19].

All the following experiments were conducted on the images of "session 1" on Multi-PIE, where the face images of 249 subjects are provided. Two experimental settings were evaluated to demonstrate the generalization ability of PPDN on face recognition. For the "setting 1" of Table 9, only images under normal illumination were used for training and testing, where seven poses of the first 100 subjects (ID from 001 to 100) were used for training and the six poses (from $-45°$ to $45°$) of the remaining individuals used for testing. One frontal face per subject was used as gallery image. Overall, 700 images were used for training and 894 images for testing. By treating the frontal face and one of the profile faces as input, the PPDN can embed the implicit transformation from profile

faces to frontal faces into the network learning, for face recognition purposes. In the "setting 2" of Table 10, 100 subjects (ID 001 to 100) with seven different poses under 20 different illumination conditions were used for training and the rest with six poses and 19 illumination conditions were used for testing. This led to 14,000 training images and 16,986 testing images. Similarly to the first setting, PPDN takes the pair of a frontal face with normal illumination and one of the profile faces with 20 illuminations from the same subject as the input. The PPDN can thus learn the evolution from both the profile to the frontal face and non-normal to normal illumination. In addition to "GoogLeNet (baseline)," we compared the PPDN to four state-of-the-art methods: controlled pose feature(CPF) [28], controlled pose image(CPI) [28], Zhu et al. [27] and Li et al. [38]. The pre-trained model, prepocessing steps, and learning rate used in the FER experiments were adopted here. Under "setting 1" the network was trained with 10,000 iterations and under "setting 2" with 30,000 iterations. Face recognition performance is measured by the accuracy of the predicted subject identity.

It can be seen that the PPDN achieves considerable improvements over "GoogLeNet (baseline)" for the testing images of hard poses (i.e., $-45°$ and $45°$) in both "setting 1" and "setting 2". Significant improvements over "GoogLeNet (baseline)" are also observed for the average over all poses (97.98 % vs 95.99 % under "setting 1" and 83.22 % vs 74.84 % under "setting 2"). The PPDN also beats all baselines by 2.52 % under "setting 2". This supports the conclusion that the PPDN can be effectively generalized to face recognition tasks, which benefit from embedding the evolution from hard to easy samples into the network parameters.

5   Conclusions

In this paper, we propose a novel peak-piloted deep network for facial expression recognition. The main novelty is the embedding of the expression evolution from non-peak to peak into the network parameters. PPDN jointly optimizes an L2-norm loss of peak-piloted feature transformation and the cross-entropy losses of expression recognition. By using a special-purpose back-propagation procedure (PGS) for network optimization, the PPDN can drive the intermediate-layer features of the non-peak expression sample towards those of the peak expression sample, while avoiding the inverse.

Appendix

The loss

$$J_1 = \sum_{i=1}^{N} \sum_{j \in \Omega} \| f_j(x_i^p, W) - f_j(x_i^n, W) \|^2 \tag{A-1}$$

has gradient

$$\nabla_W J_1 = 2 \sum_{i=1}^{N} \sum_{j \in \Omega} (f_j(x_i^p, W) - f_j(x_i^n, W)) \nabla_W f_j(x_i^n, W)$$
$$+ 2 \sum_{i=1}^{N} \sum_{j \in \Omega} (f_j(x_i^p, W) - f_j(x_i^n, W)) \nabla_W f_j(x_i^p, W). \tag{A-2}$$

The PGS is

$$\widetilde{\nabla_W J_1} = 2 \sum_{i=1}^{N} \sum_{j \in \Omega} (f_j(x_i^p, W) - f_j(x_i^n, W)) \nabla_W f_j(x_i^n, W) \tag{A-3}$$

Defining

$$A = \sum_{i=1}^{N} \sum_{j \in \Omega} (f_j(x_i^p, W) - f_j(x_i^n, W)) \nabla_W f_j(x_i^n, W) \tag{A-4}$$

and

$$B = \sum_{i=1}^{N} \sum_{j \in \Omega} (f_j(x_i^p, W) - f_j(x_i^n, W)) \nabla_W f_j(x_i^p, W) \tag{A-5}$$

it follows that

$$< \nabla_W J_1, \widetilde{\nabla_W J_1} > = -4 < A, B > + 4\|A\|^2 \tag{A-6}$$

or

$$< \nabla_W J_1, \widetilde{\nabla_W J_1} > = -4\|A\|\|B\|\cos\theta + 4\|A\|^2 \tag{A-7}$$

where θ is angle between A and B. Hence, the dot-product is greater than zero when

$$\|B\|\cos\theta < \|A\|. \tag{A-8}$$

This holds for sure as $\nabla_W f_j(x_i^n, W)$ converges to $\nabla_W f_j(x_i^p, W)$ which is the goal of optimization, but is generally true if the sizes of gradients $\nabla_W f_j(x_i^n, W)$ and $\nabla_W f_j(x_i^p, W)$ are similar on average. Since the dot-product is positive, $\widetilde{\nabla_W J_1}$ is a descent (although not a steepest descent) direction for the loss function J_1. Hence, the PGS is a descent direction for the total loss. Note that, because there are also the gradients of J_2 and J_3, this can hold even when (A-8) is violated, if the gradients of J2 and J3 are dominant. Hence, the PGS is likely to converge to a minimum of the loss.

References

1. Liu, M., Shan, S., Wang, R., Chen, X.: Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1749–1756 (2014)

2. Chen, H., Li, J., Zhang, F., Li, Y., Wang, H.: 3D model-based continuous emotion recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1836–1845 (2015)

3. Dapogny, A., Bailly, K., Dubuisson, S.: Pairwise conditional random forests for facial expression recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3783–3791 (2015)

4. Liu, P., Han, S., Meng, Z., Tong, Y.: Facial expression recognition via a boosted deep belief network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1805–1812 (2014)

5. Yu, Z., Zhang, C.: Image based static facial expression recognition with multiple deep network learning. In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, pp. 435–442. ACM (2015)

6. Liu, M., Li, S., Shan, S., Chen, X.: Au-aware deep networks for facial expression recognition. In: 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), pp. 1–6. IEEE (2013)

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

8. Li, X., Mori, G., Zhang, H.: Expression-invariant face recognition with expression classification. In: 2006 3rd Canadian Conference on Computer and Robot Vision, p. 77. IEEE (2006)

9. Zhang, Z., Luo, P., Loy, C.C., Tang, X.: Facial landmark detection by deep multi-task learning. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VI. LNCS, vol. 8694, pp. 94–108. Springer, Heidelberg (2014)

10. Zhong, L., Liu, Q., Yang, P., Liu, B., Huang, J., Metaxas, D.N.: Learning active facial patches for expression analysis. In: Computer Vision and Pattern Recognition (CVPR), pp. 2562–2569. IEEE (2012)

11. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: a comprehensive study. Image Vis. Comput. **27**(6), 803–816 (2009)

12. Kahou, S.E., Froumenty, P., Pal, C.: Facial expression analysis based on high dimensional binary features. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926, pp. 135–147. Springer, Heidelberg (2014)

13. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 539–546. IEEE (2005)

14. Lai, H., Xiao, S., Cui, Z., Pan, Y., Xu, C., Yan, S.: Deep cascaded regression for face alignment (2015). arXiv preprint arXiv:1510.09083

15. Li, H., Lin, Z., Shen, X., Brandt, J., Hua, G.: A convolutional neural network cascade for face detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5325–5334 (2015)

16. Mollahosseini, A., Chan, D., Mahoor, M.H.: Going deeper in facial expression recognition using deep neural networks (2015). arXiv preprint arXiv:1511.04110

17. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended cohn-kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 94–101. IEEE (2010)

18. Zhao, G., Huang, X., Taini, M., Li, S.Z., Pietikäinen, M.: Facial expression recognition from near-infrared videos. Image Vis. Comput. **29**(9), 607–619 (2011)

19. Gross, R., Matthews, I., Cohn, J., Kanade, T., Baker, S.: Multi-pie. Image Vis. Comput. **28**(5), 807–813 (2010)
20. Guo, Y., Zhao, G., Pietikäinen, M.: Dynamic facial expression recognition using longitudinal facial expression atlases. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 631–644. Springer, Heidelberg (2012)
21. Klaser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3D-gradients. In: BMVC 2008–19th British Machine Vision Conference, British Machine Vision Association p. 275–1 (2008)
22. Sikka, K., Wu, T., Susskind, J., Bartlett, M.: Exploring bag of words architectures in the facial expression domain. In: Fusiello, A., Murino, V., Cucchiara, R. (eds.) ECCV 2012 Ws/Demos, Part II. LNCS, vol. 7584, pp. 250–259. Springer, Heidelberg (2012)
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)
24. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
25. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014). arXiv preprint arXiv:1409.1556
26. Sun, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: Advances in Neural Information Processing Systems, pp. 1988–1996 (2014)
27. Zhu, Z., Luo, P., Wang, X., Tang, X.: Deep learning identity-preserving face space. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 113–120 (2013)
28. Yim, J., Jung, H., Yoo, B., Choi, C., Park, D., Kim, J.: Rotating your face using multi-task deep neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 676–684 (2015)
29. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: a unified embedding for face recognition and clustering. In: IEEE CVPR, pp. 815–823 (2015)
30. Sun, Y., Liang, D., Wang, X., Tang, X.: Deepid3: face recognition with very deep neural networks (2015). arXiv preprint arXiv:1502.00873
31. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010, pp. 177–186. Springer, Heidelberg (2010)
32. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch (2014). arXiv preprint arXiv:1411.7923
33. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: International Conference on Artificial Intelligence and Statistics, pp. 249–256 (2010)
34. Bartlett, M.S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., Movellan, J.: Recognizing facial expression: machine learning and application to spontaneous behavior. In: CVPR, vol. 2, pp. 568–573. IEEE (2005)
35. Xue, M., W.L., Li, L.: The uncorrelated and discriminant colour space for facial expression recognition. In: Optimization and Control Techniques and Applications, pp. 167–177 (2014)
36. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. In: Proceedings of the Second European Conference on Computational Learning Theory, pp. 23–27 (1995)

37. Liu, M., Li, S., Shan, S., Wang, R., Chen, X.: Deeply learning deformable facial action parts model for dynamic expression analysis. In: Cremers, D., Reid, I., Saito, H., Yang, M.-H. (eds.) ACCV 2014. LNCS, vol. 9006, pp. 143–157. Springer, Heidelberg (2015)
38. Li, A., Shan, S., Gao, W.: Coupled bias-variance tradeoff for cross-pose face recognition. IEEE Trans. Image Process. **21**(1), 305–315 (2012)