# Augmented Feature Representation with Parallel Convolution for Cross-domain Facial Expression Recognition

Fan Yang, Weicheng Xie (✉), Tao Zhong, Jingyu Hu, and Linlin Shen

Computer Vision Institute, School of Computer Science and Software Engineering,
Shenzhen University
Shenzhen Institute of Artificial Intelligence and Robotics for Society
Guangdong Key Laboratory of Intelligent Information Processing
{yangfan2021,zhongtao2021,2110276105}@email.szu.edu.cn
{wcxie,llshen}@szu.edu.cn

**Abstract.** Facial expression recognition (FER) has made significant progress in the past decade, but the inconsistency of distribution between different datasets greatly limits the generalization performance of a learned model on unseen datasets. Recent works resort to aligning feature distributions between domains to improve the cross-domain recognition performance. However, current algorithms use one output each layer for the feature representation, which can not well represent the complex correlation among multi-scale features. To this end, this work proposes a parallel convolution to augment the representation ability of each layer, and introduces an orthogonal regularization to make each convolution represent independent semantic. With the assistance of a self-attention mechanism, the proposed algorithm can generate multiple combinations of multi-scale features to allow the network to better capture the correlation among the outputs of different layers. The proposed algorithm achieves state-of-the-art (SOTA) performances in terms of the average generalization performance on the task of cross-database (CD)-FER. Meanwhile, when AFED or RAF-DB is used for the training, and other four databases, i.e. JAFFE, SFEW, FER2013 and EXPW are used for testing, the proposed algorithm outperforms the baselines by the margins of 5.93% and 2.24% in terms of the average accuracy.

**Keywords:** Domain generalization, Parallel convolution, Facial expression recognition, Self-attention

## 1 Introduction

Facial expression recognition (FER) is beneficial to understand human emotions and behaviors, which is widely applied in emotional computing, fatigue detection and other fields. Over the last decade, people have proposed deep learning architectures and collected a large number of datasets, which greatly facilitates the study of FER. However, people interpret facial expressions differently, their annotations to the dataset are inevitably subjective. This leads to a relatively large domain shift between different datasets, and the difference in the collection scenes and object styles will also greatly increase this shift gap, which will greatly impair the performance of the model on unseen datasets.

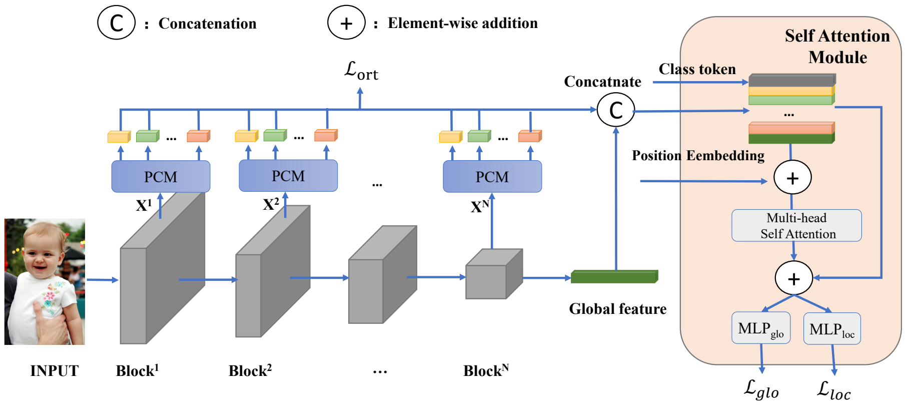


**Fig. 1.** An overview of our algorithm. $N$ denote the number of blocks. $X_i$ denote the feature maps output by $Block^i$. PCM denotes the Parallel Convolution Module in Fig. 2 (b). Both $MLP_{glo}$ and $MLP_{loc}$ consist of a fully connected layer.

Recently, many works try to learn domain-invariant features to reduce this domain shift. Chen et al. [2] argue that some local features in facial expressions are beneficial to Cross-Domain Facial Expression Recognition (CD-FER) because these local features are easier to transfer across different datasets, can represent more detailed information that is beneficial to fine-grained adaptation. However, these CD-FER algorithms employ unique output block for the feature representation of each layer, even the fusion of these outputs is unable to sufficiently encode the complex correlation among them.

In this work, we introduce a simple yet effective structure that only needs to use parallel convolution operations from different layers to extract rich hierarchical information. These features can help improve the generalization performance of the network on unseen datasets without affecting the discriminative performance on the source domain. Compared with other methods with unique convolution output, the proposed parallel convolution module augment the feature representation, and better capture the correlation among different scales of features from various layers, which is critical for the transferability ability of a recognition network.

Our main contributions are summarized as follows

- We introduce a novel parallel convolution to augment the feature representation of each layer, and a specific orthogonal loss to enhance the independence of branches for representing different semantics.

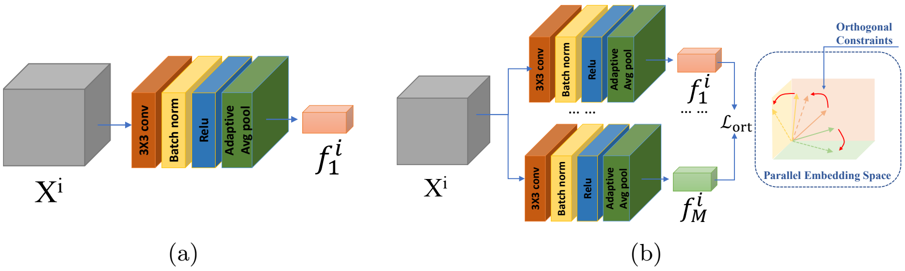
– We propose a hierarchical feature representation based on the multi-head self-attention module for cross-database FER, by modeling the complex correlation of the features from different layers with the combinations of multiple-scale features.
– By comparing the state of the arts on the task of cross-database FER, our method achieve state-of-the-art performances in terms of the average generalization performance.

## 2 Proposed Method

In this section, we introduce the proposed framework in Fig. 1, which mainly consists of three parts, i.e. a backbone network for representing the global discriminative features, a parallel convolution module in Sec. 2.1 used to extract features at different levels, and a multi-head self-attention module in Sec. 2.2 used to capture the correlation information between global discriminative and auxiliary features.

### 2.1 Parallel Convolution Module (PCM)

While one convolution output can effectively encode the expression hidden semantics with highly nonlinear representation, e.g. local variation of geometry structure and texture, it may not work well for the representation of the in-the-wild expression samples, which often include largely occluded or posed faces [2]. This challenge motivates us to construct multiple convolution outputs to represent the complex semantics implied in these samples.



(a) (b)

**Fig. 2.** (a) Original: One feature representation (b) Ours: Parallel Convolution Module (PCM). $M$ denote the number of Parallel Feature Outputs of each network block.

Specifically, we introduce a parallel structure in Fig. 2(b) to capture multi-branch features for each block of the backbone network, which is formulated as follows

$$f_j^i = \phi(\sigma(Norm(Conv(X^i)))) \tag{1}$$

where $i$ and $j$ denote the depth of blocks and number of parallel features, respectively. $\sigma$ and $\phi$ denote Relu activation function and the adaptive average pooling layer, respectively. There are two merits to using such a structure. First, parallel features can be generated by adding only a few network parameters, which does not sacrifice the training speed of the network; The second is that convolution can well capture the local information of features, such local information is more transferable for the task of FER.

In order to reduce the entanglement among the outputs of the parallel convolution, so as to enable each parallel output to learn specific semantic and improve the generalization ability of the feature representation, we further introduce the regularization term of orthogonalization as follows

$$\mathcal{L}_{ort} = -\frac{1}{B}\sum_{i=1}^{B}\sum_{m=1}^{M}\sum_{n=1}^{M}(f_m^i)^T f_n^i \tag{2}$$

where $f_m^i$ denotes the $m$-th parallel feature from the $i$-th block for each sample.

### 2.2 Multi-Head Self-Attention Module (MSH)

It is revealed in [1] that the features extracted by the network become more task-specific as the depth of the neural network increases. That is, the shallow layers often represent some relatively similar features which may have better transferability, while the deep layers will encode the features for specific tasks. Based on this, as shown in Fig. 1, we design a cascaded module to leverage hierarchical features from different depths to help improve its generalization ability.

Visual transformers (ViT) [3] can well capture global information in images with global receptive fields, and can build the interaction between global patches with the self-attention mechanism. Based on this framework, we resort to aggregating the multi-scale features from different network blocks with the parallel convolution, rather than using the sequence features of split patch embedding in the original ViT. Specifically, we use the multi-head self-attention module to enhance the information representation of parallel features, positional encoding to assist the learning of positional information between different parallel features, and a learnable class token to label the specific features.

As shown in Fig. 1, we use a self-attention module to augment the information of parallel features. Since the classification token summarizes the global information of other features, and it does not depend on the input information, thus can avoid the preference for a certain parallel information and help the model to improve its generalization performance.

Formally, we first concatenate all features as follows

$$F = ((f^1 \oplus f^2 \oplus ...f^M) \oplus f^g \oplus f^c) + f^p \tag{3}$$

where $M$ denotes the number of parallel features for each layer, $f^g$ denotes the global feature and the $f^c$ is a learnable classification token. Matrices $f^i$, $f^g$, $f^c$

are with the dimension of $(B, M \times D)$, where $B$ and $D$ denote the batch size and the feature dimension, respectively. Matrix $f^p$ is a learnable embedding with the dimension of $(B, (M+2) \times D)$ for describing the location information of the features. Then, we use a multi-head attention module to capture the key features, while integrating information from all features. Specifically, we transformed the feature $F$ into queries $q$, keys $k$ and values $v$ as follows

$$[q, k, v] = F[W_q, W_k, W_v] \tag{4}$$

To aggregate these features, the attention weights are adjusted as follows

$$A = \varepsilon(\frac{qk^T}{\sqrt{d}}) \tag{5}$$

where $\varepsilon$ denote the Softmax function, and $d$ denote the dimension of feature. Finally, the output of self-attention can be obtained as follows

$$F^{'} = Av + F \tag{6}$$

where $v$ is the value in Eq. (4).

### 2.3 Joint training loss

Based on the features with the self-attention model, i.e. $F^{'}$ in Eq. (6), the classification probabilities are formulated as follows

$$p_{i,c}^{loc} = \varepsilon(MLP_{loc}(F_{loc}^{'})) \tag{7}$$

$$p_{i,c}^{glo} = \varepsilon(MLP_{glo}(F^{'})) \tag{8}$$

where $F_{loc}^{'}$ is the feature output by the self attention module of $f^1 \oplus f^2 \oplus ...f^M$ in Eq. (3) and is a part of $F^{'}$. Finally, the two classification losses in Fig. 1 are then formulated as follows

$$\mathcal{L}_{loc} = -\frac{1}{B}\sum_{i=1}^{B}\sum_{c=1}^{K} y_{i,c} log(p_{i,c}^{loc}) \tag{9}$$

$$\mathcal{L}_{glo} = -\frac{1}{B}\sum_{i=1}^{B}\sum_{c=1}^{K} y_{i,c} log(p_{i,c}^{glo}) \tag{10}$$

where $K$ denotes the number of expression classes. $p_{i,c}^{loc}$ and $p_{i,c}^{glo}$ are the predicted probabilities of the $c$-th class specific to the local and global branches. The total loss is then formulated as follows

$$\mathcal{L} = \mathcal{L}_{glo} + \lambda\mathcal{L}_{loc} + \gamma\mathcal{L}_{ort} \tag{11}$$

where $\lambda$ and $\gamma$ are set as 1 in this work.

# 3 Experimental Results

## 3.1 Implementation Details

We use six mainstream facial expression datasets for the evaluation, follow the protocol as [2], and use the IResNet50 pretrained on the MS-Celeb-1M [13] as the backbone. The setting of the parameters specific to the newly added layers follow the Xavier algorithm [12]. For the parallel convolution in Fig. 2(b), $M$ is set as 5, the feature dimension of each parallel convolution output, i.e. $D$, is set to 16. For global feature representation, i.e. $f^g$ in Eq. (3), another convolution operation and pooling layer are performed to encode a feature vector with the dimension of $N \times D = 64$.

**Table 1.** Comparison of cross-database performances. The results are reproduced by our implementation with exactly the same source dataset, backbone network and pre-trained model. The best and 2nd best performances are labeled with bold and underline. $*$ or $\dagger$ denotes the results that are implemented by us or cited from [2].

| Method | Source set | JAFFE | SFEW | FER2013 | EXPW | Mean | Reference |
|---|---|---|---|---|---|---|---|
| *Baseline*$^*$ | RAF | 52.58 | 51.60 | **57.89** | 70.09 | 58.04 | - |
| *ICID*$^\dagger$ [14] | RAF | 50.57 | 48.85 | 53.70 | 69.54 | 55.66 | Neurocomputing2019 |
| *LPL*$^\dagger$ [15] | RAF | 53.05 | 48.85 | 55.89 | 66.90 | 56.17 | CVPR2017 |
| *FTDNN*$^\dagger$ [16] | RAF | 52.11 | 47.48 | 55.98 | 67.72 | 55.82 | SIBGRAPI2017 |
| *SAFN*$^\dagger$ [17] | RAF | **61.03** | **52.98** | 55.64 | 64.91 | 58.64 | CVPR2019 |
| *AGRA*$^*$ [2] | RAF | 58.68 | 51.37 | 57.49 | 70.73 | 59.53 | TPAMI2021 |
| *Ours*$^*$ | RAF | 59.15 | 52.75 | 57.82 | **71.42** | **60.28** | - |
| *Baseline*$^*$ | AFED | 57.74 | 47.25 | 46.55 | 49.50 | 50.26 | - |
| *ICID*$^\dagger$ [14] | AFED | 57.28 | 44.27 | 46.92 | 52.91 | 50.34 | Neurocomputing2019 |
| *LPL*$^\dagger$ [15] | AFED | 61.03 | 49.77 | 49.54 | 55.26 | 53.9 | CVPR2017 |
| *FTDNN*$^\dagger$ [16] | AFED | 57.75 | 47.25 | 46.36 | 52.89 | 51.06 | SIBGRAPI2017 |
| *SAFN*$^\dagger$ [17] | AFED | 64.79 | 49.08 | 48.89 | 55.69 | 54.61 | CVPR2019 |
| *AGRA*$^*$ [2] | AFED | **65.25** | 48.16 | 49.73 | 51.56 | 53.67 | TPAMI2021 |
| *Ours*$^*$ | AFED | 62.44 | **52.29** | **51.43** | **58.62** | **56.19** | - |

## 3.2 Comparison with the state of the arts

To study the generalization performance of our method on unseen datasets, we use RAF [7] or AFED [2] as source domain dataset, and JAFFE, SFEW2.0 [4], FER2013 [5] and EXPW [6] are used as target datasets, while only the source domain dataset are used for the training. The results are shown in Tab. 1.

Tab. 1 shows that our method achieved the best performances among six state-of-the-art algorithms in terms of the mean accuracy. For each target dataset, our algorithm either achieves the best performance, or ranks the 2nd. Meanwhile, compared with the baseline, the proposed algorithm achieved the improvement of 2.24% or 5.93% when RAF or AFED is used as source dataset. Specifically, our algorithm appears to be more effective than the competitors on the target datasets with larger number of samples, e.g. FER2013 and EXPW.

### 3.3 Algorithm analysis

In this section, we first perform ablation study in Tab. 2 to analyze the role of each module, where we can see from the 1st and 2nd rows that PCM help the model achieve an improvement of 4.56% over the baseline in terms of the average generalization performance. It is revealed in the 2nd-4th rows that using only the self-attention mechanism may affect the generalization ability. It pays attention to the features that help improve the discrimination performance on the original domain, while affecting the generalization performance on the target domain. The learnable classification token with a fixed position can effectively integrate features between different levels, and it is not biased towards a certain feature, thus can help the model improve the generalization ability.

**Table 2.** The results of ablation study. CLS-token denotes a learnable embedding for integrating information of different features in Eq. (3). OrtLoss is the regularization loss in Eq. (2)

| Backbone | PCM | MSH | CLS-token | OrtLoss | JAFFE | SFEW | FER2013 | EXPW | Mean |
|---|---|---|---|---|---|---|---|---|---|
| IResNet50 | ✗ | ✗ | ✗ | ✗ | 57.74 | 47.25 | 46.55 | 49.50 | 50.26 |
| IResNet50 | ✓ | ✗ | ✗ | ✗ | **63.84** | 48.16 | 49.73 | 57.66 | 54.84 |
| IResNet50 | ✓ | ✓ | ✗ | ✗ | 56.80 | 47.25 | 49.93 | 56.75 | 52.68 |
| IResNet50 | ✓ | ✓ | ✓ | ✗ | 61.97 | 47.93 | **52.38** | **60.29** | 55.64 |
| IResNet50 | ✓ | ✓ | ✓ | ✓ | 62.44 | **52.29** | 51.43 | 58.62 | **56.19** |

**Table 3.** Parameter sensitivity analysis for the parallel convolution based on the training of AFED. 1* denotes one parallel branch with the feature dimension being the dimension sum of features from the five parallel branches.

| Backbone | $M$ | JAFFE | SFEW | FER2013 | EXPW | Mean |
|---|---|---|---|---|---|---|
| IResNet50 | 1 | 61.03 | 47.70 | 49.59 | 57.13 | 53.86 |
| IResNet50 | 3 | **62.91** | 47.47 | 48.98 | 56.83 | 54.04 |
| IResNet50 | 5 | 61.97 | **47.93** | **52.38** | **60.29** | **55.64** |
| IResNet50 | 7 | 59.15 | 47.70 | 50.34 | 56.70 | 53.47 |
| IResNet50 | 1* | 60.56 | 45.87 | 49.23 | 55.40 | 52.76 |

In order to give insight into the features learned by our method during the training process. We visualize how the features of different domains evolve as training progresses. Specifically, we simultaneously input samples from different domains into the network to obtain features, use t-SNE to project them into the 2D space, and present the results in Fig. 3. It shows that the baseline model can separate the source samples, while it can not well distinguish the data of the target domain. As shown in the bottom row of Fig. 3, our algorithm obtain features that are still separable in the feature space. More importantly, samples from different domains are more concentrated compared with the baseline, which means that the learned features can be made have similar distributions in different domains, by better learning the complex correlation among features from different layers, thereby yielding more powerful generalization ability.

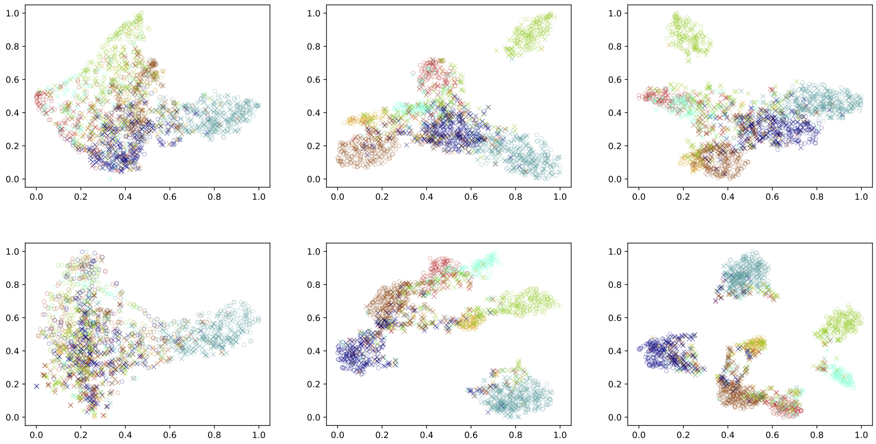


**Fig. 3.** Illustration of the feature distributions learned by IResNet (upper, baseline) and our algorithm (lower) at epochs of 1, 21 and 51 in the 1st-3rd columns, respectively. 'o' denotes the sample from the source domain (RAF-DB), 'x' denotes the sample from the target domain (SFEW2.0). Different colors represent different labels.

We also study the performance sensitivity against the number of Parallel Convolution branches, i.e. $M$ in Fig. 2, the results are shown in Tab. 3. Tab. 3 shows that the setting of $M = 5$ achieves the best average performance. While too few parallel outputs can not sufficiently capture the rich hierarchical information among different layers, too many outputs increase the possibility of feature entanglement, which may decrease the cross-database generalization performance. To study whether the improvement is resulted from the dimensional ascension by the parallel convolution, we evaluate the performance of a specific setting, i.e. the feature dimension is set as the same as that of the proposed convolution, in the last row of Tab. 3. These results show that the improvements are not resulted from the mere dimensional ascension.

In order to give insight into the working mechanism of the proposed parallel convolution, we visualize the heatmaps output by the parallel convolution in Fig. 4, where the heatmaps with the similar semantics are gathered in the same column with an alignment. Fig. 4 shows that the heatmaps in the same column appear with the similar semantic, while the outputs of different parallel branches shows with diverse and independent semantics. When the parallel convolution is performed, semantic alignment is actually not employed. In this case, the random combinations of independent semantics can thus enhance the feature representation ability for in-the-wild circumstances with complex semantics.

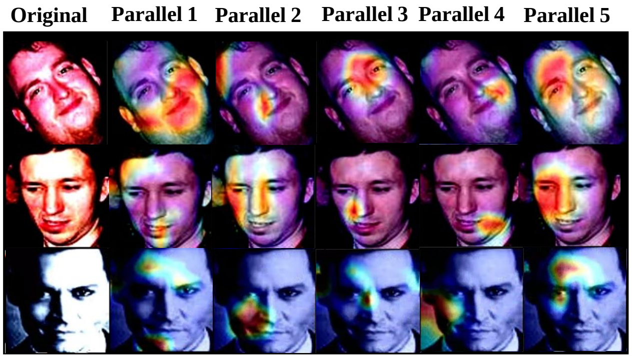


**Fig. 4.** Visualization of heatmaps from the outputs of parallel convolution branches.

## 4 Conclusion

In this work, we introduce a parallel convolution to augment the feature representation ability for in-the-wild expressions with complex semantics, and an additional regularization loss to let each branch independently respond to a semantic. Based on multiple combinations of the outputs from the parallel convolution, a self attention is followed to encode the correlations among multiple layers. Experimental results on cross-database FER show that our algorithm can better capture the complex correlations among multiple layers, and largely outperforms the state of the arts in terms of the cross-domain generalization performance. In our future work, we will give insight into the working mechanism of the parallel convolution for the generalization ability improvement. Other paradigms in addition to ViT will be investigated to test the generality of the proposed parallel convolution and the specific regularization loss.

**Acknowledgements.** The work was supported by the Science and Technology Project of Guangdong Province under grant no. 2020A1515010707, Natural Science Foundation of China under grants no. 62276170, 91959108, the Science and Technology Innovation Commission of Shenzhen under grant no. JCYJ20190808165203670.

## References

1. Yosinski, Jason, et al. "How transferable are features in deep neural networks?."Advances in neural information processing systems 27 (2014).
2. Chen, Tianshui, et al. "Cross-domain facial expression recognition: A unified evaluation benchmark and adversarial graph learning." IEEE transactions on pattern analysis and machine intelligence (2021).
3. Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).
4. Dhall, Abhinav, et al. "Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark." 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops). IEEE, 2011.
5. Goodfellow, Ian J., et al. "Challenges in representation learning: A report on three machine learning contests." International conference on neural information processing. Springer, Berlin, Heidelberg, 2013.
6. Zhang, Zhanpeng, et al. "From facial expression recognition to interpersonal relation prediction." International Journal of Computer Vision 126.5 (2018): 550-569.
7. Li, Shan, and Weihong Deng. "Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition." IEEE Transactions on Image Processing 28.1 (2018): 356-370.
8. Yan, Keyu, et al. "Cross-database facial expression recognition via unsupervised domain adaptive dictionary learning." International Conference on Neural Information Processing. Springer, Cham, 2016.
9. Zheng, Wenming, et al. "Cross-domain color facial expression recognition using transductive transfer subspace learning." IEEE transactions on Affective Computing 9.1 (2016): 21-37.
10. Ganin, Yaroslav, and Victor Lempitsky. "Unsupervised domain adaptation by backpropagation." International conference on machine learning. PMLR, 2015.
11. Piratla, Vihari, Praneeth Netrapalli, and Sunita Sarawagi. "Efficient domain generalization via common-specific low-rank decomposition." International Conference on Machine Learning. PMLR, 2020.
12. Glorot, Xavier, and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks." Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2010.
13. Guo, Yandong, et al. "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition." European conference on computer vision. Springer, Cham, 2016.
14. Ji, Yanli, et al. "Cross-domain facial expression recognition via an intra-category common feature and inter-category distinction feature fusion network." Neurocomputing 333 (2019): 231-239.
15. Li, Shan, Weihong Deng, and JunPing Du. "Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
16. Zavarez, Marcus Vinicius, Rodrigo F. Berriel, and Thiago Oliveira-Santos. "Cross-database facial expression recognition based on fine-tuned deep convolutional network." 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). IEEE, 2017.
17. Xu, Ruijia, et al. "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.