Cross-layer Contrastive Learning of Latent Semantics for Facial Expression Recognition

Weicheng Xie, Zhibin Peng, Linlin Shen*, Wenya Lu, Yang Zhang, Siyang Song

***Abstract*—Convolutional neural networks (CNNs) have achieved significant improvement for the task of facial expression recognition. However, current training still suffers from the inconsistent learning intensities among different layers, i.e., the feature representations in the shallow layers are not sufficiently learned compared with those in deep layers. To this end, this work proposes a contrastive learning framework to align the feature semantics of shallow and deep layers, followed by an attention module for representing the multi-scale features in the weight-adaptive manner. The proposed algorithm has three main merits. First, the learning intensity, defined as the magnitude of the backpropagation gradient, of the features on the shallow layer is enhanced by cross-layer contrastive learning. Second, the latent semantics in the shallow-layer and deep-layer features are explored and aligned in the contrastive learning, and thus the fine-grained characteristics of expressions can be taken into account for the feature representation learning. Third, by integrating the multi-scale features from multiple layers with an attention module, our algorithm achieved the state-of-the-art performances, i.e. 92.21%, 89.50%, 62.82%, on three in-the-wild expression databases, i.e. RAF-DB, FERPlus, SFEW, and the second best performance, i.e. 65.29% on AffectNet dataset. Our codes will be made publicly available.**

***Index Terms*—Facial expression recognition; Contrastive learning; Latent semantic alignment; Multi-layer attention**

I. Introduction

FACIAL expression is one of the most intuitive, natural and common non-verbal signals for humans to express their internal emotional state and intention, whose recognition has been widely used in various applications, such as human-computer interaction, driver fatigue detection, and medical treatment assessment. Due to powerful feature representation learning abilities, deep neural networks have been extensively studied in facial expression recognition (FER), which can well encode the muscle movements and subtle wrinkle textures from different facial displays.

Current DNNs frequently use the features produced from the latter layers as expression embedding. Actually, the middle or shallow layers of a network often imply expression-related local information, which is not well explored in the feature representation. Zeiler and Fergus [1] showed that the feature maps on different layers represent diverse information cues, i.e., while shallow layers represent the geometry features, deep layers encode complicated semantic features. However, the features on shallow layers are not explored much in the feature representation, since they are not learned with sufficient intensity compared with those on deep layers, i.e., shallow layers frequently suffer from the gradient vanishing problem during the training of networks with large numbers of layers [2], [3]. As shown in Fig. 1(a), the gradients of the loss with respect to (w.r.t.) the feature maps are attenuated as they are back-propagated to the shallow layers. This suggests

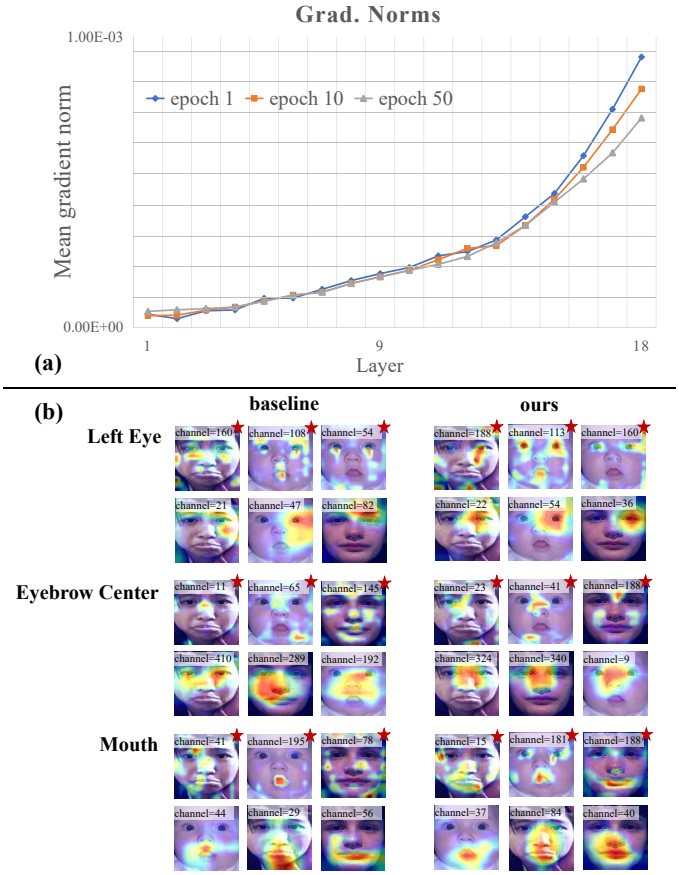


Fig. 1: (a) The norms of the back-propagated gradient for the features on shallow and deep layers. (b) The feature maps specific to different semantics by the baseline and ours, where the red stars label the feature maps output from shallow layers, while others are output from a deep layer, i.e. the last convolution block. It also shows the difference between the baseline and ours.

W. Xie, Z. Peng, L. Shen, W. Lu and Y. Zhang are with Computer Vision Institute, School of Computer Science & Software Engineering, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen, 518060, China. S. Song is with Department of Computer Science and Technology, University of Cambridge. W. Xie was also an academic visitor in School of Computer Science, University of Nottingham, Nottingham, UK. Corresponding author: Prof. Linlin Shen, Tel: 86-0755-86935089, Fax: 86-0755-26534078, llshen@szu.edu.cn

that the shallow layers are not sufficiently learned and there is a large inconsistency between the learning intensities of the shallow-layer and deep-layer features. This inconsistency may largely deteriorate the representation performance when the shallow-layer features are integrated with deep-layer features for object recognition tasks.

To alleviate this inconsistency, Yao et al. [4] proposed the knowledge distillation between teacher and student networks. Yu et al. [5] resorted to sharing the parameters of the shallow layers on the global and local tasks. Rather than using cross-network or cross-task learning, Mostafa et al. [6] devised an auxiliary supervision classifier, Sun et al. [7] proposed to learn the knowledge using an auxiliary supervision branch, Garg et al. [8] used the contrast of latent features from adjacent two layers to regularize this consistency. Instead of learning from several layers, Huang et al. [9] proposed to couple each shallow layer with all the deep layers to strengthen feature propagation. These cross-layer learning methods can improve the learning intensity at shallow layers. However, they are based on the global feature representation, which does not consider the fine-grained semantics implied in facial expressions.

Latent semantics, i.e. the muscle deformation of facial organs associated with expressions, or a fixed pattern of texture variation in expressions, are revealed to be common and useful for expression feature representation. Yang et al. [10] showed that each action unit (AU) of expressions consists of a specific semantic description, and Ruan et al. [11] revealed that facial action-aware latent features can well characterize expression similarities and variations. Especially, Zhang et al. [12] shows that semantic learning can improve network robustness against the face occlusion and poses that are ubiquitous for the in-the-wild expressions [13]. Thus, we explore the hidden semantics on the deep layer, and enable shallow layers to learn these semantic cues based on cross-layer feature alignment in a semantic-wise manner. For the semantic exploration and cross-layer semantic alignment, we take advantage of the contrastive learning paradigm that can learn object semantics by exploring image prior knowledge in a self-supervision manner [14], and its enhanced variants with feature clustering [15], [16].

Since features from shallow and deep layers present different scales of cues that are complementary, integration of them can theoretically produce multi-scale and strong representations for FER [17], [18], [19]. Specifically, it assembles the global and local features [20], i.e., encoding both deep semantics as well as shallow geometry features [21]. Consequently, such multi-scale features are more discriminative and less sensitive to the face occlusion and poses for facial expressions recorded in-the-wild. For the integration of shallow-layer and deep-layer features, the attention mechanism is widely employed to strengthen the responses of key features [22], [23], [8], [13], [24], due to its weight adaption with network back-propagation. In this work, we use the attention mechanism to weigh the contributions of the deep-layer and enhanced shallow-layer features to produce the multi-scale feature representation.

However, existing multi-scale models [17], [21], [25], [26] learned from the features in the shallow layers as a whole. ADDL [27] and MANet [21] utilized network multi-layer outputs to fuse multi-scale feature representation, while they did not well address the mismatched learning intensities of features in shallow and deep layers, which may limit the performance of the fused feature representation. Consequently, we propose a contrastive learning framework to align the latent semantics of the expression features from different layers. In this way, the learning intensities of the shallow layers are enhanced with the supervision of the deep layer. As shown in 1 (b), the feature maps by our approach are activated more on the expression-perceptive regions, compared with those by the baseline. Meanwhile, by aligning the cross-layer features in a semantic-wise manner, the proposed algorithm can sufficiently use the latent semantics that are robust against the face poses and occlusions for feature representation learning. Furthermore, the enhanced features on the shallow layers are integrated with those on the deep layer based on an attention module, to leverage their complementarity.

In summary, our main contributions are shown as follows

- We propose a simple yet effective cross-layer contrastive learning paradigm to enhance the learning intensity of features on shallow layers of CNNs.
- To take into account the characteristics of latent semantics of facial expressions, we proposed two modules to align the cross-layer features in a semantic-wise manner.
- An attention module is introduced to produce multi-scale feature representation by integrating the enhanced shallow-layer and deep-layer features. Extensive experimental results show the effectiveness of the proposed algorithm for the recognition of in-the-wild expressions.

The rest of this article is organized as follows. Section II reviews the related works. Section III describes the proposed contrastive learning and multi-scale representation modules. The experimental results and analysis are presented in Section IV. Finally, Section V presents the conclusions and discussions.

II. Related Works

A. Facial Expression Recognition

The task of recognizing expressions in the wild is challenging, since it may suffer from the problem of multi-scale geometry deformation of key expression parts, as well as face occlusions and poses.

For the representation of multi-scale features, multi-layer feature maps can represent multi-scale features [17], i.e. the spatial perception information on shallow layers and high-level semantic information on deep layers. Fan et al. [26] embedded the attention module in multi-level layers to capture rich feature representations. Ruan et al. [27] observed that the features from different layers are complementary, and thus proposed a multi-layer attention mechanism to fully exploit these cues. Zhu et al. [28] proposed cross-layer attention and center-guided attention, to leverage the features from multi-level granularity in a unified way. These works motivate us to design the multi-scale feature representation of multiple-layer outputs with the aid of an attention module.

For alleviating the influence of face poses, illumination, and occlusion [29], outlier sample suppression [30] [31], attention networks [32], [33], adversarial feature learning [34], alignment of the distributions of the occluded and non-occluded features [35], [36] were proposed. FLEPNet [37] utilized modified homomorphic filtering to normalize the illumination, which minimized the intra-class difference. FER-net [38] combined low-level texture features and high-level features to learn realistic edge variations. Siqueira et al. [39] revealed that shallow layers learn simple and local visual patterns such as oriented lines, edges, and colors, which appear more robust to face poses and occlusions than those learned on deep layers [40].

Despite shallow and local visual patterns are robust to occlusions and poses [39] and indispensable for FER, the features on shallow layers suffer from the insufficient learning intensity compared with those on deep layers, due to the gradient vanishing problem during network back-propagation. These inconsistent learning intensities hinder the effective exploitation of robust shallow-layer features for representing posed and occluded expressions.

B. Cross-Layer Feature Representation

To enhance the learning intensity for the shallow layers, Yao et al. [4] proposed knowledge distillation between teacher and student networks, to alleviate semantic gaps of the knowledge learnt at different-staged layers. Chen et al. [41] introduced feature-map transfer by semantic calibration via soft layer association, while they mainly targeted at the cross-layer learning between the teacher and student networks.

Rather than learning from multiple networks, the contrast of latent features from multiple layers on the same network can also enhance the learning intensities of shallow-layer features [8]. Yu et al. [42] introduced a guidance term to constrain the lower-level flow vector to be similar to the corresponding higher-level counterpart. By competing for a common resource, i.e. the shared layers, multiple layers can be mutually learned [39]. Yu et al. [43] proposed the technique of cross-layer bilinear pooling that simultaneously established the inter-layer interaction of features.

However, these methods align cross-layer features in a global representation, which may neglect the characteristic of the fine-grained semantics [44]. Actually, latent information, e.g. content, statistical or structural information [45] that are implicitly encoded in network outputs, may appear more robust against complicated circumstances than the global representation [12], [14]. Thus, we take the latent semantics into account during the cross-layer feature representation learning, with the aid of the self-supervision paradigm of contrastive learning [46], for better recognition of fine-grained expressions.

C. Expression Semantic Learning

Expression semantics consist of diverse muscle deformations and texture characteristics, which contribute differently for FER. Yang et al. [10] revealed that each action unit (AU) depicts a specific semantic related to facial expressions, each semantic may represent large variations of expression features [47], and its relationship is useful for guiding the representation learning [48]. For FER, semantics were frequently explored and modeled via, e.g. temporary auxiliary branches [49], separate residual blocks [50], visual semantic tokens [51] or semantic facial graph encoding AU occurrence [52], for the feature representation learning.

Especially, Zhang et al. [12] show that the learning on semantics can make the network more robust against the face occlusion and poses available for in-the-wild expressions [13]. Li et al. [53] also revealed that semantically separable features can well narrow the domain shift for cross-domain FER. Ruan et al. [11] showed that facial action-aware latent semantics can well characterize expression similarities and variations, and this high-level semantic information in the expression face is limited, which implies the advantage of the semantic-wise learning for FER. Meanwhile, Fu et al. [54] stated that semantics can be represented as the actions of key facial parts, such as the raising of lips or eyebrows, whose perturbation augmentation can well improve the robustness of FER.

Thus, these works implied that semantics can represent relatively fine-grained information that is less sensitive to posed and occluded faces, compared with the global feature presentation. However, this robustness characteristic of semantics is rarely considered in the cross-layer feature learning of expression representation. This is because the above methods [11], [54] fail to address the learning intensity gap between the hidden semantics of shallow and deep layers. Thus, we explore the latent semantics on deep layers and use them to guide the learning of latent features on shallow layers in a semantic-wise manner.

III. Proposed Method

In this section, the overall architecture of the proposed algorithm, together with the proposed cross-layer contrastive learning module, and the multi-layer feature representation module, are presented. Compared with existing methods, our method can well enhance the learning intensity of shallow-layer features, with an easy-to-implement self-supervision by deeper layers. Meanwhile, the cross-layer alignment in a semantic-wise manner can take advantage of the robustness of fine-grained semantics against posed and occluded expressions. These two main merits make our method particularly applicable to in-the-wild expression datasets.

A. Overall Architecture

The proposed algorithm includes four parts, i.e. a feature encoder module for original embedding representation, two contrastive learning modules, i.e. Contrastive Learning based on Feature Map (CLFM) and Contrastive Learning based on Batch Sample (CLBS), and the representation module of multi-scale features, i.e. Multi-scale Representation (MSR). An overview of the proposed method is shown in Fig. 2.

Without loss of generality, we assume that the feature encoder consists of L network blocks, and each block includes several convolution layers. Given a mini-batch of samples with n_b images, i.e. $\{x_i\}_{i=1}^{n_b}$, we use this encoder to extract features

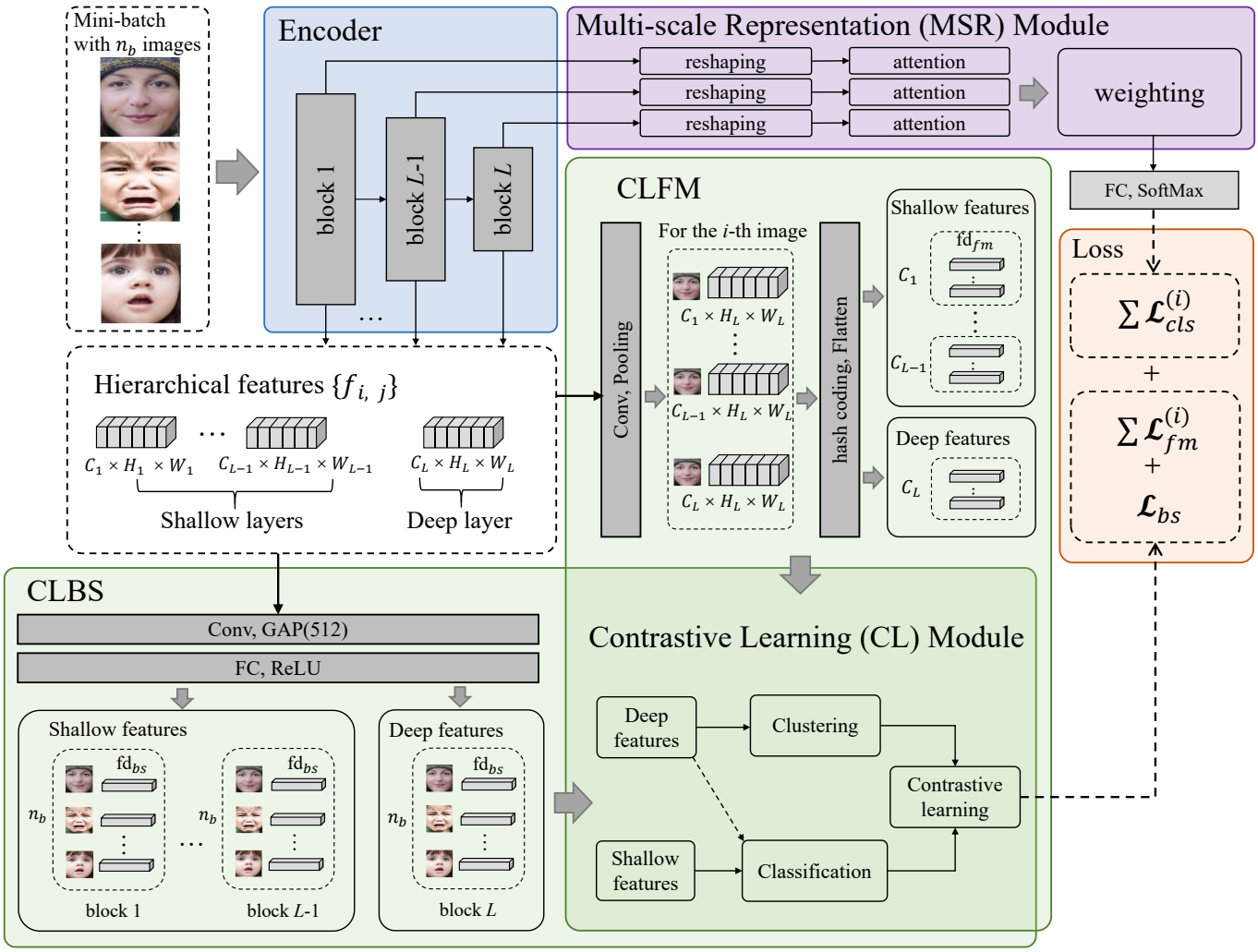


Fig. 2: Framework of the proposed algorithm. Based on the features represented with an encoder, Contrastive Learning (CL) and Multi-scale Representation (MSR) models are performed to learn cross-layer semantic features in a self-supervision paradigm and weigh the multi-layer features for a multi-scale representation. GAP(512) denotes the global average pooling with 512-dim output features. The CL and MSR modules are illustrated in Figs. 3 and 4.

from different blocks, and $f_{i,j}$ denotes the feature specific to the sample x_i from the j-th block ($1 \leq j \leq L$).

The encoded features $\{f_{i,j}\}$ are then used in the Contrastive Learning (CL) and MSR modules. CLFM and CLBS are proposed to enhance the learning intensity of the shallow layers and align the semantics across different layers, where losses in the CL module are introduced to reduce the difference between features of the same semantic. Meanwhile, MSR is proposed to fuse the multi-scale features from multiple layers for the feature representation based on the attention mechanism.

B. Contrastive Learning (CL) Module

To enhance the learning intensity of the features on the shallow layers, as well as align the latent semantics across different layers in a semantic-wise manner, the contrastive learning modules, i.e. CLFM and CLBS are proposed, and illustrated in Fig. 2.

*1) **Contrastive Learning based on Feature Map (CLFM) Module**:* As shown in Fig. 3, CLFM consists of three stages. The first stage aims to find clusters and the specific centroids of similar high-level semantic information from the deep layer outputs; The second stage assigns the features extracted from the preceding blocks with a semantic category according to the similarities between these features with the centroids; In the third stage, cross-layer features learn from each other based on the contrastive learning loss.

In the first stage, let's denote the feature corresponding to the output of the j-th layer as $f_{i,j}$, which is extracted from the i-th sample. The latent feature $g_{i,j}$ is generated as follows

$$g_{i,j} = Flatten(HashCoding(AP(Conv(f_{i,j})))) \quad (1)$$

where $Conv(\cdot)$ and $AP(\cdot)$ denote the convolution operator and average pooling. To speed up the clustering process, hash coding [55], i.e. the encoding with the coordinates of the maximum response area of each feature map is employed to reduce the dimension of the features to fd_{fm}. A flattening

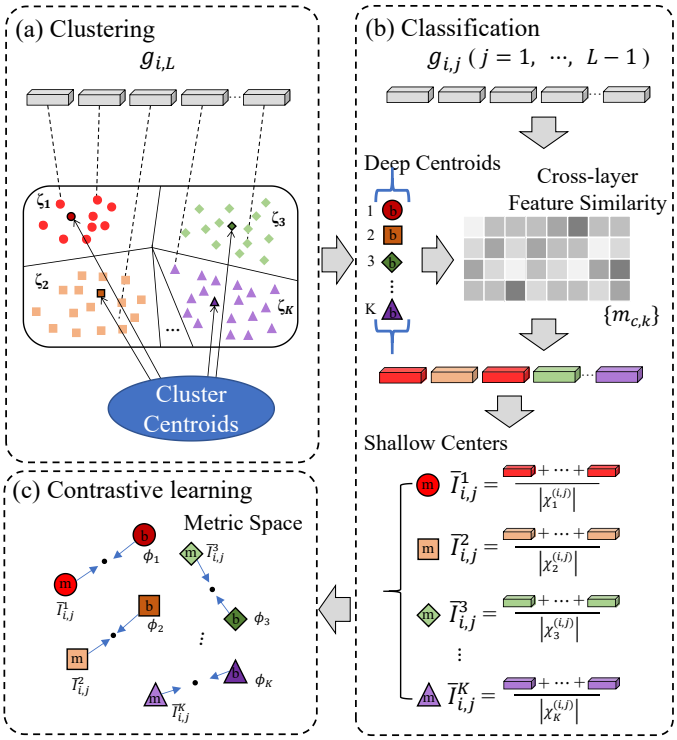


Fig. 3: The contrastive learning between the deep semantic centroids and the center features of shallow layers in the metric space.

operator is then followed to transform the feature map into a vector representation. Thus, the size of $g_{i,j}$ is $C_j \times \mathrm{fd}_{fm}$, where C_j represents the channel number of the j-th layer.

We argue that there are high-level semantics hidden in the feature maps of deep layers, and they are shared across different expression samples. Thus, we use the K-means clustering on the features from the last block, i.e. $g_{i,L}$, to construct K clusters, i.e. $\zeta_1, \cdots, \zeta_K$, to simulate the expression semantics. The corresponding cluster centroids, i.e. $\{\phi_k, 1 \le k \le K\}$, namely as 'Deep Centroids', are then found with the following minimization

$$\begin{cases} \min \mathcal{E} = \sum_{k=1}^{K} \sum_{g^{(c)}_{i,L} \in \zeta_k} \| g^{(c)}_{i,L} - \phi_k \|_2^2 \\ \phi_k = \frac{1}{|\zeta_k|} \sum_{g^{(c)}_{i,L} \in \zeta_k} g^{(c)}_{i,L} \end{cases} \tag{2}$$

where $g^{(c)}_{i,L}$ represents the c-th channel of $g_{i,L}$, ϕ_k is the feature centroid of the cluster ζ_k. Each ϕ_k represents a kind of high-level latent semantic, which indicates the sensory information related to expressions in CLFM, while in CLBS it indicates the unique information specific to each expression category. Meanwhile, the number of latent semantics of facial expression is limited, which is beneficial for us to fine tune an approximate value.

In the second stage, as shown in Fig. 3, each of the shallow-layer features on the preceding blocks, i.e. $g_{i,j}, j \in \{1, \cdots, L-1\}$, are classified according to the cosine similarity between these features and the deep centroids, i.e. $\{\phi_k\}$, as:

$$m^{i,j}_{c,k} = cos\langle\, g^{(c)}_{i,j},\, \phi_k \,\rangle \tag{3}$$

where $g^{(c)}_{i,j}$ represents the c-th channel of $g_{i,j}$. The high similarity correlation indicates that the shallow-layer feature implies similar semantic cues as that of the deep-layer feature.

Based on the similarity of the shallow-layer and deep-layer features, i.e. $\{m^{i,j}_{c,k}\}$, the features on the shallow layers can be categorized into K subsets, then the specific center features, namely as 'shallow centers', are obtained for the following cross-layer contrastive learning. We use $\chi^{(i,j)}_1, \cdots, \chi^{(i,j)}_K$ to denote the K groups for the j-th shallow layer of the i-th sample:

$$g^{(c)}_{i,j} \in \chi^{(i,j)}_k, \text{with } k = arg \max_{p \in \{1,\cdots,K\}} m^{i,j}_{c,p} \tag{4}$$

Then the mean feature specific to the k-th semantic category on the j-th shallow layer is obtained as follows

$$\bar{I}^k_{i,j} = \frac{1}{|\ \chi^{(i,j)}_k\ |} \sum_{g^{(c)}_{i,j} \in \chi^{(i,j)}_k} g^{(c)}_{i,j} \tag{5}$$

where $|\cdot|$ denotes the cardinality of the set.

In the third stage, it is desirable that the deep centroids of the L-th layer features and the shallow mean features of the j-th layer ($1 \le j \le L-1$) are pulled as close as possible. Specifically, a contrastive loss, i.e. $\mathcal{L}^{(i)}_{fm}$, is minimized to align the semantic representations on different layers. In this way, each layer of the network can better represent relevant semantic information, while shallow layers can learn the informative cues with enhanced intensity. The loss $\mathcal{L}^{(i)}_{fm}$ is formulated as follows

$$\begin{cases} \mathcal{L}^{(i)}_{fm} = -\sum_{j=1}^{L-1} \alpha_j\, \mathcal{L}^{(i;j,L)}_{fm} \\ \mathcal{L}^{(i;j,L)}_{fm} = \frac{vec(\bar{I}_{i,j})\ vec(\phi)}{||vec(\bar{I}_{i,j})||_2 \cdot ||vec(\phi)||_2} \end{cases} \tag{6}$$

where $\bar{I}_{i,j} = [\bar{I}^1_{i,j}, \cdots, \bar{I}^K_{i,j}] \in R^{\mathrm{fd}_{fm} \times K}$, $\phi = [\phi_1, \cdots, \phi_K] \in \mathcal{R}^{\mathrm{fd}_{fm} \times K}$. $vec(\cdot)$ denotes vectorization and $||\cdot||_2$ denotes the L_2 norm, $\{\alpha_j, 1 \le j \le L-1)\}$ are the hyperparameters.

*2) **Contrastive Learning based on Batch Sample (CLBS) Module**:* While CLFM can align the latent semantics between the shallow and deep layers, it does not take into account the expression characteristic implied in the expression categories. Since the expression categories naturally imply a number of the semantic cues, we further introduce the CLBS module to supplement CLFM, so as to enhance the learning intensity of the shallow-layer features. Meanwhile, CLBS takes into account the correlation among samples in a batch for the contrastive learning, which is also supplementary to CLFM that uses the correlation cues of feature maps in each sample.

As shown in Fig. 2, CLBS first obtains the feature representation of each sample with the operators of 'Conv', 'GAP (512)', 'FC' and 'ReLU'. For the clustering of the features on the deep layer, the number of clusters is set as the number of expression categories, rather than the number of latent semantics in CLFM. Then, the similar classification of shallow

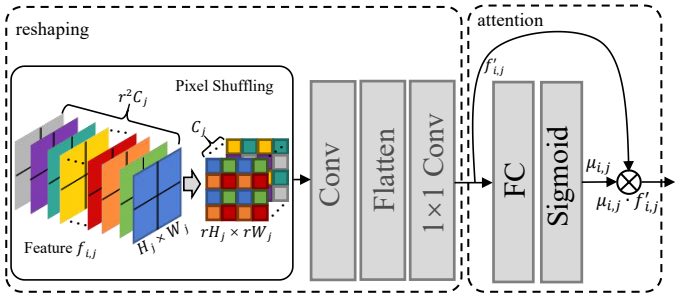


Fig. 4: The multi-scale representation module. The employed pixel shuffling is used to generate high-resolution feature maps by enabling associations between adjacent feature maps. The attention block is used to produce the weight $\mu_{i,j}$ specific to the j-th layer feature.

layer features in Eqs. (4) and (5), contrastive learning in Sec. III-B1 and Fig. 3 are performed, and the specific contrastive learning loss is formulated as follows

$$\mathcal{L}_{bs} = -\sum_{j=1}^{L-1} \beta_j \, \mathcal{L}_{bs}^{(j,L)} \tag{7}$$

where $\mathcal{L}_{bs}^{(j,L)}$ denotes the cosine similarity between the mean feature of the j-th shallow layer and the cluster centroid of the deep features on the L-th layer. β_j is the hyperparameter specific to the j-th shallow layer.

C. Multi-scale Representation (MSR) Module

While the features on deep layers are strongly relevant to a particular task, the features on the shallow layer are more general across different tasks. Meanwhile, the features on different layers are generated with different receptive fields. Thus, the multi-scale features from different layers appear complementary.

To make use of the multi-scale features from different layers, an attention module is proposed to weigh the contributions of these features. More precisely, the Pixel Shuffling [56] method is first used for the upsampling, to better utilize the distribution of original features, rather than the generated features through bilinear interpolation. Meanwhile, the coupled representation among adjacent feature maps can be produced and leveraged. Specifically, the feature maps with r^2C channels are transformed to be high-resolution features with periodic shuffling, where r is the upscaling factor for feature map magnification.

As shown in Fig. 4, based on the pixel shuffling (PS), the employed reshaping block on the feature representation of the j-th layer, i.e. $f_{i,j}$, is formulated as follows

$$\begin{cases} \mu_{i,j} = \sigma(W^T f'_{i,j}) \\ f'_{i,j} = Conv_{1\times1}(Flatten(AP(Conv(PS(f_{i,j}))))) \end{cases} \tag{8}$$

where $f'_{i,j}$ denotes a feature vector with the dimension of fd after reshaping. $Conv(\cdot)$, AP and $\sigma(\cdot)$ stand for convolution block, the average pooling and the sigmoid function. $Conv_{1\times1}$ denotes the 1×1 convolution for reducing the dimension of features and the runtime cost. W is the parameter matrix of the FC layer, and each vector can be viewed as a specific class prototype. $\mu_{i,j} \in \mathcal{R}$ is the output importance weight of the feature specific to the j-th layer.

Based on $\{\mu_{i,j}\}$ in Eq. (8), the fusion of multi-scale features from multiple layers specific to the i-th image, i.e. F_i, is obtained with an attention mechanism:

$$F_i = \sum_{j=1}^{L} \mu_{i,j} \cdot f'_{i,j} \tag{9}$$

In this way, the MSR module integrates multi-scale features from different layers, which can adaptively select the specific scale of features corresponding to the variation characteristic of expression categories. Especially, since the latent semantics on the shallow layers are enhanced, i.e. rich shallow-layer semantic cues similar to those of the deep layer can be also learned, yielding an intensity-homogeneous representation of multi-scale semantic features.

D. Joint Loss Function

The joint loss of the proposed algorithm training for a batch of samples is formulated as follows

$$\mathcal{L} = \textstyle\sum_{i=1}^{n_b} \mathcal{L}_{cls}^{(i)} + \mathcal{L}_{fm}^{(i)} + \mathcal{L}_{bs} \tag{10}$$

where $\mathcal{L}_{fm}^{(i)}$ and $\mathcal{L}_{bs}$ are formulated in Eqs. (6) and (7). $\mathcal{L}_{cls}^{(i)}$ is the cross entropy loss for the i-th sample, which is formulated as follows

$$\mathcal{L}_{cls}^{(i)} = -\sum_{t=1}^{n_c} \mathbb{1}_{t=y_i} log(T(F_i, \theta)) \tag{11}$$

where F_i is the feature representation in Eq. (9), n_c and θ denote the number of expression categories and network parameters; y_i and $T(\cdot)$ denote the ground-truth label and the network prediction probability at the y_i-th dimension. For clarity, the pseudo code of the proposed algorithm is presented in Alg. 1.

IV. Experiments

In this section, we evaluate our algorithm on four public in-the-wild expression datasets. First, type of dataset used, evaluation measures, techniques used, pre-processing techniques, and the hyperparameter setting are clarified. Then algorithm analysis of learning intensity and training complexity, etc., ablation study, sensitivity analysis, feature map visualization, and the comparison with the state of the arts are conducted to evaluate the performance of each proposed module and the overall algorithm. We follow the state of the arts to choose the same evaluation metrics, i.e. evaluating the performance of the models both quantitatively and qualitatively.

A. Databases and Implementation Details

The in-the-wild expression dataset includes more noisy, largely posed and occluded faces than the dataset collected in the lab, which is closer to real-world circumstances, and more challenging for recognition algorithms. The details of the used

Algorithm 1: Training of the proposed algorithm

Input: Training dataset $\{$ n_m Mini-batch $\}$, Mini batch $\{(x_i, y_i), 1 \leq i \leq n_b\}$, model parameters θ and iteration epoch τ.
Output: Optimized model.

```
Initialize θ with a pre-trained model;
while epoch < τ do
    Divide the training dataset into n_m mini-batches randomly
    for iteration ← 1 to n_m do
        Obtain deep semantic centroid {φ_k} by Eq. (2)
        Calculate the similarity between shallow-layer features and deep centroids by Eq. (3)
        Classify shallow-layer features by Eq. (4)
        Obtain the shallow centers of the classified features by Eq. (5)
        Compute the loss L_fm^(i) of CLFM by Eq. (6)
        Compute the loss L_bs of CLBS by Eq. (7)
        Derive multi-scale feature representation by Eq. (9)
        Compute the joint loss L by Eq. (10)
        Update θ with SGD
    end
end
```

datasets are presented below and the detailed distributions of the datasets are shown in the supplementary material.

RAF-DB [57] is a large-scale facial expression dataset containing 30,000 images, with basic or compound expressions labeled by 40 trained volunteers. There are six basic expressions, i.e. happy, surprise, sad, angry, disgust and fear, as well as neutral in the dataset. In our experiment, we use 12,271 images for training and 3,068 images for the testing. Our evaluation protocol is consistent with that in FDRL [11], RAN [33], SCN [13], ADDL [27], etc.

FERPlus [58] is extended from FER2013 [59], where the images have been re-labeled into one of 8 emotion types, i.e. neutral, happy, surprise, sad, angry, disgust, fear, and contempt. While FER2013 consists of 35,887 facial expression images, including 28,709 training images, 3,589 validation images, and 3,589 test images, with a size of 48×48, we then strictly use the code[1] officially provided by FERPlus to re-label and reduce the samples in FER2013. We report the overall accuracy on its testing set, which is consistent with that in SCN [13], RAN [33], ADDL [27], etc.

SFEW [60] is created by selecting keyframes from AFEW [61], which contains 958 training images, 436 validation images, and 372 testing images, showing one of six basic and neutral expressions. This in-the-wild database also includes posed faces, multiple faces in a scene, occlusions, and different lighting conditions. The performance is reported on its validation set, which is consistent with that in SCN [13], ADDL [27], MA-Net [21], etc.

AffectNet [62] is by far the largest database of facial expressions in the wild. It contains 450,000 facial images from the Internet with both categorical and valence-arousal annotations. For FER, 7 or 8 expressions are often employed. In this experiment, we use the seven categories, i.e. the six basic and neutral expressions, 28,3901 images for training and 3,500 images for testing. The evaluation protocol is set as the same as that in DACL [24], IPA2LT [63], EfficientFace [64], ADDL [27] and FDRL [11].

Implementation details. For pre-processing, each image is resized to 256×256, which is further randomly cropped to the size of 224×224, erased and horizontally flipped for the data augmentation. Our method is implemented with the backbone of ResNet-18 [2] based on Pytorch. Correspondingly, ResNet-18 has 4 basic blocks, then the number of blocks (L) in Fig. 2 is set to 4. And the r value in Fig. 4 is set as 2^{j-1} for the feature maps of the j-th block. The feature dimensions of fd_{fm} and fd_{bs} in Fig. 2, and fd in Eq. (8), are set as 16, 128 and 128, respectively. The detailed architectures of the backbone and its modules are shown in the supplementary material.

The network is trained for 100 epochs with a single P100 GPU based on the Adam algorithm [65], which is pre-trained on ImageNet [66]. The initial learning rate, the weight decay, and the base gamma are set as 0.001, 0.0001, and 0.9, respectively. An exponential decay strategy is employed to adjust the learning rate.

B. Algorithm Analysis

Learning intensity analysis. To study the learning intensity of features on shallow layers, we compare the average gradient norms by the baseline and our method, together with the average gradient norms of different semantic groups in Fig. 5.

Fig. 5(a) shows that the proposed algorithm can effectively enhance the learning intensity of the features on shallow layers, compared with the baseline, i.e. the gradient norms specific to the shallow layers after the training with our algorithm are enlarged. As shown in Fig. 5(b), the gradient norms of different semantic groups on the shallow layers show large differences, which means that the proposed semantic clustering is necessary to take the characteristic of each semantic into account during the feature representation learning.

Training complexity analysis. To study the runtime cost of the proposed modules in addition to that of the original training, we conduct a comparison between the proposed algorithm and other methods in terms of the number of parameters (Params), Floating Point operations (FLOPs) in Table I. EfficientFace [64] uses Label Distribution Generator (LDG) in training, and its model size is 23.5M. Compared with ADDL [27], our model is lighter, i.e. 6.1M smaller than its 20.6M. Table I shows that the proposed modules do not bring too much additional burden over the baseline training in terms of Params and FLOPs.

More rigorously, we mainly introduce a clustering process and the similarity calculation between features in addition to the baseline training, while their runtime costs are only $O(\tau \cdot max(n_b, C) \cdot K \cdot max(\mathrm{fd}_{fm}, \mathrm{fd}_{bs}))$ and $O(max(n_b, C) \cdot K \cdot max(\mathrm{fd}_{fm}, \mathrm{fd}_{bs}))$, where τ, n_b, C, K and fd_{fm} are the

[1]The source code link is https://github.com/microsoft/FERPlus

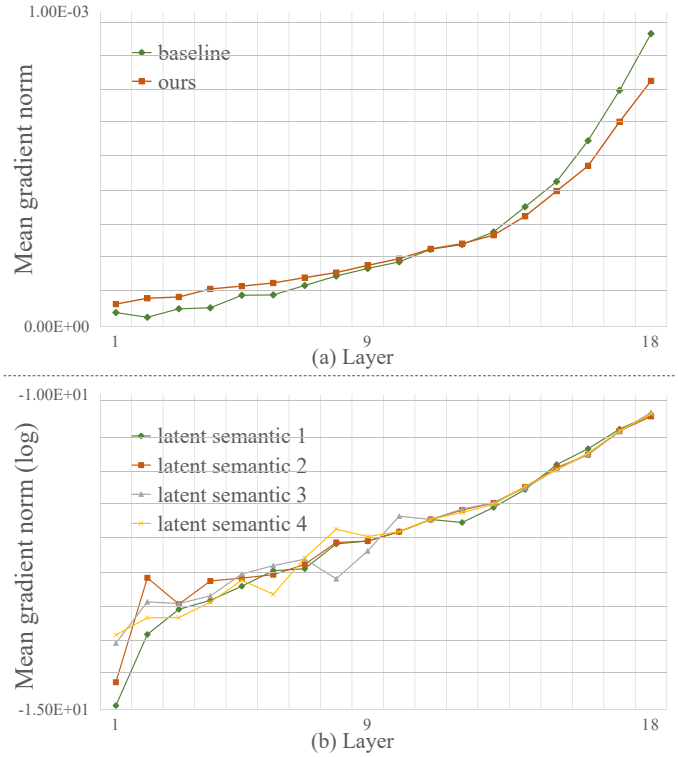


Fig. 5: The mean norms of the back-propagated gradients with training on RAF-DB. (a) Comparison between baseline and the proposed model. (b) Comparison of gradient norms specific to different latent semantics, where the norms are normalized with logarithmic function.

TABLE I: Comparison of model complexities between the other methods and ours.

Methods	Params	FLOPs
IPA2LT [63]	23.5M	4.11G
gACNN [32]	134.3M	15.48G
RAN [33]	11.2M	14.54G
SCN [13]	11.2M	1.82G
MA-Net [21]	50.5M	3.65G
ADDL [27]	20.6M	3.82G
EfficientFace [64]	1.3M+23.5M (LDG)	0.15G+4.11G (LDG)
Training the baseline	11.2M	1.82G
Training ours	14.5M	2.01G

numbers of iterations, samples, feature maps, clusters and feature dimension defined in Sec. III. These time complexities are linear to the numbers of feature maps and samples. Therefore, our proposed modules do not introduce much additional runtime cost over the complexity of the baseline training.

Confusion matrix. The confusion matrices of the performances for the considered four databases are presented in Fig. 6. It is shown that our method outperforms the baseline for most categories of expressions, and improves the accuracy of the 'Neutral' and 'Sad' categories on almost all the testing datasets. Specifically, our method improves the accuracy from 89% to 98% for 'Neutral', and from 81% to 90% for 'Sad' on RAF-DB. Since there are minor deformations or texture variations in the key expression regions, e.g. 'mouth', 'eyes' and 'eyebrows', the specific semantic cues of these two expressions are relatively fine-grained. To this end, our algorithm conducts cross-layer representation learning in a semantic-wise manner, which can better represent these fine-grained cues than the global feature representation, thereby enhancing the discriminative capacities for these expressions.

TABLE II: The performances of baseline, Simsiam [67] and ours on RAF-DB and AffectNet when different numbers of labeled samples are used in the training. The notation '20% supervised' represents using the cross-entropy loss in the supervised learning with 20% of the training samples. The notation 'unsupervised*' represents using our proposed unsupervised cross-layer contrastive loss, while 'unsupervised' denotes using the contrastive loss of Simsiam.

Methods	RAF-DB	AffectNet
Baseline (Only 20% supervised)	78.52	53.09
Ours (Only 20% supervised)	**82.86**	**57.43**
Simsiam (20% supervised + 80% unsupervised)	83.25	59.86
Ours (20% supervised + 80% unsupervised*)	**86.28**	**61.69**
Baseline (100% supervised)	85.78	58.20
Ours (100% supervised)	**92.21**	**65.29**

The performances of our contrastive learning under partial supervision. To study the performance of our constrastive learning when only a part of labels are used in model training, we use 20% of the labeled samples in the training set for the supervised cross-entropy loss, while the remaining 80% unlabeled samples in an unsupervised contrastive learning loss. ResNet-18 [2] is used as the encoder, and the contrastive learning framework of Simsiam [67] is used in the comparison, the results on RAF-DB and AffectNet are shown in Table II.

We conclude the following observations: (i) Table II shows that our proposed model achieved the best performances for both RAF-DB and AffectNet, in the case of 'Only 20% supervised'. For the reasons, our model limits the feature learning within a limited number of semantics, which enables the network to learn more discriminative features with a small number of samples. (ii) For the case of 20% samples for supervised learning and 80% samples for unsupervised learning, the proposed algorithm outperforms Simsiam [67] by the margin of 3.03% on RAF-DB, and 1.83% on AffectNet. For the reasons, in addition to the learning of features in deep layer employed in Simsiam [67], our unsupervised method can also well explore the features in the shallow layers to against the influences of face poses, occlusion, etc. [39]. (iii) Our model achieves the performances of 86.28% on RAF-DB and 61.69% on AffectNet under the setting of '20% supervised + 80% unsupervised*', which are already better than those using 100% labels in the baseline model, i.e. 85.78% and 58.20%.

Evaluation of model robustness against semantic variations. To investigate the robustness performance of our model against semantic variations, we propose to remove a large proportion of samples or all the samples of a category during the training. In this way, new semantics that are unseen for the trained network can be simulated, and the accuracy on them can thus reflect the capacity of a learned network generalizing to unseen semantics.

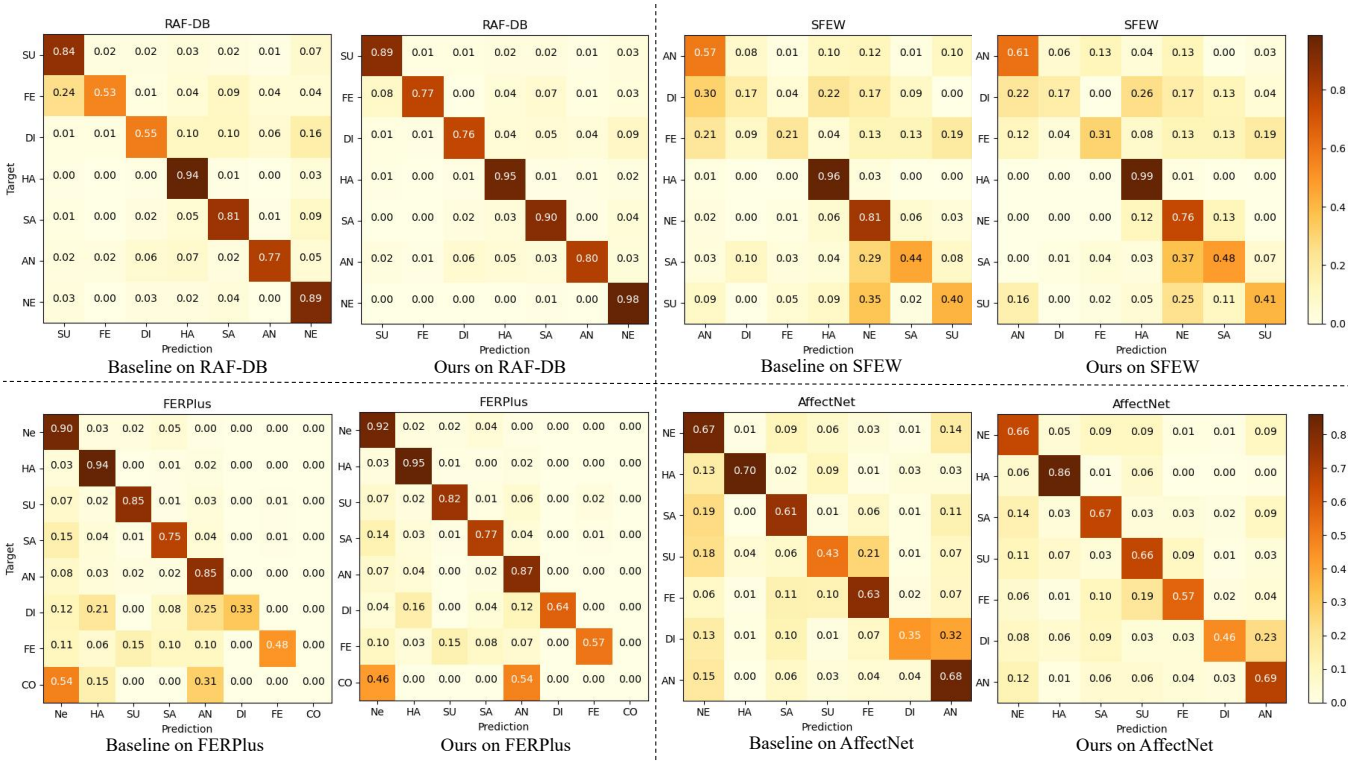


Fig. 6: The confusion matrices on the testing sets of RAF-DB, SFEW, FERPlus and AffectNet. 'SU', 'FE', 'DI', 'HA', 'SA', 'AN', and 'NE' represent 'Surprise', 'Fear', 'Disgust', 'Happy', 'Sad', 'Angry', and 'Neutral' expressions, respectively.

TABLE III: The recognition accuracy (%) of the baseline, SCN [13] and ours on RAF-DB when different numbers of samples in one category are retained for the training. '0% neutral' represents removing 'neutral' expression samples from the training, while the training samples of the other classes are retained. '10% neutral' represents removing 90% of 'neutral' training samples from RAF-DB during the training. The performance is evaluated on the testing set of RAF-DB.

Setting	Baseline	SCN [13]	Ours
0% neutral	77.90	79.07	**88.66**
0% happy	73.66	73.96	**88.40**
0% sad	76.08	78.19	**86.02**
0% surprise	78.49	80.12	**88.04**
0% angry	81.88	83.05	**89.41**
0% disgust	82.33	84.35	**90.94**
0% fear	82.07	86.47	**90.48**
10% neutral	84.45	85.01	**89.96**
10% happy	82.20	82.69	**88.59**
10% sad	81.32	81.88	**86.64**
10% surprise	83.12	84.32	**89.37**
10% angry	84.03	84.68	**89.86**
10% disgust	84.91	85.85	**89.24**
10% fear	83.70	85.56	**90.74**
30% neutral	83.34	84.62	**91.46**
30% happy	84.75	86.08	**89.08**
30% sad	83.41	84.49	**88.46**
30% surprise	85.01	86.64	**90.84**
30% angry	84.62	85.63	**91.04**
30% disgust	85.20	85.53	**90.91**
30% fear	85.07	86.83	**90.84**

For this experiment, we classify a testing sample to be the removed class if the maximum prediction probability (after Softmax normalization) is lower than a threshold of 0.25, i.e. the confidence degree of the network prediction is lower than this threshold. For the detailed explanation of this experiment, please refer to the supplementary material. The recognition accuracies (%) of the baseline, SCN [13] and ours on RAF-DB are shown in Table III, where different numbers of samples in one category are used for the training. Meanwhile, to shed light on the activation of the learned feature maps, we present the visualization of feature maps on shallow layers in Fig. 7.

We conclude the following observations: (i) Table III shows that our method consistently outperforms SCN [13] and baseline when a category of expression samples are removed during the training. (ii) Table III also shows that our model trained on largely-reduced data can achieve an accuracy, e.g. 91.46%, that is approximate to the performance using the entire training dataset, i.e. 92.21% on RAF-DB, which reveals the effectiveness of our model in combating against the overfitting of semantic variations. (iii) Compared with the baseline, Fig. 7 shows that our model can still focus on expression-related regions on images of the unseen (removed) category during training. It reflects that the features on shallow layers can also learn the expression-related semantics for this unseen category, are thus not overfitted to the semantic variations.

C. Ablation Study and Sensitivity Analysis

To show the performance of each proposed module, an ablation study is performed in Table IV. The performance

Fig. 7: Feature map visualization for the example testing samples of 'Happy', 'Surprise', 'Angry', and 'Sad' from RAF-DB, the models are learned by the baseline and ours, when the training samples with the same category as each testing sample, are removed during the training. Feature maps are selected from the 1st and 2nd blocks of the learned network.

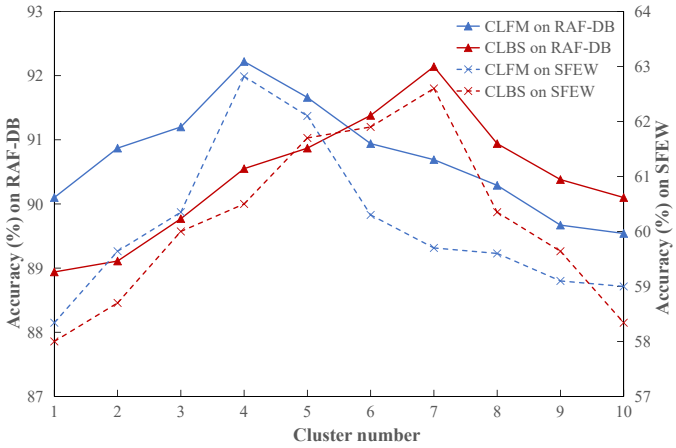


Fig. 8: The sensitivity testing of the accuracy against the number of clusters, i.e. K in Eq. (2).

TABLE IV: Ablation study with recognition accuracy (%) of the proposed modules on RAF-DB, AffectNet and FERPlus.

MSR	CL		RAF-DB	AffectNet	FERPlus
	CLFM	CLBS			
×	×	×	85.78	58.20	87.63
×	×	✓	90.38	62.66	88.58
×	✓	×	90.34	61.69	88.44
×	✓	✓	90.94	62.29	88.75
✓	×	×	87.40	59.28	87.94
✓	×	✓	91.88	62.97	89.22
✓	✓	×	91.69	63.66	89.08
✓	✓	✓	92.21	65.29	89.50

TABLE V: Accuracy (%) sensitivity of our algorithm against the settings of the hyperparameters $\{\alpha_j, \beta_j\}$ on RAF-DB.

α_1	α_2	α_3	β_1	β_2	β_3	Accuracy
0.01	0	0	0.01	0	0	91.34
0	0.01	0	0	0.01	0	91.88
0	0	0.01	0	0	0.01	91.10
0.01	0,01	0.01	0.01	0.01	0.01	91.56
0.1	0	0	0.1	0	0	91.10
0	0.1	0	0	0.1	0	92.21
0	0	0.1	0	0	0.1	90.38
0.1	0.1	0.1	0.1	0.1	0.1	90.94
0.5	0	0	0.5	0	0	89.39
0	0.5	0	0	0.5	0	90.34
0	0	0.5	0	0	0.5	89.56
0.5	0.5	0.5	0.5	0.5	0.5	89.77

TABLE VI: The performances of different strategies for obtaining the semantic centers on RAF-DB and AffectNet, where the backbone is ResNet-18.

Methods	RAF-DB	AffectNet
Baseline	85.78	58.20
$Center^{(t)}$	89.15	63.66
Ours (proposed)	**92.21**	**65.29**

sensitivity against the hyperparameters of $\{\alpha_j, \beta_j\}$ and the number of clusters are presented in Table V and Fig. 8, respectively.

Ablation study. The ablation study for each of the proposed module is conducted on RAF-DB, AffectNet and FERPlus and the results are shown in Table IV.

As shown in Table IV, MSR enables the learned network to achieve the improvements of 1.62%, 1.08% and 0.31% on RAF-DB, AffectNet and FERPlus over the baseline, CLFM improves the baseline by the margins of 4.56%, 3.49% and 0.81%, and CLBS achieves the improvements of 4.60%, 4.46% and 0.95%. Meanwhile, the integration of MSR, CLFM and CLBS achieves the best performances on all the three databases, which reveals the complementarity of these modules in feature representation learning.

Table IV shows that neither mere CLFM nor MSR could achieve a large improvement on the basis of CLBS, while the integration of MSR and CLFM always achieves large improvements. This is because the modules of CLFM and MSR are complementary, i.e. CLFM demands the attention information in MSR to well trade off the semantics in both shallow and deep layers, while MSR demands CLFM to well enhance the learning intensity of the shallow-layer semantics via semantic alignment of shallow and deep layers.

Table IV also shows some biases on different datasets. Specifically, it shows that CLFM helps CLBS to improve the accuracy from 88.58% to 88.75% for the FERPlus dataset, and 90.38% to 90.94% for RAFDB. However, this observation is different for AffectNet, which may be because AffectNet is more complex than RAF-DB and FERPlus, i.e. its training set has a more serious long-tail distribution problem and there is a larger distribution bias between its training and testing sets.

Evaluation of the hyperparameters $\{\alpha_j, \beta_j\}$. To study the layers used for the contrastive learning in CLFM and CLBS, we conduct the sensitivity analysis of the proposed algorithm

against $\{\alpha_j, \beta_j\}$ in Table V.

Table V shows that the cross-layer contrastive learning with the feature maps from the 2nd and 4th blocks performs the best, and our algorithm achieved the best performance with the setting of $\alpha_2 = \beta_2 = 0.1; \alpha_j = 0, j \neq 2; \beta_j = 0, j \neq 2$. Thus, the contrastive learning of the outputs from the 2nd and 4th blocks, together with this manual parameter setting, are employed in the proposed algorithm for the following evaluation and comparison.

Evaluation of the number of clusters K. To study the performance sensitivity against the hyperparameter of cluster number, i.e. K in Eq. (2), we evaluate the performances of CLFM and CLBS with different K values on RAF-DB and SFEW in Fig. 8. For convenience, we use K_{FM} and K_{BS} to represent the K value in CLFM and CLBS, respectively.

For CLFM, Fig. 8 shows that the proposed algorithm achieves the best performance with the setting of $K_{FM} = 4$. While too few clusters can not sufficiently represent the semantics of different expressions, too many clusters may result in representation redundancy, i.e. different clusters of feature maps may represent entangled semantics, making the learning ineffective. By formulating the high-level information to a few categories of semantics, the learning of expressions is semantic-wise disentangled, which enables us to reduce the features with less important semantics, such as hair, identity information, etc.

For CLBS, the setting of $K_{BS} = 7$ performs the best for the two databases. Based on this, we argue that each expression class naturally implies an independent category of latent semantic, the contrastive learning with the cluster number being that of the expression classes, can guide the network to explore this semantic for cross-layer feature representation learning.

Evaluation of the strategy of clustering-based centers. To evaluate the strategy of our clustering-based centers, we introduced another strategy to calculate the semantic centers for the comparison, i.e. directly using the class center of each class to calculate the contrastive loss, rather than the suggested clustering centers. First, we formulate the class center of each class $Center^{(t)}$ as:

$$Center^{(t)} = \frac{1}{n_t} \sum_{i=1}^{n_b} \mathbb{1}_{t=y_i} g_{i,L} \tag{12}$$

where n_t represents the number of samples of the t-th class in the mini-batch, and n_b is the number of mini-batch samples. $g_{(i,L)}$ obtained by Eq. (1) is the feature of the i-th sample in the L-th layer, and y_i is its label. Then we use Eq. (3) to calculate the cosine similarity between the class center and the sample feature representation, and Eq. (4) to assign each sample to the class whose center is the most similar to the sample feature. Finally, the average feature of each class obtained by (5) and the class center $Center^{(t)}$ obtained by Eq. (12) are used to calculate the contrastive learning loss by Eq. (6). The performances of this class-center strategy, together with the baseline and the proposed clustering-based strategy (ours) are shown in Table VI.

Table VI shows that the proposed strategy outperforms the class-center strategy on both datasets. For the reasons, we found that the features of some hard samples may be initialized to be far away from its class center at the beginning of training (Fig. 9(a)), this will impair the robustness of the following optimization. By contrast, the proposed clustering strategy can obtain more robust centers for contrastive learning in early iterations.

D. Visualization

To shed light on the working mechanism of the proposed algorithm, we compare the visualizations of the network outputs in Figs. 9, 10, 11, and 12 after training RAF-DB.

For CLBS, we visualize the 2D feature representations of the baseline and our method in Fig. 9, where the number of clusters is set as that of expression categories. Fig. 9 shows that the proposed algorithm can better separate different categories of expression samples compared with the baseline, i.e. the yielded feature representations by ours show better inter-class separation and intra-class compactness, which become more obvious when training for more epochs. Especially, our algorithm can better distinguish neutral expressions from others, compared with the baseline.

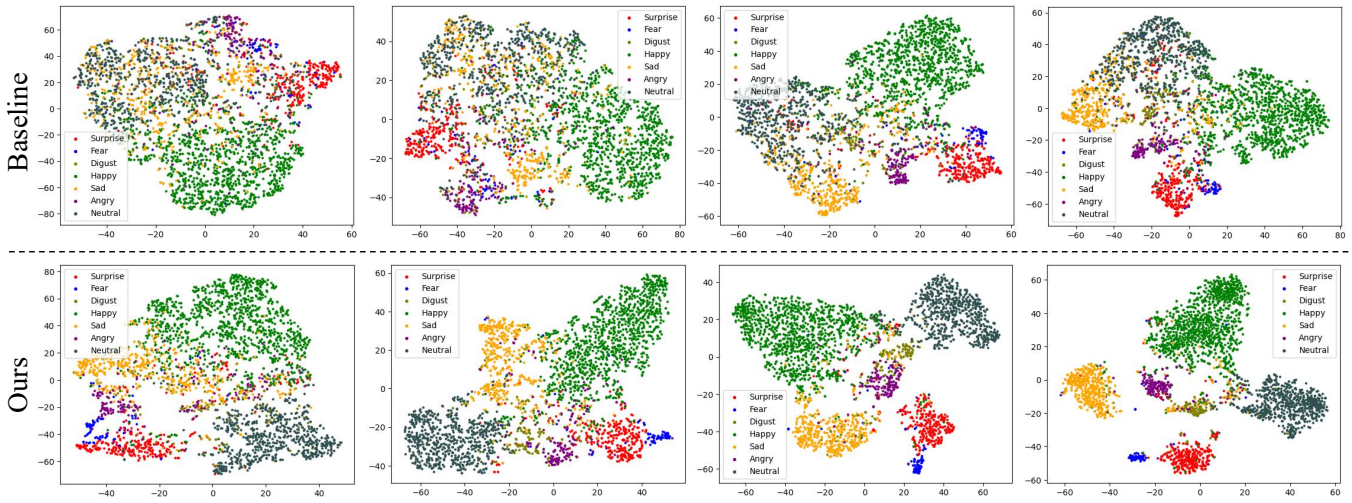
For CLFM, we visualize four example feature maps specific to each of four semantics in one row of Fig. 10, i.e. each row of feature maps are selected from the same semantic cluster of the outputs from the 1st block. Fig. 10 shows that different groups of feature maps correspond to largely diverse geometry structures and textures, i.e. each feature group may represent a kind of latent semantic, which reveals the rationality of the employed semantic-wise manner in cross-layer alignment.

To study the feature representations of different network blocks by the proposed modules, i.e. CLFM and CLBS (CL), and MSR, we show the feature maps output from these blocks, by training with the baseline and the variants of MSR and CL+MSR in Fig. 11. Fig. 11 shows that CL+MSR can locate broader salient regions than the baseline and the variant with MSR. That is, the responses on the feature maps, i.e. feature learning intensities of shallow layers can be well enhanced by our algorithm. The enhanced shallow-layer features enable the model to better utilize features that are robust to face pose and occlusion to improve expression recognition performance.

To study the convergence of our algorithm, we visualize the evolution of training and testing losses in Fig. 12. As shown in Fig. 12, for the training set, our model is not much different from the baseline, and the loss is slightly larger than the baseline in the preceding 15 epochs. For the testing set, the loss of our model is larger than that of the baseline in the preceding 5 epochs, while smaller than that of the baseline after the 5th epoch. These results show that our proposed model can alleviate the phenomenon of premature convergence.

E. Comparison with State-of-the-art Methods

To compare the proposed algorithm with other related algorithms, Table VII shows the performances of our algorithm and the state-of-the-art methods on RAF-DB, SFEW, FERPlus and AffectNet, where the employed network, pre-trained model, whether extra-data or oversampling is employed, optimizer, published year, as well as the baseline performances are also



(a) Epoch 0 (b) Epoch 5 (c) Epoch 10 (a) Epoch 15

Fig. 9: Feature visualization of the baseline (the 1st row) and our proposed method (the 2nd row) using t-SNE on the testing dataset of RAF-DB.

TABLE VII: Comparison of the accuracy (%) with state-of-the-art algorithms. The best and 2nd best performances are marked with bold and underline, respectively. 'Extra-data' indicates that the model was trained using extra data besides of the basic training set. Specifically, SCN [13] combines the RAF-DB [68] and AffectNet [62] for training. CPG [69] utilizes datasets such as Vggface2 [70] and LFW [71]. ADDL [27] utilizes Multi-PIE [72] and RAF-DB [57] to train the Disturbance Feature Extraction Model (DFEM).

Methods	Network	Pre-trained	Extra-data	Oversampling	Optimizer	Year	RAF-DB	SFEW	FERPlus	AffectNet
gACNN [32]	VGG-16	ImageNet	-	-	SGD	2018	85.07	-	-	58.78
DLP-CNN [68]	DCNN	-	-	-	-	2017	84.13	51.05	-	-
RMT-Net [73]	VGG-16	-	-	-	SGD	2021	87.27	-	-	61.98
IPA2LT [63]	ResNet-80	-	-	-	-	2018	86.77	58.29	-	57.31
PLD [58]	VGG-13	-	-	-	-	2016	-	-	85.10	-
RUL [74]	ResNet-18	MS_Celeb_1M	-	-	-	2021	88.98	-	88.75	-
FDRL [11]	ResNet-18	MS_Celeb_1M	-	-	Adam	2021	<u>89.47</u>	<u>62.16</u>	-	-
SPDNet [75]	DCNN	-	-	-	-	2018	87.00	58.14	-	-
MA-Net [21]	ResNet-18	MS_Celeb_1M	-	✓	SGD	2020	88.40	59.40	-	64.53
CPG [69]	ResNet-50	ImageNet	✓	-	SGD	2020	-	-	-	63.57
RAN [33]	ResNet-18	MS_Celeb_1M	-	✓	-	2020	86.90	54.19	88.55	-
SCN [13]	ResNet-18	MS_Celeb_1M	✓	✓	Adam	2020	87.03	-	88.01	-
ADDL [27]	ResNet	MS_Celeb_1M	✓	✓	Adam	2022	89.34	<u>62.16</u>	-	**66.20**
RAN-VGG16 [33]	VGG-16	VGG_Face	-	✓	-	2020	-	56.40	<u>89.16</u>	-
EfficientFace [64]	ShuffleNet [76]	MS_Celeb_1M	-	-	SGD	2021	88.36	-	-	63.70
SeNet50 [77]	SeNet-50	VGG_Face2	-	-	SGD	2018	-	-	88.80	-
EPMG [78]	VGG-Face	VGG-Face	-	-	SGD	2022	87.10	-	-	62.10
DACL [24]	ResNet-18	MS_Celeb_1M	-	-	SGD	2021	87.78	-	-	65.20
baseline	ResNet-18	ImageNet	-	-	Adam	-	85.78	58.03	87.63	58.20
Ours (proposed)	ResNet-18	ImageNet	-	-	Adam	-	**92.21**	**62.82**	**89.50**	<u>65.29</u>

presented. For fairness, the other protocols, e.g. the data splitting protocol, are kept the same as those of the state of the arts.

Among the competing methods, IPA2LT [63], RUL [74], SCN [13] solved the problem of inconsistency between labels and samples, RAN [33], MA-Net [21], EfficientFace [64] differentiated the contributions of spatial feature representation or relieved the influence of occlusion and pose factors, EPMG [78] and FDRL [11] modeled the relationship between expression-related facial regions, and ADDL [27] learned to explicitly disentangle disturbing factors. As a fundamental difference to distinguish the above works, we enhance and explore the shallow layer-semantics to improve robustness against face poses and occlusions. Table VII shows that our algorithm achieves the performances of 92.21%, 89.50%, 62.82% and 65.29% on RAF-DB, FERPlus, SFEW, and AffectNet, respectively, outperforms the corresponding baselines by the margins of 6.43%, 1.87%, 4.79%, 7.09%, respectively.

Meanwhile, our method achieved state-of-the-art performances on three of the four databases, where the improvements of 2.74%, 0.66%, and 0.34% over the 2nd best are achieved on RAF-DB, SFEW, and FERPlus databases, respectively. Compared with the algorithms, i.e. FDRL [11] and RAN-VGG-16 [33] that achieved the 2nd best, our algorithm is easy to implement, and does not need to train multiple networks, i.e. feature decomposition and reconstruction networks

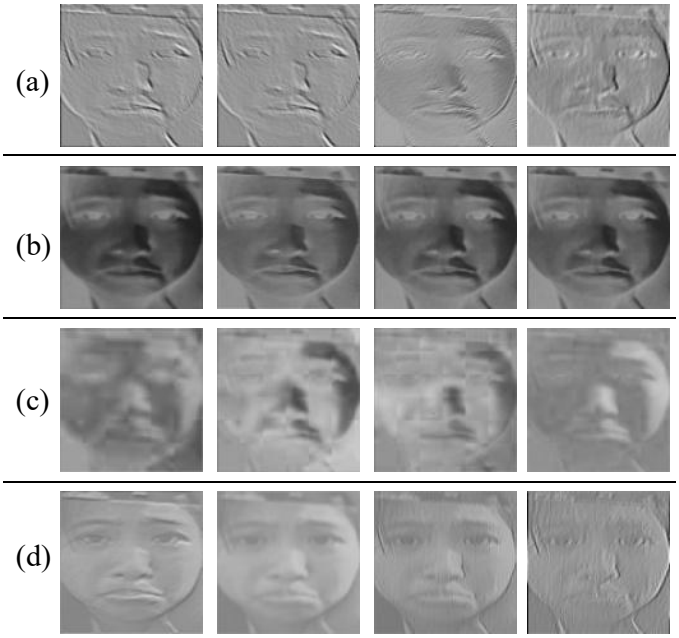


Fig. 10: Visualization of feature maps from different semantic groups of the 1st network block trained by our method.

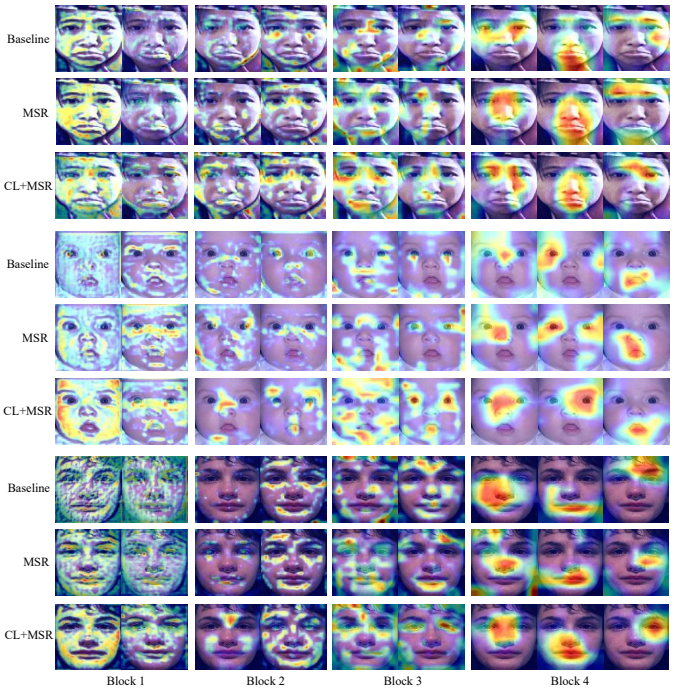


Fig. 11: The comparison of feature maps transformed from the outputs of different blocks, with the baseline (the 1st, 4th, 7th rows), multi-scale representation (MSR) module (the 2nd, 5th, 8th rows), and contrastive learning and multi-scale representation (CL+MSR) modules (the 3rd, 6th, 9th rows).

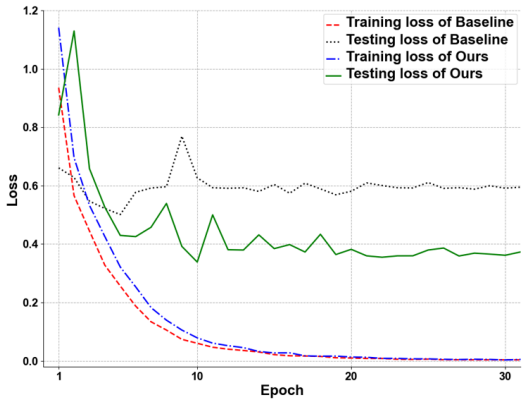


Fig. 12: Comparison of the training and testing loss evolution between baseline and our method on RAF-DB.

TABLE VIII: The mean (ν) and standard deviation (ρ) of the classification accuracy of our model under ten groups of seed settings on RAF-DB, SFEW, FERPlus, AffectNet.

Dataset	RAF-DB	SFEW	FERPlus	AffectNet
ν	92.151	62.65	89.458	65.266
ρ	0.055	0.17	0.045	0.032

in [11] or cascaded attention networks in [33].

For AffectNet, our algorithm achieved the 2nd best performance among ten algorithms. The algorithm that achieved the best performance on AffectNet, i.e. ADDL [27] published in 2022, needs to train a model with the number of parameters (Params) and Floating Point operations (FLOPs) being $20.6M$ and $3.82G$, respectively, which largely exceed ours, i.e. $14.5M$ and $2.01G$. Meanwhile, our method better trades off the performances on RAF-DB, SFEW, and AffectNet than ADDL.

Statistical significance analysis. To evaluate the significance of our improvements over the 2nd or 3rd best approaches, we resort to hypothesis testing. We set up ten groups of random seeds for each dataset, and report the mean (ν) and standard deviation (ρ) of classification accuracy in Table VIII, where the accuracy is achieved by re-training our model. The second-ranked performance of FDRL [11] on RAF-DB, i.e. 89.47 is smaller than $\nu - 3 \times \rho = 91.986$, indicating that our algorithm outperforms FDRL on RAF-DB under the significance level of 0.05. Similarly, our performance on SFEW, FERPlus or AffectNet is significantly better than that of FDRL [11], RAN-VGG16 [33] or DACL [24], respectively, under the significance level of 0.05. Meanwhile, it's worth noting that our algorithm rarely introduces additional random factors over the baseline, which may be the reason that it can achieve stable performances on the evaluated datasets.

V. Conclusions and Discussions

In this work, a simple yet effective framework of cross-layer contrastive learning is proposed to enhance the learning intensity on the shallow layers, as well as align the latent semantics between different layers. Based on the enhanced shallow-layer features, the multi-scale features from different layers are integrated adaptively with an attention mechanism.

Extensive experimental results on four public in-the-wild databases show: (1) the learning intensity on the shallow layers can be well enhanced by our method; (2) the alignment of the feature semantics between shallow and deep layers is beneficial for the recognition of fine-grained expressions; (3) the proposed algorithm can well deal with the posed and occluded faces, and achieve almost the best performances on the in-the-wild databases. The excellent performance on in-the-wild databases, good interpretability, and only a small amount of additional runtime overhead over the baseline, make our proposed model promising in practical applications.

Although competitive performance is achieved, there is still room for further improvement. First, more candidate combinations of cross layers, in addition to the employed combination of the 2nd and 4th blocks in this work, will be explored for enhancing the contrastive learning. Correspondingly, the hyperparameters of their regularization weights can be made adaptive. Second, more effective clustering algorithms for better representation of the latent semantics will be investigated. Third, more advanced contrastive learning paradigms can be considered to better align the semantics between different layers. Finally, the algorithm is general and can be exploited in the fields of micro-expression detection, cross-dataset recognition, and general object recognition.

ACKNOWLEDGMENT

The work was supported by the Natural Science Foundation of China under grants no. 62276170, 82261138629, the Science and Technology Project of Guangdong Province under grants no. 2023A1515011549, 2023A1515010688, the Science and Technology Innovation Commission of Shenzhen under grant no. JCYJ20220531101412030. The authors would like to thank the anonymous reviewers for their helpful comments and constructive suggestions. We are also very grateful to our colleague Basker George for helping revise the linguistics of this work.

REFERENCES

[1] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 818–833.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.

[3] G. Larsson, M. Maire, and G. Shakhnarovich, "Fractalnet: Ultra-deep neural networks without residuals," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017.

[4] A. Yao and D. Sun, "Knowledge transfer via dense cross-layer mutual-distillation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 294–311.

[5] M. Yu, H. Zheng, Z. Peng, J. Dong, and H. Du, "Facial expression recognition based on a multi-task global-local network," *Pattern Recognit. Lett*, vol. 131, pp. 166–171, 2020.

[6] H. Mostafa, V. Ramesh, and G. Cauwenberghs, "Deep supervised learning using local errors," *Front. Neurosci.*, p. 608, 2018.

[7] D. Sun, A. Yao, A. Zhou, and H. Zhao, "Deeply-supervised knowledge synergy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 6997–7006.

[8] Y. Garg, K. S. Candan, and M. L. Sapino, "San: Scale-space attention networks," in *Proc. Int. Conf. Data Eng. (ICDE)*, 2020, pp. 853–864.

[9] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 4700–4708.

[10] H. Yang, L. Yin, Y. Zhou, and J. Gu, "Exploiting semantic embedding and visual feature for facial action unit detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 10 482–10 491.

[11] D. Ruan, Y. Yan, S. Lai, Z. Chai, C. Shen, and H. Wang, "Feature decomposition and reconstruction learning for effective facial expression recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 7660–7669.

[12] X. Zhang, Y. Yan, J.-H. Xue, Y. Hua, and H. Wang, "Semantic-aware occlusion-robust network for occluded person re-identification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 7, pp. 2764–2778, 2021.

[13] K. Wang, X. Peng, J. Yang, S. Lu, and Y. Qiao, "Suppressing uncertainties for large-scale facial expression recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 6897–6906.

[14] J. Li, P. Zhou, C. Xiong, and S. C. Hoi, "Prototypical contrastive learning of unsupervised representations," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021.

[15] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, pp. 18 661–18 673, 2020.

[16] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "With a little help from my friends: Nearest-neighbor contrastive learning of visual representations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 9588–9597.

[17] H. Yang, U. Ciftci, and L. Yin, "Facial expression recognition by de-expression residue learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 2168–2177.

[18] Y. Fan, V. O. Li, and J. C. Lam, "Facial expression recognition with deeply-supervised attention network," *IEEE Trans. Affect. Comput.*, vol. 13, no. 2, pp. 1057–1071, 2022.

[19] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 3074–3082.

[20] H. Zhang, W. Su, J. Yu, and Z. Wang, "Weakly supervised local-global relation network for facial expression recognition," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2020, pp. 1040–1046.

[21] Z. Zhao, Q. Liu, and S. Wang, "Learning deep global multi-scale and local attention features for facial expression recognition in the wild," *IEEE Trans. Image Process.*, vol. 30, pp. 6544–6556, 2021.

[22] P. D. Marrero Fernandez, F. A. Guerrero Pena, T. Ing Ren, and A. Cunha, "Feratt: Facial expression recognition with attention net," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, June 2019.

[23] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha, *et al.*, "Resnest: Split-attention networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2022, pp. 2736–2746.

[24] A. H. Farzaneh and X. Qi, "Facial expression recognition in the wild via deep attentive center loss," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, 2021, pp. 2402–2411.

[25] Y. Liu, J. Peng, W. Dai, J. Zeng, and S. Shan, "Joint spatial and scale attention network for multi-view facial expression recognition," *Pattern Recognit.*, vol. 139, p. 109496, 2023.

[26] Y. Fan, V. Li, and J. C. Lam, "Facial expression recognition with deeply-supervised attention network," *IEEE Trans. Affect. Comput.*, 2020.

[27] D. Ruan, R. Mo, Y. Yan, S. Chen, J.-H. Xue, and H. Wang, "Adaptive deep disturbance-disentangled learning for facial expression recognition," *Int. J. Comput. Vis. (IJCV)*, pp. 1–23, 2022.

[28] L. Zhu, H. Fan, Y. Luo, M. Xu, and Y. Yang, "Temporal cross-layer correlation mining for action recognition," *IEEE Trans. Multimedia*, vol. 24, pp. 668–676, 2021.

[29] M. Karnati, A. Seal, D. Bhattacharjee, A. Yazidi, and O. Krejcar, "Understanding deep learning techniques for recognition of human emotions using facial expressions: a comprehensive survey," *IEEE Trans. Instrum. Meas.*, 2023.

[30] W. Xie, H. Wu, Y. Tian, M. Bai, and L. Shen, "Triplet loss with multistage outlier suppression and class-pair margins for facial expression recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 2, pp. 690–703, 2021.

[31] Y. Gu, H. Yan, X. Zhang, Y. Wang, Y. Ji, and F. Ren, "Towards facial expression recognition in the wild via noise-tolerant network," *IEEE Trans. Circuits Syst. Video Technol.*, 2022.

[32] Y. Li, J. Zeng, S. Shan, and X. Chen, "Occlusion aware facial expression recognition using cnn with attention mechanism," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2439–2450, 2018.

[33] K. Wang, X. Peng, J. Yang, D. Meng, and Y. Qiao, "Region attention networks for pose and occlusion robust facial expression recognition," *IEEE Trans. Image Process.*, vol. 29, pp. 4057–4069, 2020.

[34] C. Wang, S. Wang, and G. Liang, "Identity-and pose-robust facial expression recognition through adversarial feature learning," in *Proc. ACM Int. Conf. Multimedia*, 2019, pp. 238–246.

[35] B. Pan, S. Wang, and B. Xia, "Occluded facial expression recognition enhanced through privileged information," in *Proc. ACM Int. Conf. Multimedia*, 2019, pp. 566–573.

[36] B. Xia and S. Wang, "Occluded facial expression recognition with step-wise assistance from unpaired non-occluded images," in *Proc. ACM Int. Conf. Multimedia*, 2020, pp. 2927–2935.

[37] M. Karnati, A. Seal, A. Yazidi, and O. Krejcar, "Flepnet: feature level ensemble parallel network for facial expression recognition," *IEEE Trans. on Affect. Comput.*, vol. 13, no. 4, pp. 2058–2070, 2022.

[38] K. Mohan, A. Seal, O. Krejcar, and A. Yazidi, "Fer-net: facial expression recognition using deep neural net," *Neural Comput. Appl.*, vol. 33, pp. 9125–9136, 2021.

[39] H. Siqueira, S. Magg, and S. Wermter, "Efficient facial feature learning with wide ensemble-based convolutional neural networks," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 34, no. 04, 2020, pp. 5800–5809.

[40] B. Fasel, "Robust face analysis using convolutional neural networks," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, vol. 2, 2002, pp. 40–43.

[41] D. Chen, J.-P. Mei, Y. Zhang, C. Wang, Z. Wang, Y. Feng, and C. Chen, "Cross-layer distillation with semantic calibration," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 35, no. 8, 2021, pp. 7028–7036.

[42] W. Yu, X. Sun, K. Yang, Y. Rui, and H. Yao, "Hierarchical semantic image matching using cnn feature pyramid," *Comput. Vis. Image Underst. (CVIU)*, vol. 169, pp. 40–51, 2018.

[43] C. Yu, X. Zhao, Q. Zheng, P. Zhang, and X. You, "Hierarchical bilinear pooling for fine-grained visual recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 574–589.

[44] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, pp. 9912–9924, 2020.

[45] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019.

[46] S. Roy and A. Etemad, "Self-supervised contrastive learning of multi-view facial expressions," in *Proc. ACM Int. Conf. Multimodal Interact.*, 2021, pp. 253–257.

[47] R. Vemulapalli and A. Agarwala, "A compact embedding for facial expression similarity," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 5683–5692.

[48] G. Li, X. Zhu, Y. Zeng, Q. Wang, and L. Lin, "Semantic relationships guided representation learning for facial action unit recognition," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 33, no. 01, 2019, pp. 8594–8601.

[49] J. She, Y. Hu, H. Shi, J. Wang, Q. Shen, and T. Mei, "Dive into ambiguity: Latent distribution mining and pairwise uncertainty estimation for facial expression recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 6248–6257.

[50] Z. Wang, F. Zeng, S. Liu, and B. Zeng, "Oaenet: Oriented attention ensemble for accurate facial expression recognition," *Pattern Recognit.*, vol. 112, p. 107694, 2021.

[51] Q. Huang, C. Huang, X. Wang, and F. Jiang, "Facial expression recognition with grid-wise attention and visual transformer," *Inf. Sci. (INS)*, vol. 580, pp. 35–54, 2021.

[52] J. Zhou, X. Zhang, Y. Liu, and X. Lan, "Facial expression recognition using spatial-temporal semantic graph network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2020, pp. 1961–1965.

[53] Y. Li, Z. Zhang, B. Chen, G. Lu, and D. Zhang, "Deep margin-sensitive representation learning for cross-domain facial expression recognition," *IEEE Trans. Multimedia*, 2022.

[54] Y. Fu, X. Wu, X. Li, Z. Pan, and D. Luo, "Semantic neighborhood-aware deep facial expression recognition," *IEEE Trans. Image Process.*, vol. 29, pp. 6535–6548, 2020.

[55] Z. Wen, H. Wu, W. Xie, and L. Shen, "Group-wise feature orthogonalization and suppression for gan based facial attribute translation," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, 2021, pp. 3767–3774.

[56] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 1874–1883.

[57] S. Li and W. Deng, "Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 356–370, 2018.

[58] E. Barsoum, C. Zhang, C. C. Ferrer, and Z. Zhang, "Training deep networks for facial expression recognition with crowd-sourced label distribution," in *Proc. ACM Int. Conf. Multimodal Interact.*, 2016, pp. 279–283.

[59] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, *et al.*, "Challenges in representation learning: A report on three machine learning contests," *Neural Netw.*, vol. 64, pp. 59–63, 2015.

[60] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon, "Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2011, pp. 2106–2112.

[61] A. Dhall, R. Goecke, S. Luccy and T. Gedeon, "Collecting large, richly annotated facial-expression databases from movies," *IEEE Multimedia*, vol. 19, no. 03, pp. 34–41, 2012.

[62] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild," *IEEE Trans. Affect. Comput.*, vol. 10, no. 1, pp. 18–31, 2017.

[63] J. Zeng, S. Shan, and X. Chen, "Facial expression recognition with inconsistently annotated datasets," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 222–237.

[64] Z. Zhao, Q. Liu, and F. Zhou, "Robust lightweight facial expression recognition network with label distribution training," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 35, no. 4, 2021, pp. 3510–3519.

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[66] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2009, pp. 248–255.

[67] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 15 750–15 758.

[68] S. Li, W. Deng, and J. Du, "Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 2852–2861.

[69] C.-Y. Hung, C.-H. Tu, C.-E. Wu, C.-H. Chen, Y.-M. Chan, and C.-S. Chen, "Compacting, picking and growing for unforgetting continual learning," *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, pp. 13 647–13 657, 2019.

[70] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *Proc. IEEE Int. Conf. Automatic Face & Gesture Recognit. (FG)*, 2018, pp. 67–74.

[71] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, pp. 189–248, 2016.

[72] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vis. Comput.*, vol. 28, no. 5, pp. 807–813, 2010.

[73] B. Chen, W. Guan, P. Li, N. Ikeda, K. Hirasawa, and H. Lu, "Residual multi-task learning for facial landmark localization and expression recognition," *Pattern Recognit.*, vol. 115, p. 107893, 2021.

[74] Y. Zhang, C. Wang, and W. Deng, "Relative uncertainty learning for facial expression recognition," *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 34, pp. 17 616–17 627, 2021.

[75] D. Acharya, Z. Huang, D. Pani Paudel, and L. Van Gool, "Covariance pooling for facial expression recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 367–374.

[76] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 6848–6856.

[77] S. Albanie, A. Nagrani, A. Vedaldi, and A. Zisserman, "Emotion recognition in speech using cross-modal transfer in the wild," in *Proc. ACM Int. Conf. Multimedia*, 2018, pp. 292–301.

[78] J. Zhang and H. Yu, "Improving the facial expression recognition and its interpretability via generating expression pattern-map," *Pattern Recognit.*, vol. 129, p. 108737, 2022.