Information Sciences 649 (2023) 119640

Contents lists available at ScienceDirect

Information Sciences

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

Face2Nodes: Learning facial expression representations with relation-aware dynamic graph convolution networks

Fan Jiang [a], Qionghao Huang [a], Xiaoyong Mei [a,*], Quanlong Guan [b,c], Yaxin Tu [a], Weiqi Luo [b,c], Changqin Huang [a,*]

[a] Key Laboratory of Intelligent Education Technology and Application of Zhejiang Province, Jinhua, 321004 Zhejiang, China
[b] Department of Computer Science, College of Information Science and Technology, Jinan University, Guangzhou 510632, China
[c] Guangdong Institute of Smart Education, Jinan University, Guangzhou 510632, China

ARTICLE INFO

Keywords:
Facial expression recognition
Facial patch embedding
Dynamic graph construction
Relation-aware graph convolution

ABSTRACT

Deep convolutional neural networks (CNNs) have become the standard model architecture for facial expression recognition (FER). However, CNN-based models struggle to capture the structural correlations between different local regions in a face image. Recent methods based on Vision Transformer (ViT) have been introduced to capture long-range dependencies among local regions. Nonetheless, ViT-based approaches are vulnerable to facial regions unrelated to expressions and may learn redundant correlation representations due to their self-attention mechanism. To address these issues, we propose a novel graph-based model called Face2Nodes, which can flexibly learn the graph representations of facial expressions without requiring additional auxiliary facial information such as landmarks. Our Face2Nodes consists of two key components: a multi-scale feature fusion-based patch embedding and a relation-aware dynamic graph convolution network. The patch embedding method uses a multi-scale feature fusion mechanism to obtain more discriminative graph node features for further graph representation learning. A dynamic graph is constructed using the dilated k-nearest neighbors algorithm, and a relation-aware graph convolution operator is designed to learn the latent informative correlations among different nodes in the graph. Extensive experiment results show that Face2Nodes achieves state-of-the-art performance on several popular in-the-wild FER datasets, with overall accuracies of 91.41%, 91.02%, and 66.69% on the FERPlus, RAF-DB, and AffectNet databases, respectively. Furthermore, we found that CNN-based FER approaches have a more significant performance gap between pre-training and training from scratch than Face2Nodes, demonstrating that our model is more data-efficient than CNN-based approaches.

1. Introduction

In recent years, automatic facial expression recognition (FER) has gained increasing attention in the computer vision community due to its potential to enable computers to understand emotions and interact with humans. The practical implications of FER extend across diverse domains, encompassing pivotal areas such as human-computer interaction [1], detection of student engagement [2], and advancement of healthcare services [3].

* Corresponding author.
E-mail addresses: cdmxy@zjnu.edu.cn (X. Mei), cqhuang@zju.edu.cn (C. Huang).

https://doi.org/10.1016/j.ins.2023.119640
Received 14 June 2023; Received in revised form 19 August 2023; Accepted 28 August 2023

Available online 1 September 2023

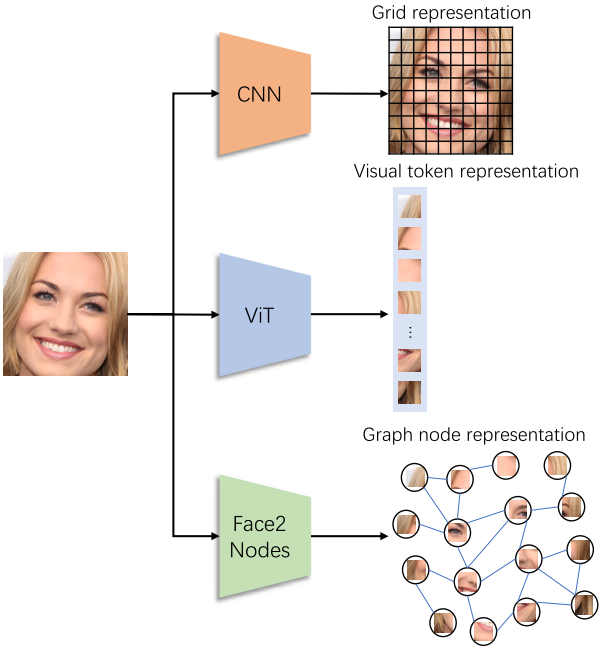

Fig. 1. The three facial expression representations: (1) Grid representation in CNN. (2) Visual token representation in ViT. (3) Graph node representation in our Face2Nodes.

Convolutional neural networks (CNNs) have achieved remarkable performance in FER and have become the standard network architecture for this task. Existing CNN-based FER models can be divided into three categories: holistic-based, local-based, and holistic-local-based methods. Holistic-based approaches extract global facial features by taking whole face images as inputs [4,5]. However, these approaches are susceptible to disturbances stemming from factors like illumination and pose, due to their limited capacity to accurately represent critical facial regions. To address this issue, local-based approaches have been proposed to learn representations of local regions by dividing a face image into several overlapped or non-overlapped local patches using attention mechanisms or facial landmarks [6,7]. Since holistic and local information complement each other, many works combine holistic and local learning patterns to extract global and part features from diverse perspectives [8,9]. Despite their ability to learn global or local representations of facial expressions, CNN-based models struggle to capture long-range correlations among different facial regions due to the spatial locality of the convolution operator. To overcome this weakness, recent Vision Transformer [10] (ViT) based methods have been proposed to extract latent dependencies between face regions, enhancing the final facial expression representations [11, 12]. These ViT-based models typically employ a stem CNN to divide a face image into several local patches, which are then fed into cascaded Transformer blocks to learn the correlations among patches. However, the visual token sequence structure in ViT is redundant and inflexible in capturing human facial expressions. Additionally, the learned token sequence may contain redundant or noisy information, as each updated patch is obtained from the features of all other patches through the self-attention mechanism. Furthermore, prior graph-based FER approaches [13,14] typically rely on auxiliary facial information such as landmarks to capture facial structural representations, complicating the model training and inference process. Therefore, effectively learning structural correlation representations among salient facial regions without any auxiliary facial clues for FER remains a challenging problem.

Fig. 1 illustrates the facial expression representation learning paradigms of CNN, ViT, and Face2Nodes. Both CNN-based and ViT-based methods treat the face image as a grid or visual token sequence structure, which is not flexible enough to learn irregular and deformable correlations among facial regions. To address this issue, we propose a novel graph-based model called Face2Nodes, which can flexibly learn relation-aware graph representations among different facial regions. Our Face2Nodes model represents the face image as a graph structure without the need for auxiliary facial clues, such as landmarks. Our model consists of two main components: a multi-scale feature fusion-based patch embedding (MSF^{2}PE) and a relation-aware dynamic graph convolution network (RDGCN). The MSF^{2}PE module divides a face image into several local patches and embeds them into feature vectors using a multi-scale feature fusion mechanism. This allows our model to fully exploit both low-level and high-level features, providing more discriminative patch features for further graph representation learning of facial expressions. The RDGCN module learns structural graph representations between different local facial patches by treating each patch as a graph node and dynamically constructing a graph for the unordered set of nodes in each RDGCN block using a dilated k-nearest neighbors (*k-NN*) algorithm. We also propose a novel graph convolution operator called relation-aware graph convolution (RG-Conv), which is designed to capture the latent informative correlations of salient graph nodes that are important for extracting facial expression representations. Unlike conventional graph convolution operators, which mainly focus on extracting the most salient node feature while dismissing other informative node features in a local graph, our relation-aware aggregation function utilizes edge features between nodes to model positive correlations among facial regions. Extensive experiment results on several FER datasets show that our Face2Nodes outperforms existing FER

methods. Furthermore, we found a larger gap in the test performance between pre-training and training from scratch in CNN-based FER models compared to Face2Nodes.

In summary, our main contributions are as follows:

- We propose a novel graph-based model for FER that can flexibly learn the structural correlations among facial regions in a face image. To the best of our knowledge, this is the first effort to adapt the vision graph neural network [15] to the FER task.
- We introduce a multi-scale feature fusion method for patch embedding that extracts patch features of a face image, including both low-level and high-level facial expression features.
- We design a new relation-aware graph convolution (RG-Conv) operator for a dynamic *k-NN* graph that captures more positive and vital latent correlations between graph nodes.
- We conducted extensive experiments on popular in-the-wild datasets, both in pre-training and training-from-scratch settings. The results of these experiments demonstrate that our Face2Nodes achieves state-of-the-art performance. Furthermore, we found that state-of-the-art CNN-based FER methods exhibit a more significant gap in test results between pre-training and training from scratch, compared to our Face2Nodes.

The remainder of this paper is organized as follows. In Section 2, we review the existing work on CNN-based and ViT-based FER approaches and graph convolutional networks. In Section 3, we describe the details of our proposed MSF^{2}PE and RDGCN. In Section 4, we present the experiment results and visualizations on three benchmark datasets. Finally, in Section 5, we present our conclusions and discuss future work.

2. Related work

In this section, we first review the classical and recent state-of-the-art FER approaches. Then we revisit the applications of graph convolutional networks in computer vision, especially its application to FER tasks.

2.1. Facial expression recognition

Early works on FER focused on the use of hand-crafted features that reflect the folds and geometric changes caused by expressions, such as histograms of oriented gradients (HOG) [16], scale-invariant feature transform (SIFT) [17], and local binary patterns (LBP) [18]. However, these hand-crafted features often lack the flexibility and generalization required to effectively handle in-the-wild scenarios, such as occlusions and variant poses. Recently, with the advent of various deep learning methods [19,20], deep learning-based approaches, particularly CNNs, have become dominant in the field of FER. As a pioneering work, Tang [21] employed a CNN to extract deep features and a linear SVM classifier for FER. To bring locally neighboring faces of the same class closer together, Liu et al. [22] proposed a new deep locality-preserving CNN (DLP-CNN) with a locality-preserving loss. Additionally, patch-based and region-based CNN frameworks have been introduced to capture the facial expression features of salient face regions. The region attention network (RAN) [7] adaptively learns the importance of facial regions by aggregating and embedding various region features produced by a CNN backbone. The patch-gated CNN (PG-CNN) [6] decomposes an input image into several patches based on the positions of related facial landmarks and reweights all patches using patch-gated units. To address the challenges posed by occlusions and non-frontal poses in FER in the wild, MA-Net [8] extracts multi-scale global features and salient local features using a multi-scale module and a local attention module. More recently, inspired by the success of vision transformers in image recognition tasks, several Transformer-based methods for FER have been proposed. For example, FER-VT [11] models the long-range dependencies among different regions of a face image using a grid-wise attention mechanism and learns high-level semantic representations using a vision Transformer encoder. To explore the relations among different facial parts, TransFER [12] combines a CNN with an improved ViT module featuring multi-attention dropping modules.

Although CNN-based and ViT-based FER models have achieved impressive performance, their ability to flexibly learn structural correlation representations in face images is limited due to the grid representation used by CNNs and the visual token sequence representation used by ViTs.

2.2. Graph convolutional networks

Graph convolutional networks (GCNs) can be broadly categorized into two approaches: spatial-based and spectral-based. Spatial-based GCNs [23,24] consider the spatial neighboring nodes of each node and perform convolution on a graph, while spectral-based GCNs [25,26] explore the locality of graph convolution using spectral graph theory.

GCNs have been widely applied to computer vision tasks such as point cloud classification [27] and skeleton-based action recognition [28]. In the field of FER, recent studies have demonstrated the effectiveness of graph convolutional networks (GCNs) in learning discriminative facial expression representations from images or videos. Liu et al. [13] proposed a graph-structured action unit (AU) based FER framework that models action-unit-guided facial region features in a graph structure, with AU-related regions as nodes and distances between two landmarks as edges. Zhao et al. [14] proposed a GCN-based two-stream learning framework for geometry-aware FER to enrich emotion-related feature representations with facial structural knowledge. Zhou et al. [29] proposed a spatial-temporal semantic graph network for video-based FER that exploits dynamic facial topology structure to capture spatial

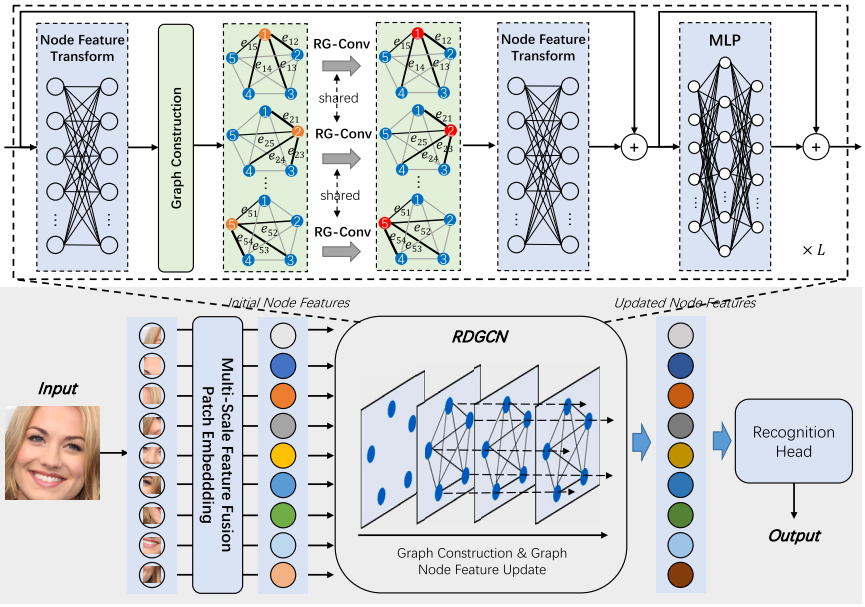

Fig. 2. The framework of the Face2Nodes. The model contains three components: (1) multi-scale feature fusion-based patch embedding, (2) relation-aware dynamic graph convolution network, and (3) recognition head.

and temporal patterns of facial expression. Zhao et al. [30] presented a geometry-guided dynamic FER model based on GCNs and Transformer that uses facial landmarks to construct a spatial-temporal graph for facial expression sequence representations.

Overall, the GCN-based FER models typically use CNNs as the backbone network for feature extraction and employ a GCN to encode geometric features that enrich the final global feature. In contrast, our approach only uses images as inputs, without requiring any auxiliary facial information such as action units or landmarks.

3. Methodology

In this section, we first introduce the framework of our Face2Nodes. Then we describe the patch embedding method based on multi-scale feature fusion and relation-aware dynamic graph convolution network based on dilated *k-NN* and RG-Conv. Finally, we present the learning objective function of our model.

3.1. Framework of Face2Nodes

As illustrated in Fig. 2, our Face2Nodes comprises three parts: the MSF^{2}PE module, the RDGCN module, and the recognition head.

The **MSF^{2}PE** module is designed to extract both low-level and high-level feature embeddings of facial image patches using a multi-scale feature fusion mechanism. The feature vectors of the patches are treated as a set of initial features for unordered nodes and are fed into the RDGCN module to learn the graph representations of facial expressions.

The **RDGCN** module contains several cascaded blocks consisting of a relation-aware graph convolution module and a multi-layer perceptron (MLP). We use a dilated *k-NN* algorithm to dynamically construct a graph for the nodes before applying the aggregation and update operators of RG-Conv. To learn the latent correlations among key facial regions, we utilize edge features as input for the neighborhood aggregator and reweight different edge features to extract positive correlations within the graph. We also transform node features using a fully connected layer both before graph construction and after RG-Conv, in order to increase the diversity of node features. To address the gradient vanishing issue during training, we utilize skip connections to the RG-Conv and MLP modules as suggested by He et al. [31]. The output node features of the RDGCN serve as the final graph representation of a facial expression to be recognized.

The **recognition head** is designed to classify the graph representation into a facial expression. It is implemented using two fully connected layers with batch normalization and the Gaussian error linear unit (GELU) [32] activation function.

3.2. Multi-scale feature fusion-based patch embedding

Patch-based models for FER typically divide an input image into several patches. Existing patch embedding methods can be broadly categorized into two approaches: linear projection-based and convolution-based. Recent works have shown that convolution-based patch embedding methods outperform linear projection-based methods due to their ability to model local spatial context [33]. However, most convolution-based patch embedding methods only exploit high-level semantic features, which may limit their ability to capture the correlations of low-level patch features.

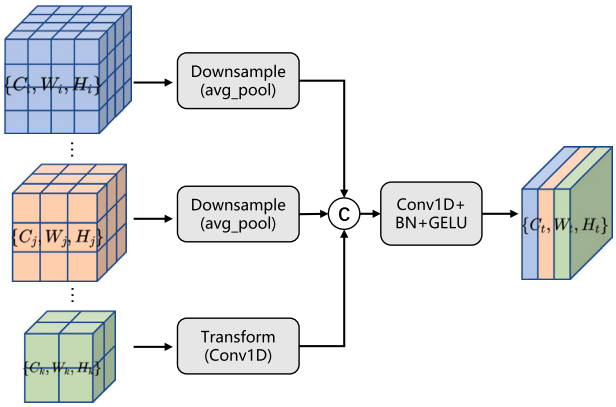

Fig. 3. The details of the multi-scale feature fusion process for patch embedding. The downsample operator is implemented by a 1×1 convolution and an average pooling function.

Inspired by the work of Lin et al. [34], we propose a multi-scale feature fusion-based method for patch embedding, which is designed to fuse multiple-level features from different stages. Given an image with size $C \times H \times W$, our goal is to divide it into N patches and encode all patches into a set of feature vectors $X = \{x_i \in \mathbb{R}^D | i = 1, 2, ..., N\}$, where D is the feature dimension. The MSF^{2}PE module consists of multi-scale feature extraction and multi-scale feature fusion. We adopt the feature extraction framework used by He et al. [31] to extract multi-scale feature maps from four different stages $S = \{1, ..., 4\}$. We denote the features as X_s, where $s \in S$ is the stage of the feature extractor. Specifically, the feature maps $X_1 \in C_1 \times \frac{W}{4} \times \frac{H}{4}$ are extracted in the first stage, $X_2 \in C_2 \times \frac{W}{8} \times \frac{H}{8}$ are extracted in the second stage, $X_3 \in C_3 \times \frac{W}{16} \times \frac{H}{16}$ are extracted in the third stage, and $X_4 \in C_4 \times \frac{W}{32} \times \frac{H}{32}$ are extracted in the fourth stage. For the multi-scale feature fusion process (shown in Fig. 3), we first transform the resolution of $X_s \in \mathbb{R}^{C_s \times W_s \times H_s}$ into a target size $W_t \times H_t$, where W_t and H_t denote the width and height of the fused feature map, respectively. For simplicity, we treat feature X_4 as the target size of the feature map, *i.e.*, $W_t = W_{X_4}$ and $H_t = H_{X_4}$. Specifically, we use a 1×1 convolution and average pooling to downsample all features except X_4:

$$X'_s = \text{AvgPool}(\text{Conv1D}(X_s)), s = (1, 2, 3), \tag{1}$$

where $X'_s \in \mathbb{R}^{C'_s \times W_t \times H_t}$ denotes the updated feature map. The output feature X_4 of the last stage is transformed using a 1×1 convolution operator:

$$X'_4 = \text{Conv1D}(X_4), \tag{2}$$

where $X'_4 \in \mathbb{R}^{C'_4 \times W_t \times H_t}$ is the transformed feature map. We then concatenate the transformed features (X'_1, X'_2, X'_3, X'_4) along the channel dimension:

$$X_c = \text{Concat}(X'_1, X'_2, X'_3, X'_4), \tag{3}$$

where Concat is channel-wise concatenation and $X_c \in \mathbb{R}^{C_a \times W_s \times H_s}$ with $C_a = C'_1 + C'_2 + C'_3 + C'_4$. Finally, we feed the concatenated feature X_c into a feature fusion function to obtain the fused feature:

$$X_{\text{f}} = \phi(X_c; \Theta_\phi), \tag{4}$$

where $\phi(\cdot; \Theta_\phi)$ denotes a single-layer MLP with parameters Θ_ϕ, and $X_{\text{f}} \in \mathbb{R}^{C_t \times W_t \times H_t}$ denotes the final fused feature. In practice, we implement ϕ using a 1×1 convolution layer with batch normalization and GELU activation function.

3.3. Relation-aware dynamic graph convolution networks

Dynamic Graph Construction. The patch embedding features $X = \{x_i \in \mathbb{R}^D | i = 1, 2, ..., N\}$ can be viewed as a discrete set of nodes $\mathcal{V} = \{v_i | i = 1, 2, ..., N\}$. We define the input graph at the l-th graph convolutional layer as $\mathcal{G}^l = (\mathcal{V}^l, \mathcal{E}^l)$, where $\mathcal{E}^l$ represents all edges of the graph. We consider each node v_i as the centroid of a local graph. In general, graph convolutional networks with shallow layers have a fixed graph structure, which can lead to an over-smoothing phenomenon. To address this issue and increase the receptive field in conventional graph convolution networks, we use a dilated *k-NN* algorithm [35] to dynamically search for k neighboring nodes $\mathcal{N}(v_i)$ for node v_i before every graph convolution layer. Instead of sampling the nearest neighboring nodes from the top k neighbors, we uniformly search for k nodes by skipping every d neighbors within $k \times d$ neighboring nodes, where d denotes the dilation rate. For example, for a centroid node v_i, the dilated neighboring nodes of v_i at the l-th layer can be represented by $\mathcal{N}^l_d(v_i)$. The nearest neighbors $\{n_1, n_2, ..., n_{k \times d}\}$ are the sorted $k \times d$ neighboring nodes. Thus, the k neighbors with dilation d of v_i can be expressed as:

$$\mathcal{N}_d^l(v_i) = \{n_1, n_{1+d}, n_{1+2d}, \ldots, n_{1+(k-1)d}\}. \tag{5}$$

In addition, we use a pairwise distance metric to sample the nearest neighbors for node v_i. In practice, we calculate the Euclidean distance between nodes for the dilated *k-NN* algorithm in feature space. Finally, we obtain a set of dilated *k-NN* graphs $\mathcal{G}^l = \{\mathcal{G}_i^l \in \mathbb{R}^{k\times D} | i = 1, 2, \ldots, N\}$ at the l-th layer. In this way, the constructed *k-NN* graph is different for each layer. On the other hand, the dilation mechanism is beneficial for enlarging the receptive field of deeper layers without increasing the kernel size of graph convolution.

Relation-aware Graph Convolution. GCNs are an extension of CNNs for graph representation learning. Classical spatial-based graph convolution operators typically aggregate the neighboring node features of a central node, and then use the feature of the central node and the aggregated feature to update the center node feature. For most graph convolutional network frameworks, a general graph convolution with aggregation and update operations can be formulated as follows:

$$x_i^{l+1} = f\left(x_i^l, g\left(x_i^l, \mathcal{N}(x_i^l); \Theta_g\right); \Theta_f\right), \tag{6}$$

where $g(\cdot;\Theta_g)$ is a node feature aggregation function parameterized by Θ_g and $f(\cdot;\Theta_f)$ is an update function with learnable parameters Θ_f. x_i^l and x_i^{l+1} are the node features of node v_i at the l-th layer and $(l+1)$-th layer, respectively. $\mathcal{N}(x_i^l)$ is the set of neighbor nodes of x_i^l at the l-th layer. Several implementation variants exist for the aggregation function $g(\cdot;\Theta_g)$, such as max-pooling and attention aggregators. However, traditional aggregators, which are designed to explore the neighboring information of a node for graph data, may not be well adapted to learning facial expression representations in a face image. For example, the max-pooling aggregator is used by many graph convolution frameworks due to its simplicity, but it only focuses on the most salient node feature in a local graph while ignoring other informative nodes. Consequently, we believe that max-pooling or other aggregators may fail to capture latent correlations between key facial regions such as the mouth and cheeks for FER. To address this issue, we propose a relation-aware aggregation function to capture informative correlations among facial parts for conventional graph convolution operators, which we call RG-Conv for short. Our RG-Conv contains a relation-aware node feature aggregator $g(\cdot;\Theta_g)$ and a node feature updater $f(\cdot;\Theta_f)$, with the relation-aware aggregator $g(\cdot;\Theta_g)$ being the core operator of RG-Conv. The key idea behind the relation-aware aggregator is to use edge features, rather than the features of neighboring nodes, for neighborhood aggregation. Furthermore, we compute edge weights to extract important and positive correlations since the *k-NN* graph may contain noisy or redundant edges. More specifically, for the feature x_i of node v_i at the l-th layer, we define the aggregation operation as follows:

$$g(\cdot;\Theta_g) = \sum_{x_j^l \in \mathcal{N}(x_i^l)} h(x_j^l - x_i^l; \Theta_h) \cdot (x_j^l - x_i^l), \tag{7}$$

where $h(\cdot;\Theta_h)$ is a learnable function with parameters Θ_h that learns importance weights for correlations between different facial parts. We implement $h(\cdot;\Theta_h)$ using a fully connected layer and a sigmoid function in our experiments. x_j^l is the neighboring node feature of center node v_i. It is worth noting that we use edges $x_j^l - x_i^l$ (*i.e.*, relations) between a centroid node and its neighboring nodes as input for the aggregator. In other words, $h(x_j^l - x_i^l; \Theta_h)$ can be viewed as the importance weights between x_i^l and x_j^l. Moreover, we summarize all the weighted correlation edge features as aggregating features in the graph. As a result, our RG-Conv is able to capture latent relations among facial regions. For the node feature updater $f(\cdot;\Theta_f)$, we concatenate the node feature x_i^l with the aggregating features from $g(\cdot;\Theta_g)$ as its input. The feature x_i^{l+1} of node v_i updated by the RG-Conv operator at the $(l+1)$-th layer can be formally defined as follows:

$$x_i^{l+1} = f([x_i^l, g(\cdot;\Theta_g)]; \Theta_f), \tag{8}$$

where $x_i^{l+1} \in \mathbb{R}^{2\times D}$, $[\cdot,\cdot]$ is a concatenation operation, and the input graph feature dimension of the updater $f(\cdot;\Theta_f)$ is $2 \times D$. The node feature updater can be realized using a learnable function. In our model, we implement the feature update operator $f(\cdot;\Theta_f)$ as an MLP with batch normalization and a GELU activation function.

RDGCN Block. The RDGCN module consists of L cascaded identical blocks. Each RDGCN block mainly contains graph construction, RG-Conv, and an MLP. The RDGCN module consists of L cascaded identical blocks, with each RDGCN block containing graph construction, RG-Conv, and an MLP. Specifically, for the l-th block, we first transform the features $X^l \in \mathbb{R}^{N\times D}$ of all nodes in graph $\mathcal{G}^l$ into the same feature domain to improve node feature diversity. We then construct a *k-NN* graph and perform the RG-Conv operator to update the graph feature, followed by a linear transformation layer. These operations can be expressed as follows:

$$Z^l = \mathcal{F}_{out}^l(\mathrm{RGConv}(\mathcal{F}_{in}^l(X^l; \Theta_{\mathcal{F}_{in}})); \Theta_{\mathcal{F}_{out}}) + X^l, \tag{9}$$

where $Z^l \in \mathbb{R}^{N\times D}$, $\mathcal{F}_{in}^l(\cdot;\Theta_{\mathcal{F}_{in}})$ and $\mathcal{F}_{out}^l(\cdot;\Theta_{\mathcal{F}_{out}})$ are fully connected layers. Each RG-Conv operator is followed by batch normalization and a GELU activation function. Finally, we use a two-layer MLP $\mathcal{M}(\cdot;\Theta_{\mathcal{M}})$ to further enhance node feature transformation capacity:

$$X^{l+1} = \mathcal{M}(Z^l; \Theta_{\mathcal{M}}) + Z^l, \tag{10}$$

where $X^{l+1} \in \mathbb{R}^{N\times D}$ is the output of the l-th RDGCN block. In our experiments, the dimension of the hidden layer in $\mathcal{M}(\cdot;\Theta_{\mathcal{M}})$ is set to four times X^l, *i.e.*, $4 \times D$. The output of the last RDGCN block is the final learned graph representation of the facial expression.

Algorithm 1: Face2Nodes.

Input: Training Images $\mathcal{X}$ and ground-truth labels $\mathcal{Y}$ with C categories, $MaxEpoch$, num_iters, L blocks of RDGCN
Output: The predicted distribution of facial expression categories
Initialize the learning parameters Θ with random values or pre-trained weights, $e = 0$
while $e < MaxEpoch$ **do**
  for $i = 0$ **to** num_iters **do**
    Sample a mini-batch set_{batch} from $(\mathcal{X}, \mathcal{Y})$;
    Extract node features X_f from set_{batch} by Eq. (4);
    for $l = 0$ **to** L **do**
      Construct a dilated *k-NN* graph $\mathcal{G}^l$ by Eq. (5);
      Update the graph node features X^l by Eq. (9) and Eq. (10);
    end for
    Compute the cross-entropy loss with label smoothing $\mathcal{L}_{SCE}$ by Eq. (11);
    Update the learning parameters Θ;
  end for
  $e = e + 1$;
end while

3.4. Objective function

We use a cross-entropy loss function as the learning objective for our model. To mitigate the issue of overfitting in the classifier, we employ label smoothing to regularize the cross-entropy loss. The loss with label smoothing can be expressed as follows:

$$\mathcal{L}_{SCE} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{W_{y_i}^T x_i}}{\sum_{j=1}^{C} e^{W_j^T x_i}} \cdot q(x_i), \tag{11}$$

where N is the size of a mini-batch, C is the number of classes, and $q(\cdot)$ is the updated label distribution regularized by label smoothing. The label distribution $q(\cdot)$ can be described as:

$$q(x_i) = (1-\xi)\pi(x_i) + \frac{\xi}{C}, \tag{12}$$

where ξ denotes a probability hyperparameter, and $\pi(\cdot)$ denotes the ground-truth label distribution. Algorithm 1 details the training procedure of our Face2Nodes.

4. Experiments

In this section, we evaluate the effectiveness of our Face2Nodes on three widely used datasets for facial expression recognition, using two different training settings. We conduct extensive ablation studies to assess the contribution of each component of Face2Nodes and further validate the model architecture with different hyperparameters. Additionally, we provide visualizations of the learned feature distributions, class activation maps, and feature maps to intuitively demonstrate the superiority of our method.

4.1. Datasets

FERPlus [36] is an extension of the original FER2013 benchmark [37], which is a large-scale and unconstrained dataset collected automatically by a web search engine. The dataset includes 28,709 training images, 3,589 validation images, and 3,589 test images. These images depict eight basic expressions: happiness, surprise, sadness, anger, disgust, fear, neutral, and contempt. All images have been resized to 48x48 grayscale pixels. Due to unreliable original emotion labels in the FER2013 dataset, it was relabeled by ten crowd-sourced annotators using a voting strategy to assign images to one of the eight emotion categories.

RAF-DB [22] is a real-world facial expression database that contains 29,672 face images collected from the Internet. These images were independently annotated by 40 trained annotators with basic or compound expression categories. In our experiments, we used the single-label subset with seven basic expressions. This subset includes 12,271 training images and 3,068 test images.

AffectNet [38] is currently the largest publicly available FER dataset. It contains over one million images downloaded from the Internet using three major search engines with emotion-related labels. Of these images, approximately 450,000 have been manually annotated with either discrete categorical or continuous dimensional labels. In our experiments, we followed the methodology of Xue et al. [12] and used 280,000 images for training and 3,500 images for testing with seven expression categories.

4.2. Experiment setup

For network training, we used the Adam optimizer with an initial learning rate of 0.001 and a weight decay of 0.0001. A cosine annealing schedule is used as the learning rate scheduler. The objective function is the cross-entropy loss with label smoothing regularization, where the probability ξ in Eq. (12) is set to 0.1. The mini-batch size is 64, and training is conducted for 300 epochs. Training samples are augmented with random horizontal flips, random scale, and color jitter. To demonstrate the effectiveness of our proposed methods, we adapt the isotropic network architecture of ViG [15] to FER as our baseline model. Our experiments are conducted on a single NVIDIA TESLA V100 GPU.

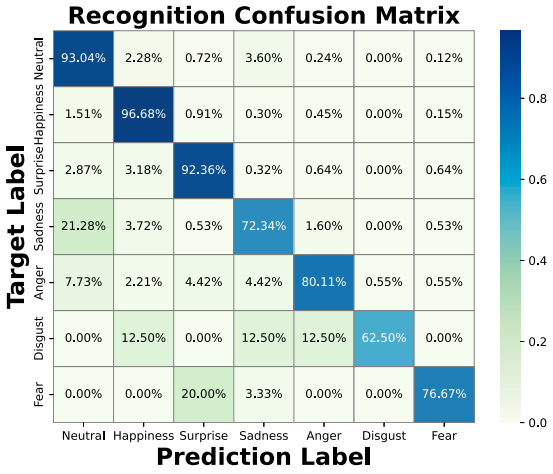

Fig. 4. The confusion matrix of Face2Nodes on the FERPlus dataset.

4.3. Implementation details

We use ResNet-18 [31] as the feature extraction framework for patch embedding. An image was divided into $N = 49$ patches and transformed into a set of node feature vectors with a dimension of $D = 512$. The RDGCN consists of a stack of $L = 8$ identical basic blocks. The input and output feature dimensions of each RDGCN block are $D = 512$. We use dilated *k-NN* to find $k = 9$ nearest neighbors and construct a local graph for each node, with a dilated rate of $\lceil l/4 \rceil$ at the l-th block. The recognition head is implemented by an MLP with two fully connected layers. The first fully connected layer is followed by batch normalization and the GELU activation function, and the hidden layer dimension of the MLP is $D/2 = 256$. We use multi-task cascaded convolutional networks (MTCNN) [39] to detect and align faces in raw FER datasets. All images are aligned and resized to 224×224 pixels. Following [40], we pre-train our Face2Nodes on the MS-Celeb-1M face dataset [41] in the pre-training experiment setting. Face2Nodes is implemented using the PyTorch toolkit.

4.4. Comparison with the state-of-the-art methods

We compare the performance of our Face2Nodes with existing state-of-the-art approaches in terms of overall evaluation accuracy on three widely used datasets: FERPlus, RAF-DB, and AffectNet. For methods that do not provide experiment results for training from scratch, we report results on the three FER datasets by reimplementing them using official source codes. To ensure a fair comparison, we conduct experiments three times on the datasets for MA-Net and EAC and choose the best result as the outcome. We focus on the comparison of four recent CNN-based and ViT-based FER methods, *i.e.*, MA-Net [8], EAC [40], FER-VT [11] and TransFER [12]. MA-Net and EAC are CNN-based FER models that both adopt ResNet as the facial expression feature learning network. FER-VT and TransFER are ViT-based FER models that both use a Transformer encoder as the correlation learning module of the facial region.

Results on FERPlus. Compared to the prior state-of-the-art methods on the FERPlus benchmark, our model achieves the best performance with an accuracy of 91.41% in the pre-training setting, as shown in Table 1a. Both FER-VT [11] and TransFER [12] are Transformer-based FER models. EAC [40] is the latest CNN-based model for FER, which uses a ResNet-18 pre-trained on MS-Celeb-1M as the backbone network. Our Face2Nodes also achieves the highest accuracy of 89.71% in the setting of training from scratch, outperforming FER-VT's 89.28% and EAC's 75.77%. For training from scratch, it is worth noting that our method outperforms SelfCureNet [43] and EAC by 6.29% and 13.94%, respectively. Fig. 4 illustrates the confusion matrix of Face2Nodes on the test set of FERPlus. The classification for the *Happiness* expression achieves the highest accuracy of 96.68%, followed by *Neutral* with 93.04% accuracy, *Surprise* with 92.36% accuracy, and *Anger* with 80.11% accuracy. Due to insufficient training data for the *Disgust* expression, the accuracy of *Disgust* is at a relatively low level (62.50%). Specifically, *Disgust* is wrongly recognized as *Happiness* by 12.5%, *Sadness* by 12.5%, and *Anger* by 12.5%.

Results on RAF-DB. Table 1b compares Face2Nodes with other state-of-the-art methods on the RAF-DB dataset. With a pre-trained model, Face2Nodes achieves an accuracy of 91.02%, which is an impressive performance compared to the other models. Face2Nodes also achieves the best performance (84.77%) over the other compared models for training from scratch. We observed a significant performance gap between pre-training and training from scratch for CNN-based FER methods. For instance, MA-Net achieves 88.40% accuracy with a pre-trained model, but only 67.48% accuracy when trained from scratch. Fig. 5 shows the confusion matrix of Face2Nodes on the test set of RAF-DB. Face2Nodes achieves more than 88% accuracy for four expressions, *i.e.*, *Happiness*, *Neutral*, *Sadness*, and *Surprise*. Nevertheless, the *Fear* expression is only correctly classified 63.51%. As a result, 16.22% of the *Fear* expressions are wrongly classified as *Surprise*, and 10.81% of the *Fear* expressions are wrongly classified as *Sadness*, which demonstrates that the learned representations of Face2Nodes for *Fear* are close to *Surprise* and *Sadness*.

As shown in Table 1c, our pre-trained model achieves an accuracy of 66.69% on AffectNet, outperforming the recent state-of-the-art CNN-based (*e.g.*, EAC [40]) and ViT-based (*e.g.*, TransFER [12]) methods. For instance, our model outperforms the latest EAC

Table 1
Comparison with other state-of-the-art results on the three FER datasets. * denotes the experiment results of our re-implementations from the official source codes. + denotes that both AffectNet and RAF-DB are used as the joint training dataset. _ denotes the second-best performance.

(a) Comparison on FERPlus

Method	Acc. (% Pre-trained)	Acc. (% From scratch)
PLD [36]	85.10	–
CCNN [42]	87.40	–
SelfCureNet [43]	88.01	83.42
RAN [7]	88.55	–
KTN [44]	90.49	–
FER-VT [11]	90.04	89.28
TransFER [12]	90.83	–
EAC [40]	89.64	75.77*
MRAN [45]	89.59	–
Face2Nodes (ours)	**91.41**	**89.71**

(b) Comparison on RAF-DB

Method	Acc. (% Pre-trained)	Acc. (% From scratch)
RAN [38]	86.90	–
GA-FER [14]	87.52	–
SelfCureNet+ [43]	88.14	78.31
FER-VT [11]	88.26	84.31
MA-Net [8]	88.40	67.48*
TransFER [12]	90.91	–
EAC [40]	89.99	73.73*
MRAN [45]	90.03	–
EDGL-FLP [46]	89.63	–
SSF-ViT [47]	90.98	–
Face2Nodes (ours)	**91.02**	**84.77**

(c) Comparison on AffectNet

Method	Acc. (% Pre-trained)	Acc. (% From scratch)
RAN [38]	59.50	–
SelfCureNet [43]	60.23	47.28
KTN [44]	63.97	–
MA-Net [8]	64.53	55.49*
TransFER [12]	66.23	–
EAC [40]	65.32	55.75*
EDGL-FLP [46]	61.25	–
SSF-ViT [47]	66.04	–
MRAN [45]	66.31	–
Face2Nodes (ours)	**66.69**	**58.31**

(55.75%) by 2.56% in the training from-scratch setting. We observe that the test performance on AffectNet is not desirable compared with the test results of FERPlus and RAF-DB. We conjecture that this may be due to the presence of numerous uncertain facial expression images and noisy labels in the AffectNet dataset, which may have disturbed the representation learning of our model. It is worth noting that TransFER exploits both MS-Celeb-1M and ImageNet [48] as the joint pre-training datasets, while we only pre-trained our model on MS-Celeb-1M. Fig. 6 illustrates the confusion matrix of Face2Nodes on the test set of AffectNet. Face2Nodes achieves the best accuracy of 86.40% for the classification of *Happiness*. However, the classification accuracy for other expressions is much lower than that for *Happiness*. For instance, the classification for *Fear* achieves an accuracy of 60.20%. Moreover, *Fear* is wrongly classified as *Surprise* by 18.60%, similar to the results on FERPlus and RAF-DB. Moreover, 14.38% of the classifications for *Anger* are incorrectly recognized as *Neutral*, which is inconsistent with our human intuition. Our Face2Nodes can flexibly learn relation-aware graph representations among different facial regions and capture the latent informative correlations of salient graph nodes important for extracting facial expression representations. In addition, our Face2Nodes only uses k-nearest neighbor nodes (facial patches) to update the representation of a node, minimizing interference from regions irrelevant to facial expressions. We found that some face images labeled as *Anger* in the test set of AffectNet actually depicted *Neutral* expressions. This suggests that the training and test sets of AffectNet include numerous noisy samples, which may explain why the test performance on AffectNet was not desirable.

Model Complexity. Table 2 compares model complexity against state-of-the-art CNN-based and ViT-based FER methods on the RAF-DB dataset. Table 2 compares the model complexity of our Face2Nodes with state-of-the-art CNN-based and ViT-based FER methods on the RAF-DB dataset. We observe that our Face2Nodes has the lowest computational cost compared to the other state-of-

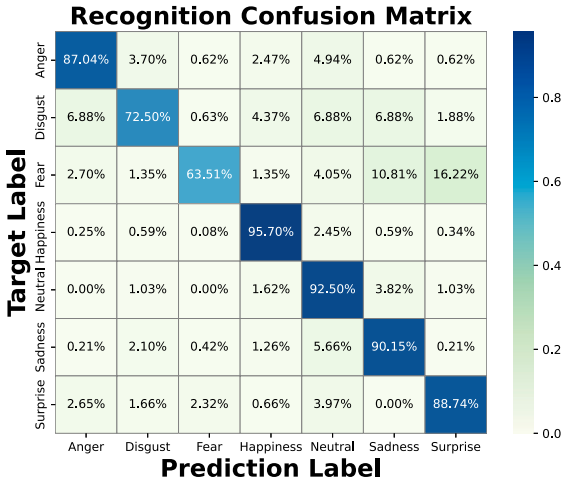

Fig. 5. The confusion matrix of Face2Nodes on the RAF-DB dataset.

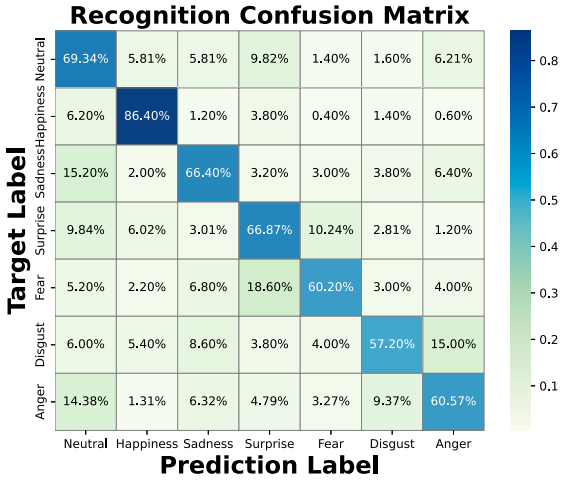

Fig. 6. The confusion matrix of Face2Nodes on AffectNet dataset.

Table 2
Comparison of complexity and accuracy against state-of-the-art CNN-based and ViT-based methods on RAF-DB.

Method	Params (M)	FLOPs (G)	Acc. (%)
MA-Net [8]	50.54	3.65	88.40
EAC [40]	23.52	11.69	89.99
FER-VT [11]	50.83	6.08	88.26
Face2Nodes (ours)	36.58	3.06	91.02

the-art CNN-based and ViT-based FER methods and achieves the best performance (*i.e.*, 91.02%) with fewer parameters compared to the other compared models. In particular, in terms of time complexity, our Face2Nodes is 3.06G FLOPs, while the FLOPs of FER-VT, MA-Net, and EAC are 6.08G, 3.65G, and 11.69G, respectively. This demonstrates that our model is faster than the recent CNN-based and ViT-based FER methods in terms of inference speed.

In summary, the comparison results on the three datasets demonstrate that our graph-based architecture has a more powerful ability to model latent facial region correlations than both CNN-based and ViT-based approaches. On the other hand, the comparison provides an insight that CNN-based and Transformer-based FER methods consume more data than our graph-based framework while achieving similar performance.

Table 3
Overall accuracy (%) of the models with different patch embedding methods on FERPlus.

Method	Pre-trained	Gain	From scratch	Gain
Base Patch Embedding [15]	89.47±0.25	-	86.54±0.33	-
MSF^{2}PE (ours)	**91.41**±0.13	**1.94**	**89.71**± 0.29	**3.17**

Table 4
Overall accuracy (%) of the models with different graph convolution operators on FERPlus.

Method	Pre-trained	Gain	From scratch	Gain
MR-GraphConv [35]	90.72±0.27	-	88.65±0.32	-
RG-Conv (ours)	**91.41**±0.17	**0.69**	**89.71**±0.23	**1.06**

Table 5
Accuracy with different depths of RDGCN on FERPlus.

Depth	4	6	8	10	12
Acc. (%)	90.90±0.16	91.18±0.15	**91.41**±0.13	90.95±0.10	90.99±0.11

Table 6
Accuracy with a different number of graph nodes on FERPlus.

# Nodes	Acc. (%)	Gain (%)
49	**91.41**±0.13	-
196	90.37±0.09	-1.04
784	89.86±0.06	-1.55

4.5. Ablation study

To demonstrate the effectiveness of our proposed modules, we conduct extensive ablation studies on FERPlus since it is a relatively clean dataset compared with RAF-DB and AffectNet.

Effectiveness of MSF^{2}PE. To validate the efficacy of our proposed MSF^{2}PE, we compared it to a baseline method that uses basic convolution-based patch embedding from [15]. Both methods used RG-Conv as the graph convolution operator in RDGCN. As shown in Table 3, models incorporating MSF^{2}PE significantly outperformed those using the baseline patch embedding method in both pre-training and from-scratch settings. For the pre-training mode, our MSF^{2}PE improved the baseline method from 89.47% to 91.41%. In particular, our method significantly improved the baseline model by 3.17% in training from scratch. These results demonstrate that multi-scale feature fusion-based patch embedding provides more powerful patch features than conventional patch embedding approaches.

Effectiveness of RG-Conv. To validate the effectiveness of our proposed RG-Conv operator, we compared it to a baseline method that employs MR-GraphConv [35] as the base graph convolution operator in RDGCN. Both methods utilized MSF^{2}PE as the patch embedding module. Table 4 shows the results of the ablation study on the effect of the RG-Conv operator. The model incorporating RG-Conv outperformed the baseline model using MR-GraphConv in both pre-training and from-scratch training modes on the FERPlus dataset. Specifically, our RG-Conv improved the baseline method by 0.69% and 1.06% in pre-training and training-from-scratch modes, respectively. These results show that our proposed RG-Conv is more effective than MR-GraphConv in modeling node relations. MR-GraphConv uses a max-pooling aggregator that emphasizes only significant neighbor node features in a local *k-NN* graph, disregarding other potentially informative neighbor nodes. In contrast, our RG-Conv calculates correlation weights for edge features representing relationships among facial regions between a central node and its neighboring nodes. These weighted correlation edge features are then summed to update a node's representation, facilitating more beneficial information exchange between nodes than MR-GraphConv.

Determination of RDGCN depth. We conducted ablation experiments using five different depths to assess the impact of varying the number of RDGCN blocks (*i.e.*, 4, 6, 8, 10, 12). Table 5 shows that the model with 8 RDGCN blocks achieved the best performance of 91.41%. Furthermore, model performance increased as the number of RDGCN blocks increased from 4 to 8 but decreased when RDGCN depth was greater than 8. This suggests that models with ten or more RDGCN blocks tend to overfit the limited training dataset, leading to relatively poor performance on the test dataset.

The Number of Graph Nodes. Our model features an isotropic architecture in which each RDGCN layer has a fixed number of input graph nodes, allowing us to define the number of graph nodes in the MSF^{2}PE module. We considered three input graph node configurations (*i.e.*, 49, 196, and 784) to balance performance and training efficiency. As shown in Table 6, the model with 196 input graph nodes achieved an accuracy of 90.37%, lower than the model with 49 graph nodes by 1.04%. The model with 784 graph nodes only achieved an accuracy of 89.86%, lower than the model with 49 nodes by 1.55%. This result demonstrates that more

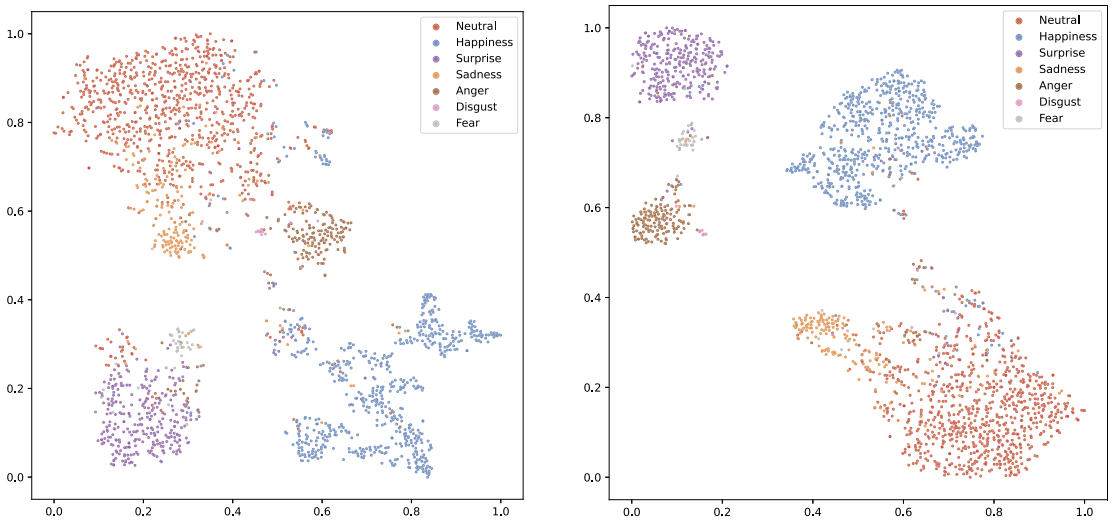

Fig. 7. Visualization of the learned features on the test set of FERPlus for Face2Nodes and the baseline model using t-SNE. The colors of the emotion categories are consistent for the two sub-figures. The feature embedding distribution of our Face2Nodes has a lower intra-class distance and a clearer inter-class boundary than the baseline model.

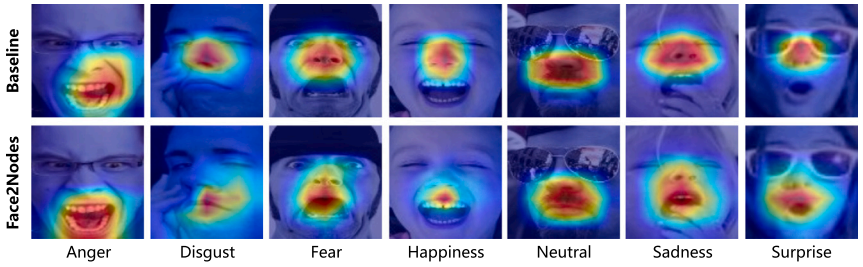

Fig. 8. Visualization of class activation maps for our Face2Nodes and the baseline model. The generated images of different facial expressions are illustrated from the test set of RAF-DB. Compared to the baseline model, our model has larger attention areas and provides higher responses on key facial parts such as mouths, effectively covering key facial areas for various expressions.

graph nodes can harm facial expression representation generalization ability. A possible explanation for this phenomenon is that a larger number of graph nodes results in more fine-grained patch representations, which may cause models to overfit on the limited training dataset.

4.6. Visualization analysis

We conduct visualizations and analyses on the learning feature distribution, class activation maps (CAM), and feature maps to further evaluate our method.

Feature Distribution Visualization. To demonstrate the effective feature learning capabilities of our Face2Nodes, we visualized the learned features of Face2Nodes and a baseline model on the FERPlus test set using t-SNE [49], as shown in Fig. 7. We observed that the learned feature distribution of our Face2Nodes has a lower intra-class distance and a clearer inter-class boundary than the baseline model, reflecting our model's strong discriminative feature learning ability. In contrast, the feature distribution of the baseline model shows that *Anger* samples are more dispersed and mixed with samples of other expressions such as *Disgust* and *Happiness*. This makes it difficult for the baseline model to accurately classify samples of *Anger*, *Disgust*, and *Happiness*.

CAM Visualization. To demonstrate that our Face2Nodes model can effectively learn latent informative correlations among facial regions, we used Grad-CAM [50] to generate localization maps highlighting critical facial areas of original face images on the RAF-DB test set. As shown in Fig. 8, compared to the baseline model, our model exhibits larger attention areas and higher responses on key facial parts (*e.g.*, mouths), covering key facial areas for various expressions. For example, our Face2Nodes focuses on non-occluded salient facial parts (*i.e.*, mouth and cheeks) in the surprise expression activation map, while the baseline model tends to focus on occlusion regions. The activation maps on occlusion face images show that our method is also robust under in-the-wild scenarios. This visualization supports our claim that RG-Conv can effectively learn positive and important correlations among graph nodes (*i.e.*, facial parts), enabling our model to understand facial expressions from a structural graph relation perspective.

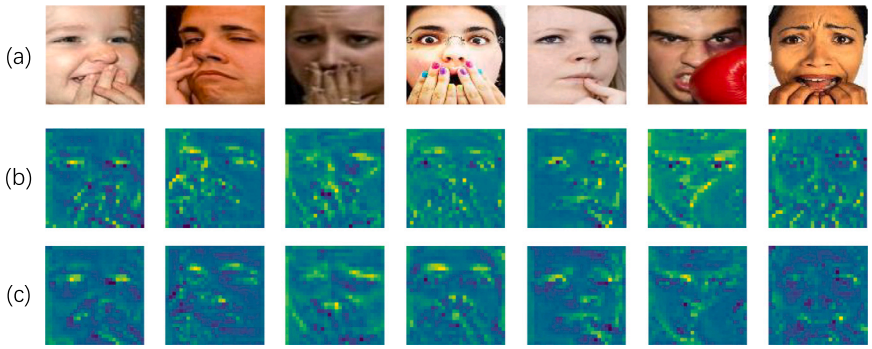

Fig. 9. (a) Raw facial images with different expressions from the test set of RAF-DB. (b) The feature maps extracted by the ViT-based method. (c) The feature maps extracted by our Face2Nodes. We observe that the ViT-based model focuses on more redundant occlusion areas unrelated to facial emotion (*i.e.*, hands), while our model focuses more on key facial parts associated with expression (*i.e.*, eyes and cheeks).

Feature Map Visualization. To demonstrate that correlation representations learned by the ViT-based model [10] are more susceptible to interference from facial emotion-irrelevant information than our Face2Nodes, we visualized feature maps extracted by both models in the same layer. As shown in Fig. 9, feature maps extracted by the ViT-based model are more likely to be interfered with by non-salient regions of facial expressions compared to our Face2Nodes. For example, in the first column, we can see that the ViT-based model focuses on more redundant occlusion areas unrelated to facial emotion (*i.e.*, hands), while our model focuses more on key facial parts associated with expression (*i.e.*, eyes and cheeks). It is important to note that the ViT-based model uses a self-attention mechanism to calculate the attention weights of all facial patches and updates each patch's features based on the weighted features of all other patches. As a result, expression-related patches may contain redundant information from expression-irrelevant patches. In contrast, Face2Nodes only uses k-nearest neighbor nodes (patches) to update a patch's features, minimizing interference from regions irrelevant to facial expressions. Furthermore, the ViT-based model exhibits larger attention regions and higher responses on marginal facial areas unrelated to expressions (*e.g.*, fifth face image). These results indicate that while the ViT-based method tends to learn redundant correlation representations of facial expressions, our Face2Nodes can learn more accurate facial expression correlation representations.

4.7. Limitations

Despite the impressive representation ability of our Face2Nodes for FER tasks, there are several limitations to our approach. Since the time complexity of dynamic graph construction based on the dilated *k-NN* algorithm is proportional to the number of graph nodes, the efficiency of the graph construction method needs further improvement during model training. Additionally, although our model outperforms state-of-the-art FER methods, it only achieves an accuracy of 66.69% on the AffectNet test set. This shows that the quality of real-world FER datasets still limits the recognition accuracy of our Face2Nodes.

5. Conclusion and future work

In this paper, we propose Face2Nodes, a novel graph-based architecture that overcomes the limitations of CNN-based and ViT-based methods in modeling structural correlations for facial expression recognition. To effectively leverage multi-level facial expression representations, we proposed MSF^{2}PE, a method for extracting both low-level and high-level feature embeddings of patches. This approach allows us to fully exploit the rich information contained in facial expressions at multiple levels. To address the limitations of conventional graph convolution in aggregating neighboring features, we introduce a new graph convolution operator, RG-Conv, to flexibly learn latent informative correlations among graph nodes. To the best of our knowledge, this is the first attempt to adapt Vision GNN to FER. We conducted extensive experiments on three popular benchmark datasets using both pre-training and from-scratch training approaches. Our state-of-the-art results and ablation studies demonstrate the effectiveness of our methods. Furthermore, we find that CNN-based FER methods have a larger performance gap between pre-training and training from scratch than our Face2Nodes, indicating that our model is more data-efficient than CNN-based approaches.

In future work, we plan to develop a more efficient algorithm for dynamic graph construction in RDGCN than the dilated *k-NN* method to reduce the computational cost of our model during training. We also aim to investigate more robust training strategies to mitigate the interference of noisy label samples during the training procedure of our Face2Nodes.

CRediT authorship contribution statement

Fan Jiang: Conceptualization, Methodology, Software, Writing – original draft. **Qionghao Huang:** Formal analysis, Writing – review & editing. **Xiaoyong Mei:** Investigation, Visualization. **Quanlong Guan:** Software, Validation. **Yaxin Tu:** Data curation. **Weiqi Luo:** Resources. **Changqin Huang:** Project administration, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (No. 2022ZD0117104), in part by the National Natural Science Foundation of China (No. 62337001), and the Key Research and Development Program of Zhejiang Province (No. 2022C03106), and the National Natural Science Foundation of China (No. 62207028), and the Open Research Fund of College of Teacher Education, Zhejiang Normal University (No. jykf22024).

References

[1] M. Yeasin, B. Bullot, R. Sharma, Recognition of facial expressions and measurement of levels of interest from video, IEEE Trans. Multimed. 8 (3) (2006) 500–508.
[2] A. Kaur, A. Mustafa, L. Mehta, A. Dhall, Prediction and localization of student engagement in the wild, in: 2018 Digital Image Computing: Techniques and Applications (DICTA), IEEE, 2018, pp. 1–8.
[3] C. Bisogni, A. Castiglione, S. Hossain, F. Narducci, S. Umer, Impact of deep learning approaches on facial expression recognition in healthcare industries, IEEE Trans. Ind. Inform. 18 (8) (2022) 5619–5627, https://doi.org/10.1109/TII.2022.3141400.
[4] H. Siqueira, S. Magg, S. Wermter, Efficient facial feature learning with wide ensemble-based convolutional neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, 2020, pp. 5800–5809.
[5] R. Mo, Y. Yan, J.-H. Xue, S. Chen, H. Wang, D^3Net: dual-branch disturbance disentangling network for facial expression recognition, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 779–787.
[6] Y. Li, J. Zeng, S. Shan, X. Chen, Patch-gated CNN for occlusion-aware facial expression recognition, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 2209–2214.
[7] K. Wang, X. Peng, J. Yang, D. Meng, Y. Qiao, Region attention networks for pose and occlusion robust facial expression recognition, IEEE Trans. Image Process. 29 (2020) 4057–4069.
[8] Z. Zhao, Q. Liu, S. Wang, Learning deep global multi-scale and local attention features for facial expression recognition in the wild, IEEE Trans. Image Process. 30 (2021) 6544–6556.
[9] X. Wu, J. He, C. Huang, Q. Huang, J. Zhu, X. Huang, H. Fujita, FER-CHC: facial expression recognition with cross-hierarchy contrast, Appl. Soft Comput. (2023) 110530.
[10] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: transformers for image recognition at scale, arXiv preprint, arXiv:2010.11929.
[11] Q. Huang, C. Huang, X. Wang, F. Jiang, Facial expression recognition with grid-wise attention and visual transformer, Inf. Sci. 580 (2021) 35–54.
[12] F. Xue, Q. Wang, G. Guo, TransFER: learning relation-aware facial expression representations with transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3601–3610.
[13] Y. Liu, X. Zhang, Y. Lin, H. Wang, Facial expression recognition via deep action units graph network based on psychological mechanism, IEEE Trans. Cogn. Dev. Syst. 12 (2) (2019) 311–322.
[14] R. Zhao, T. Liu, Z. Huang, D.P.-K. Lun, K.K. Lam, Geometry-aware facial expression recognition via attentive graph convolutional networks, IEEE Trans. Affect. Comput. 14 (2) (2023) 1159–1174.
[15] K. Han, Y. Wang, J. Guo, Y. Tang, E. Wu, Vision GNN: an image is worth graph of nodes, Adv. Neural Inf. Process. Syst. 35 (2022) 8291–8303.
[16] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, 2005, pp. 886–893.
[17] H. Soyel, H. Demirel, Localized discriminative scale invariant feature transform based facial expression recognition, Comput. Electr. Eng. 38 (5) (2012) 1299–1309.
[18] C. Shan, S. Gong, P.W. McOwan, Facial expression recognition based on local binary patterns: a comprehensive study, Image Vis. Comput. 27 (6) (2009) 803–816.
[19] H. Wang, J. Li, H. Wu, E. Hovy, Y. Sun, Pre-trained language models and their applications, Engineering (2022), https://doi.org/10.1016/j.eng.2022.04.024.
[20] D. Wang, M. Li, Stochastic configuration networks: fundamentals and algorithms, IEEE Trans. Cybern. 47 (10) (2017) 3466–3479.
[21] Y. Tang, Deep learning using support vector machines, arXiv preprint, arXiv:1306.0239.
[22] S. Li, W. Deng, J. Du, Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2852–2861.
[23] Y. Jiang, H. Lin, Y. Li, Y. Rong, H. Cheng, X. Huang, Exploiting node-feature bipartite graph in graph convolutional networks, Inf. Sci. 628 (2023) 409–423.
[24] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, arXiv preprint, arXiv:1710.10903.
[25] B. Yu, H. Xie, Z. Xu, PN-GCN: positive-negative graph convolution neural network in information system to classification, Inf. Sci. 632 (2023) 411–423.
[26] Y.G. Wang, M. Li, Z. Ma, G. Montufar, X. Zhuang, Y. Fan, Haar graph pooling, in: International Conference on Machine Learning, PMLR, 2020, pp. 9952–9962.
[27] C. Huang, F. Jiang, Q. Huang, X. Wang, Z. Han, W. Huang, Dual-graph attention convolution network for 3-d point cloud classification, IEEE Trans. Neural Netw. Learn. Syst. (2022) 1–13, https://doi.org/10.1109/TNNLS.2022.3162301.
[28] X. Zhang, C. Xu, X. Tian, D. Tao, Graph edge convolutional neural networks for skeleton-based action recognition, IEEE Trans. Neural Netw. Learn. Syst. 31 (8) (2019) 3047–3060.
[29] J. Zhou, X. Zhang, Y. Liu, X. Lan, Facial expression recognition using spatial-temporal semantic graph network, in: 2020 IEEE International Conference on Image Processing (ICIP), IEEE, 2020, pp. 1961–1965.
[30] R. Zhao, T. Liu, Z. Huang, D.P. Lun, K.-M. Lam, Spatial-temporal graphs plus transformers for geometry-guided facial expression recognition, IEEE Trans. Affect. Comput. (2022) 1–17, https://doi.org/10.1109/TAFFC.2022.3181736.
[31] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[32] D. Hendrycks, K. Gimpel, Gaussian error linear units (GELUs), arXiv preprint, arXiv:1606.08415.
[33] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, L. Zhang, CvT: introducing convolutions to vision transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 22–31.
[34] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117–2125.
[35] G. Li, M. Muller, A. Thabet, B. Ghanem, DeepGCNs: can GCNs go as deep as CNNs?, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9267–9276.
[36] E. Barsoum, C. Zhang, C.C. Ferrer, Z. Zhang, Training deep networks for facial expression recognition with crowd-sourced label distribution, in: Proceedings of the 18th ACM International Conference on Multimodal Interaction, 2016, pp. 279–283.
[37] I.J. Goodfellow, D. Erhan, P.L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D. Lee, et al., Challenges in representation learning: a report on three machine learning contests, in: International Conference on Neural Information Processing, Springer, 2013, pp. 117–124.
[38] A. Mollahosseini, B. Hasani, M.H. Mahoor, Affectnet: a database for facial expression, valence, and arousal computing in the wild, IEEE Trans. Affect. Comput. 10 (1) (2017) 18–31.
[39] K. Zhang, Z. Zhang, Z. Li, Y. Qiao, Joint face detection and alignment using multitask cascaded convolutional networks, IEEE Signal Process. Lett. 23 (10) (2016) 1499–1503.
[40] Y. Zhang, C. Wang, X. Ling, W. Deng, Learn from all: erasing attention consistency for noisy label facial expression recognition, in: European Conference on Computer Vision, Springer, 2022, pp. 418–434.
[41] Y. Guo, L. Zhang, Y. Hu, X. He, J. Gao, MS-Celeb-1M: a dataset and benchmark for large-scale face recognition, in: European Conference on Computer Vision, Springer, 2016, pp. 87–102.
[42] C. Huang, Combining convolutional neural networks for emotion recognition, in: 2017 IEEE MIT Undergraduate Research Technology Conference (URTC), IEEE, 2017, pp. 1–4.
[43] K. Wang, X. Peng, J. Yang, S. Lu, Y. Qiao, Suppressing uncertainties for large-scale facial expression recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6897–6906.
[44] H. Li, N. Wang, X. Ding, X. Yang, X. Gao, Adaptively learning facial expression representation via C-F labels and distillation, IEEE Trans. Image Process. 30 (2021) 2016–2028.
[45] D. Chen, G. Wen, H. Li, R. Chen, C. Li, Multi-relations aware network for in-the-wild facial expression recognition, IEEE Trans. Circuits Syst. Video Technol. 33 (8) (2023) 3848–3859, https://doi.org/10.1109/TCSVT.2023.3234312.
[46] Z. Zhang, X. Tian, Y. Zhang, K. Guo, X. Xu, Enhanced discriminative global-local feature learning with priority for facial expression recognition, Inf. Sci. 630 (2023) 370–384.
[47] X. Chen, X. Zheng, K. Sun, W. Liu, Y. Zhang, Self-supervised vision transformer-based few-shot learning for facial expression recognition, Inf. Sci. 634 (2023) 206–226.
[48] J. Deng, W. Dong, R. Socher, L. Li, K. Li, L. Fei-Fei, ImageNet: a large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.
[49] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.
[50] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.