Pattern Recognition 123 (2022) 108401

Contents lists available at ScienceDirect

Pattern Recognition

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

Co-attentive multi-task convolutional neural network for facial expression recognition

Wenmeng Yu[a,b], Hua Xu[a,b,*]

[a] State Key Laboratory of Intelligent Technology and Systems, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[b] Beijing National Research Center for Information Science and Technology(BNRist), Beijing 100084, China

ARTICLE INFO

Article history:
Received 30 June 2020
Revised 5 September 2021
Accepted 24 October 2021
Available online 26 October 2021

Keywords:
Facial expression recognition
Facial landmarks detection
Multi-task learning

ABSTRACT

Previous research on Facial Expression Recognition (FER) assisted by facial landmarks mainly focused on single-task learning or hard-parameter sharing based multi-task learning. However, soft-parameter sharing based methods have not been explored in this area. Therefore, this paper adopts Facial Landmark Detection (FLD) as the auxiliary task and explores new multi-task learning strategies for FER. First, three classical multi-task structures, including Hard-Parameter Sharing (HPS), Cross-Stitch Network (CSN), and Partially Shared Multi-task Convolutional Neural Network (PS-MCNN), are used to verify the advantages of multi-task learning for FER. Then, we propose a new end-to-end Co-attentive Multi-task Convolutional Neural Network (CMCNN), which is composed of the Channel Co-Attention Module (CCAM) and the Spatial Co-Attention Module (SCAM). Functionally, the CCAM generates the channel co-attention scores by capturing the inter-dependencies of different channels between FER and FLD tasks. The SCAM combines the max- and average-pooling operations to formulate the spatial co-attention scores. Finally, we conduct extensive experiments on four widely used benchmark facial expression databases, including RAF, SFEW2, CK+, and Oulu-CASIA. Extensive experimental results show that our approach achieves better performance than single-task and multi-task baselines, fully validating multi-task learning's effectiveness and generalizability[1].

1. Introduction

Facial Expression Recognition (FER) aims to recognize the emotional state of facial images. It plays an increasingly significant role in multiple application fields, including human-computer interaction (HCI), video transcriptions, and social communication. Prototypical facial expressions contain anger, disgust, fear, happiness, sadness, and surprise [1]. Differences between facial expressions are mainly dependent on the subtle distortion of facial muscles. Moreover, facial expression recognition is influenced by non-subjective factors, such as lighting and pose variations.

In the past few years, researchers have made considerable efforts to improve the performance of FER in lab-controlled and real-world environments. From traditional hand-crafted features (e.g., Gabor filter [2], Histogram of Gradients [3], and Local Binary Patterns [4]) to deep neural networks (e.g., Deep Convolutional Neural Network [5] and Generative Adversarial Network [6]), plenty of algorithms have been developed for mining expression-related intrinsic features and eliminating interference of external non-subjective factors. Ding et al. [7] adopted a two-stage training algorithm to fine-tune the face network and enhance expression recognition performance. Cai et al. [8] developed a new constraint function, island loss, to learn discriminative features among different expressions. Wang et al. [9] designed a region attention network and adopted ensemble learning to improve the generalizability and robustness, which achieved the state-of-the-art results on multiple public databases.

However, the above methods ignore the role of other facial attributes on FER. Intuitively, facial expressions are more directly related to how facial landmarks are distorted rather than the presence or absence of specific locations. Many previous works take the role of facial landmarks into account. First, facial landmarks are necessary for facial alignment, which is very helpful for reducing the data variance and improving model performance [10]. Second, allowing the model to perceive the location of facial landmarks can also improve the robustness. Therefore, Jung et al. [11], Zhang et al. [12] treated the landmarks trajectory as external in-

* Corresponding author at: State Key Laboratory of Intelligent Technology and Systems, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China.
E-mail addresses: ywm18@tsinghua.org.cn (W. Yu), xuhua@tsinghua.edu.cn (H. Xu).

[1] Codes and detailed instructions can be available at https://github.com/thuiar/cmcnn.

https://doi.org/10.1016/j.patcog.2021.108401
0031-3203/

puts and achieved better performance. Nevertheless, they need additional facial landmarks annotations during the inference stage, which causes the following three problems: (1) These methods are not end-to-end models during the inference stage. (2) The final performance largely depends on the quality of facial landmarks. (3) Facial landmarks are introduced as complementary features of facial expressions and are only integrated into the feature layer.

To solve the above problems, we introduce facial landmarks detection (FLD) as the auxiliary task and adopt multi-task learning strategies. Multi-task learning is to transfer knowledge from other relevant tasks and improve the generalizability of the main task [13]. Multi-task learning systems can be divided into two main categories: hard parameter sharing and soft parameter sharing. The former means that the underlying parameters of different tasks are fully shared, while the latter is only partially shared. Moreover, Pons and Masip [14] suggested that facial landmarks detection and facial action units [15] detection can effectively improve the recognition accuracy of FER. However, previous works on multi-task learning for FER are mainly focused on hard parameter sharing methods. In recent years, more and more soft parameter sharing strategies were developed while not used to FER. For example, Cross-Stitch Network (CSN) [16] and Partially Shared Multi-task Convolutional Neural Network (PS-MCNN) [17].

In this paper, we focus on soft parameter sharing strategies and propose an end-to-end Co-attentive Multi-task Convolutional Neural Network (CMCNN). First, for the convenience of multi-task learning, we adopt the same Deep Convolutional Neural Network (DCNN) backbone [18] for FLD and FER tasks. Then, we validate the effectiveness of multi-task learning for FER with three multi-task methods, including hard parameter sharing (HPS), CSN [16], and PS-MCNN [17]. Furthermore, inspired by CSN, we propose a new multi-task learning model, the CMCNN. Instead of direct sharing feature maps between two tasks, we design a Co-Attention (Co-ATT) module used in adjacent convolutional modules. The Co-ATT includes the Channel Co-Attentive Module (CCAM) and the Spatial Co-Attentive Module (SCAM). The CCAM aims to compute the co-attention scores by considering the correlation between different tasks, which denote the weight factors in different channels. The SCAM aims to compute the co-attention scores along the spatial axis, which denote the weight factors in different spatial areas. Motivated by Woo et al. [19], we combine the max- and average-pooling operations to obtain the spatial attention scores. Lastly, the outputs of CCAM and SCAM are integrated to generate inputs of the next layer. To fully demonstrate the advantage of our approach, we conduct plenty of experiments on two real-world facial expression databases, RAF [18] and SFEW2 [20], and two lab-controlled facial expression databases, CK+ [21] and OULU-CASIA [22]. Extensive experimental results show that our method is more effective than the single-task and multi-task baselines. Then, we validate the generalizability of our method by multiple sets of transfer validation experiments. Finally, we perform detailed comparative experiments to analyze the different roles of the CCAM and the SCAM.

In this work, we propose a co-attentive multi-task convolutional neural network. Our main contributions in this paper can be summarized as follows:

- Unlike previous hard parameter sharing methods, we introduce FLD as the auxiliary task of FER based on soft parameter sharing methods and validate the effectiveness of three different multi-task strategies, including HPS, CSN, and PS-MCNN.
- Based on the CSN, we propose an end-to-end co-attentive convolutional neural network that enables a more significant considerable facial area with the FLD subtask.
- We conduct detailed experiments to analyze all aspects of our method. Extensive experimental results validate the effectiveness and generalizability of our approach. Besides, we achieve a significant improvement compared with the single-task and multi-task baselines on all four benchmark databases.

The remainder of this paper is organized as follows. In Section 2, we review the literature related to FER, FLD, and multi-task learning. In Section 3, we make a detailed explanation for our approach. Section 4 describes our experimental settings. Section 5 shows our experimental results and detailed analysis. Finally, Section 6 provides the conclusion of this paper and discusses future work.

2. Related work

In this section, we briefly review the most related work in facial expression recognition, facial landmark detection, and multi-task learning.

2.1. Facial expression recognition

Recently, the DCNN has been a popular tool for facial expression recognition. Various standard DCNNs, such as VGG networks, Inception networks, and Residual networks, are adopted to extract deep facial representations and achieve better results on FER [7,23]. Ensemble of CNNs got the best-reported score in Emotiw2015 [24] sub challenge on image-based facial expression recognition [25,26]. Cai et al. [8] developed island loss, derived from center loss [27], to reduce the intra-class variations and increase the inter-class variations in the learned features. Furthermore, Li and Deng [18] designed a DCNN with a locality preserving loss layer to learn more discriminative features for the FER task. In this paper, our basic architecture for the task FER and the task FLD is derived from Li and Deng [18].

2.2. Facial landmark detection

FLD is a significant task for FER. Algorithms for FLD can be classified into three major categories: holistic methods, constrained local model methods, and regression-based methods [10]. In this paper, we mainly focus on regression-based methods. Similar to FER, CNN-based approaches, one of the regression-based methods, are also used widely in FLD [28,29] and achieved promising performance. Feng et al. [30] proposed a new loss function, the wing loss, to enhance the performance in minor errors. In this work, we introduce FLD as the auxiliary task to improve the performance of FER and use wing loss as our supervision function in the task FLD.

2.3. Multitask learning

Multi-task learning (MTL) attracts increasing attention in recent years. It aims to promote generalizability and robustness in single-task learning. Various computer vision tasks benefit from MTL, for example, semantic segmentation [31,32], action recognition [33,34], and etc. CNNs based MTL theory has also been greatly developed in recent years. Early methods [35,36] assumed that different tasks share the first several hidden layers and then have task-specific parameters in the subsequent layers, which is called hard parameter learning. However, it needs intensive experiments to find the optimal split point [17]. To solve this problem, Misra et al. [16] designed a cross-stitch network to capture all split-architectures, and Cao et al. [17] proposed a partially shared structure to share partial features in multiple tasks. However, the above methods neglect the significant difference in features' importance. Therefore, in this work, we propose the channel co-attentive module to re-weight the shared channel features and the spatial co-attentive module to pay more attention to the highlighted spatial areas.

Table 1
BaseDCNN architecture for single tasks, FLD and FER. Specifically, a batch normalization layer and a ReLU activation layer are added after each convolutional layer. "Conv": Convolution, "MP": MaxPooling Operation.

Layer	1	2	3	4	5	6	7	8	9
Type	Conv	MP	Conv	MP	Conv	Conv	MP	Conv	Conv
Kernel	3	2	3	2	3	3	2	3	3
Output	64	-	96	-	128	128	-	256	256
Stride	1	2	1	2	1	1	2	1	1
Pad	1	0	1	0	1	1	0	1	1

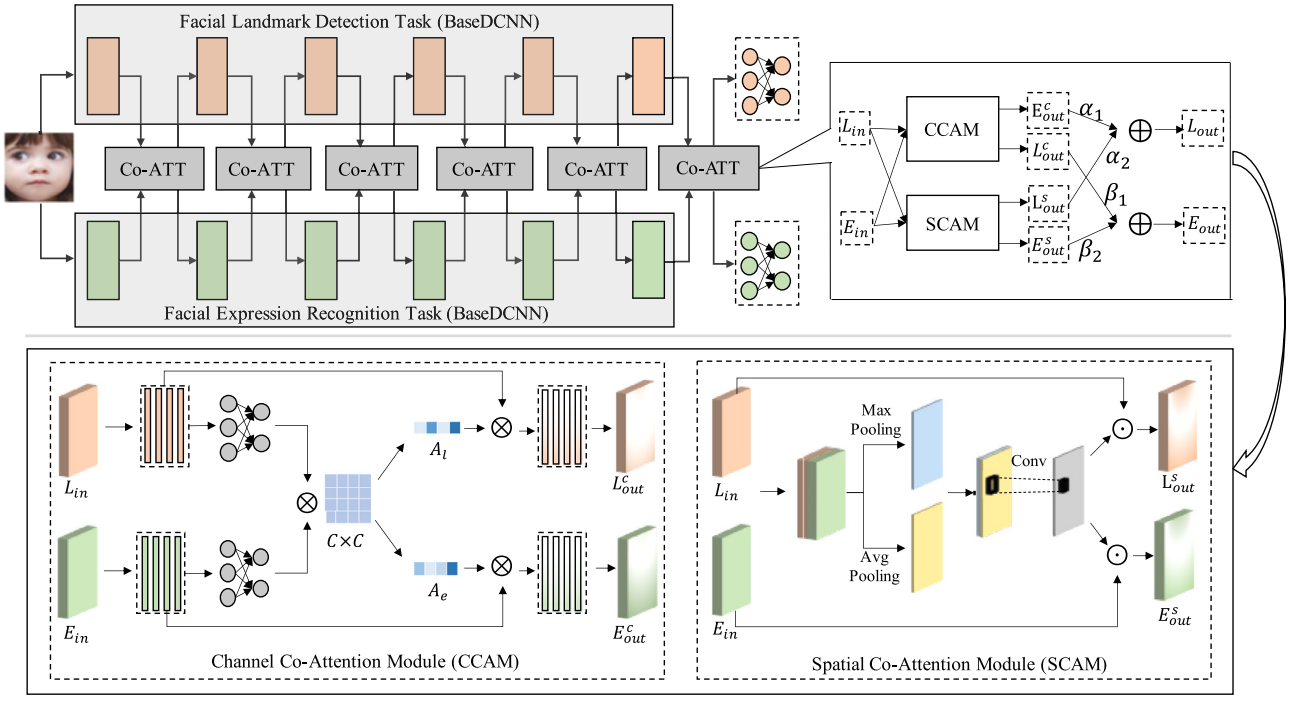

Fig. 1. Co-attentive multi-task convolutional neural network. The upper part is the overall architecture. The lower part is the detailed description of the CCAM (Left) and the SCAM (Right). L_{in} and E_{in} are outputs of the last convolutional component in FLD and FER tasks. L_{out} and E_{out} are inputs of the next convolutional component in FLD and FER tasks. α_1, α_2, β_1 and β_2 are four trainable parameters. C in the CCAM is the number of channels. $\oplus$, $\otimes$ and $\odot$ represent element plus, matrix multiply, and element multiply, respectively.

3. Methodology

In this paper, we propose an end-to-end co-attentive multi-task convolutional neural network. In this section, first, we introduce the overall architecture of our approach. Then, we explain the channel co-attention module and the spatial co-attention module in detail. Finally, we introduce the multi-task loss function used in our method.

3.1. Co-attentive multi-task convolutional neural network

For the scalability of multi-task strategies, we adopt two identical DCNNs to perform the FER task and the FLD task, respectively. The DCNN is composed of six convolutional components. Each component contains one convolutional layer, one batch normalization layer, one ReLU activation layer, and one max pooling layer (optional), as shown in Table 1. We refer to the outputs in the last convolutional component of FLD and FER tasks as L_{in} and E_{in} and the inputs in the next convolutional component of FLD and FER tasks as L_{out} and E_{out}, respectively.

The upper part in Fig. 1 is the overall architecture of CMCNN. Between two adjacent convolutional components, we propose a channel and spatial co-attentive module (Co-ATT). It takes L_{in} and E_{in} as inputs and takes L_{out} and E_{out} as outputs. The Co-ATT mainly contains two modules: the channel co-attention module (CCAM) and the spatial co-attention module (SCAM). They generate the corresponding outputs, L^c_{out} and E^c_{out}, L^s_{out} and E^s_{out}, respectively. The lower part is the detailed description of the CCAM (Left) and the SCAM (Right). We explain their details in the following two sections. Lastly, we generate the final outputs of Co-ATT by re-weighting the outputs of these two modules. It is worth noting that we exchange the outputs of the CCAM for the purpose of decreasing feature redundancy and enlarging feature sharing:

$$\begin{bmatrix} L_{out} & - \\ - & E_{out} \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \end{bmatrix} \times \begin{bmatrix} E^c_{out} & L^c_{out} \\ L^s_{out} & E^s_{out} \end{bmatrix} \tag{1}$$

where α_1, α_2, β_1 and, β_2 are trainable parameters.

After extracting facial features with the baseDCNN, we use fully connected layers as classifiers. In this paper, we adopt a 2-layer fully connected network in the task FLD and a 3-layer fully connected network for the task FER. Besides, batch normalization is used to avoid over-fitting. Except for the last layer, we use ReLU as the activation function.

3.2. Channel Co-attention module

Typically, the output of convolutional components contains multiple independent channel features, which make different contributions to final results. Therefore, it is essential to apply different weights for different channels before sharing features, which is

the aim of our CCAM. For the convenience in the following parts, we assume the C, H, and W represent the number of channels, height, and width of feature maps, respectively.

Firstly, given inputs $L_{in} \in R^{C\times H\times W}$ from the task FLD and $E_{in} \in R^{C\times H\times W}$ from the task FER, we use two independent two-layer fully connected networks to achieve the feature nonlinear transformation and avoid the overhead of high-dimensional calculations:

$$F_l = f(\hat{L_{in}}; W_l) \tag{2}$$

$$F_e = f(\hat{E_{in}}; W_e) \tag{3}$$

where $\hat{L_{in}} \in R^{C\times HW}$ and $\hat{E_{in}} \in R^{C\times HW}$ are the results of L_{in} and E_{in} flattening along channel axis, respectively. $f(\cdot)$ is implemented by MLP layers with trainable weights, W_l and W_e. As a result, F_l and $F_e \in R^{C\times D}$, $D = \delta \times H \times W$, $0 < \delta < 1$.

Then, inspired by Vaswani et al. [37], we generate the attention matrix by computing the similarity between different channel features:

$$S = \frac{F_e F_l^T}{\sqrt{D}} \in R^{C\times C} \tag{4}$$

where S_{ij} represents the attention weight between the ith channel in FER and jth channel in FLD.

We use softmax as the normalization function. Normalizing along the different axis will generate attention scores with different task preferences. Specifically, for a channel in task FER, we normalize S along the first C axis and consider the attention score with all channels in task FLD:

$$S^e_{ij} = \frac{exp(S_{ij})}{\sum_{k=1}^{C} exp(S_{kj})}; \qquad A_e = \sum_{j=1}^{C} S^e_{\bullet j} \tag{5}$$

where $S^e_{\bullet j} \in R^{C\times 1}$ is the jth column in matrix S.

Similarly, for a channel in the FLD task, we normalize S along the second C axis and consider the attention score with all channels in task FER:

$$S^l_{ij} = \frac{exp(S_{ij})}{\sum_{k=1}^{C} exp(S_{ik})}; \qquad A_l = \sum_{i=1}^{C} S^l_{i\bullet} \tag{6}$$

where $S^l_{i\bullet} \in R^{1\times C}$ is the ith row in matrix S.

Lastly, we apply channel attentions, A_e and A_l, into the corresponding original inputs and get outputs of CCAM:

$$E^c_{out} = A_e \odot E_{in} \tag{7}$$

$$L^c_{out} = A_l \odot L_{in} \tag{8}$$

where $\odot$ means element-size multiply operation. $E^c_{out} \in R^{C\times H\times W}$ and $L^c_{out} \in R^{C\times H\times W}$.

3.3. Spatial Co-attention module

Different from the CCAM, the SCAM focuses on the inter-spatial relationship of features. We use both max- and average-pooling operations along the channel axis to integrate the spatial information, which is shown to be effective in highlighting local detail information [19].

Firstly, we concatenate L_{in} and E_{in} along the channel axis and generate two feature maps with max- and average-pooling operations:

$$F_{\max} = MaxPool([L_{in}; E_{in}]) \tag{9}$$

$$F_{avg} = AvgPool([L_{in}; E_{in}]) \tag{10}$$

where $F_{\max}$ and $F_{avg} \in R^{1\times H\times W}$.

Then, we concatenate $F_{\max}$ and F_{avg} along the channel axis and obtain shared attention map using the standard convolutional transformation:

$$A_s = \sigma([F_{\max}; F_{avg}] * W_s) \tag{11}$$

where $*$ means the convolutional operation with kernel parameter $W_s \in R^{1\times 7\times 7}$. σ is the sigmoid activation function.

Lastly, we apply A_s into the L_{in} and E_{in} and get the outputs of SCAM:

$$E^s_{out} = A_s \odot E_{in} \tag{12}$$

$$L^s_{out} = A_s \odot L_{in} \tag{13}$$

where E^s_{out} and $L^s_{out} \in R^{C\times H\times W}$.

3.4. Multi-task loss

Formally, the FLD is a regression task, while the FER is a classification task. Therefore, we adopt different loss functions for them. For the task FLD, we use wing loss [30] as the supervision:

$$loss_L = \begin{cases} \omega \ln(1 + x/\epsilon) & \text{if } \ x < \omega; \\ x - M & \text{otherwise} \end{cases} \tag{14}$$

where $x = |y_l - \hat{y}_l|$, ω and ϵ are two super hypermeters. ω sets the range of the nonlinear part to $(-\omega, \omega)$ and ϵ limits the curvature of the nonlinear region. Following Feng et al. [30], we assume $\omega = 10$ and $\epsilon = 2$. $M = \omega - \omega \ln(1 + \omega/\epsilon)$ is a constant.

For the task FER, we use cross entropy as the supervision:

$$loss_E = -[y_e \log \hat{y}_e + (1 - y_e) \log(1 - \hat{y}_e)] \tag{15}$$

where y_e and $\hat{y}_e$ are the true values and predictive values, respectively.

Then, the total optimization objective of our approach is:

$$loss = loss_E + \lambda \bullet loss_L \tag{16}$$

where λ is a hyperparameter for controlling the weight of $loss_L$.

4. Experimental settings

To fully demonstrate the effectiveness of our method, we conduct extensive experiments on four benchmark expression databases, including two lab-controlled databases, CK+ [21,38] and Oulu-CASIA [22], and two real-world databases, RAF [18] and SFEW2 [20]. In addition to comparing different single-task and multi-task baselines, we perform three sets of transfer experiments to verify our method's generalizability and robustness in different data scenarios. Then, we show the performance of our method in three aspects, including feature visualization, time cost analysis, and parameter analysis. Lastly, we make a detailed ablation study to explore the contributions of different modules in our approach. For a better understanding, in this section, we give a detailed introduction to our experimental settings.

4.1. Benchmark databases

RAF. The Real-world Affective Faces Database (RAF) [18] is composed of 29,672 real-world facial images, which are collected from various search engines. In this database, annotations contain basic expressions and compound expressions. We only use the images with basic expression annotations. Finally, we use 12,771 images for training and 3068 images for validation.

SFEW2. Static Facial Expressions in the Wild (SFEW 2) [20] is the most widely used benchmark database for facial expression

Table 2
Facial expression and facial landmarks annotations statistics of RAF, SFEW2, CK+, and Oulu-CASIA. "NA" means vacancy.

#	RAF	SFEW2	CK+	Oulu-CASIA
Types of Expression	Happy Neutral Sad Disgust Fear Surprise Angry	Happy Neutral Sad Disgust Fear Surprise Angry	Happy Contempt Sad Disgust Fear Surprise Angry	Happy Sad Disgust Fear Surprise Angry
Number of Landmarks	5 or 37	NA	68	NA

recognition in the wild. It comprises 1766 images, including 958 for training, 436 for validation, and 372 for testing. Each image has been assigned to one of seven expression categories, including neutral and the six basic expressions. The expression labels of the training and validation sets are provided, while those of the testing set are held back by the challenge organizers. Therefore, we report the results on validation sets.

CK+. The Extended Cohn-Kanade Database (CK+) [21,38] contains 327 videos collected from 118 subjects. All these video sequences are from the neutral face to the peak expression. Following Cai et al. [8], the last three frames of each sequence are collected associated with the provided expression label. Therefore, 981 images (7 expressions, contempt and six basic expressions) are involved in our experiments.

Oulu-CASIA. The Oulu-CASIA database [22] contains 2880 videos, each of which contains one of the six basic expressions collected from 80 subjects. Following the previous work evaluated on the Oulu-CASIA database [8], only the 480 videos collected by the VIS system under normal indoor illumination are employed in our experiments. The same as CK+, for each video, the last three frames are collected as the peak frames of the labeled expression. Therefore, the Oulu-CASIA database contains 1440 images for our experiments.

4.2. Multi-task baselines

In this work, we compare our method with three classical multi-task baselines. For a fair comparison, we use the consistent convolutional neural network (shown in Table 1) for all multi-task baselines and our method.

HPS. Hard Parameter Sharing (HPS) is a relatively simple and intuitive multi-task strategy, which shares the bottom layer parameters and separates the top layer parameters. In this work, two tasks share the same base DCNN and are separated by different fully connected layers to achieve different learning goals.

CSN. Cross-Stitch Network (CSN) [16] is a soft sharing method, which uses cross-stitch units to capture all split architectures of different tasks. Different tasks have independent but the same network architecture. In this work, we add the cross-stitch unit between two adjacent network layers, including convolutional layers and fully connected layers.

PS-MCNN. Partially Shared Multi-Task Convolutional Neural Network (PS-MCNN) [17] is also a soft sharing method, which adopts an additional shared network to realize the interaction between different task-specific networks. The author of PS-MCNN proposed the LCLoss for encoding the geometric structure. However, due to the uncertainty of identity in our databases, we remove the LCLoss module.

4.3. Data preprocessing

Facial Landmark Annotations. Table 2 indicates that different databases have different landmark annotations, even not. We need to make a unified label for the landmarks in all databases. Because facial landmarks detection is not our primary goal, a mature detection tool can meet our needs. Finally, we choose the OpenFace2.0 toolbox [39] to obtain 68 facial landmarks for all databases. In our work, facial landmarks have two usages: aligning faces and supervising the FLD subtask.

Face Detection and Alignment. Facial alignment is an essential step in expression recognition. The RAF database provides aligned faces, but the other three databases do not. Therefore, for the SFEW2, the CK+, and the Oulu-CASIA databases, we use Multi-Task Cascade Convolutional Neural Networks (MTCNN) [40] to detect the face. All facial images are resized to 100×100. After successful face detection, we use three points constrained affine transformation for face alignment. Coordinates of the left eye, right eye and the midpoint of corners of the lips were used for alignment.

Data Augmentation. Because training data is relatively small, without using additional data, we use online data augmentation methods, including horizontal flipping and rotating to the left or right (between -10 and 10 degrees). These operations are applied to the training set only.

4.4. Experimental details

In this section, we introduce our experimental details, including training/testing strategy, training details, hypermeter selection, and evaluation metrics.

Training/Testing Strategy. All multi-task baselines and our method are trained and tested on static images. Following Cai et al. [8], a 10-fold cross-validation strategy is employed for the CK+, and Oulu-CASIA databases, where each database is further split into ten subsets, and the subjects in any two subsets are mutually exclusive. For each run, data from 8 sets are used for training, and the remaining two subsets are used for validation and testing, respectively.

Training Details. We train our model from scratch using Adam as the optimizer with an initial learning rate of 0.01. The decay strategy of the learning rate is decreased by ten times every ten epochs. Besides, we use L2 normalization for model weights, and the weight decay is 0.005. We terminate the training process when the model's performance on the validation set has not improved for eight epochs. Then we choose the best model to get the results on the test set.

Hypermeter Selection. For all multi-task learning baselines and our approach, we adjust their hyperparameters using the grid search strategy. Besides, for a fair comparison, we select five random seeds in each experiment, including 1, 12, 123, 1234, and 12345. And then, the average performance is reported.

Evaluation Metrics. For the FER task, due to the category imbalance in FER databases, we record our experimental results with three evaluation indicators, including total accuracy (ACC_{total}), average accuracy ($ACC_{average}$), and macro f1 (F1). In this work, we choose $ACC_{average}$ as our primary indicator. We compute them as the following:

$$ACC_{total} = \frac{1}{N}\sum_{i}^{N} I(y_i == \hat{y}_i) \tag{17}$$

$$ACC_{average} = recall_{macro} \tag{18}$$

$$F_1 = 2\frac{recall_{macro} \times precision_{macro}}{recall_{macro} + precision_{macro}} \tag{19}$$

$$precision_{macro} = \frac{\sum_{i=1}^{N_c} precision_{C_i}}{N_c}, recall_{macro} = \frac{\sum_{i=1}^{N_c} recall_{C_i}}{N_c} \tag{20}$$

Table 3
(%) Results for FER on RAF and SFEW2 with different multi-task learning methods. "BaseDCNN (FER)" and "BaseDCNN (FLD)" are the single-task baselines for FER and FLD tasks, respectively.

Method	RAF				SFEW2			
	ACC_{total}	$ACC_{average}$	F1	NRMSE	ACC_{total}	$ACC_{average}$	F1	NRMSE
BaseDCNN (FER)	82.73	73.19	74.83	-	32.98	30.81	29.82	-
BaseDCNN (FLD)	-	-	-	3.73	-	-	-	**22.3**
HPS	83.02	73.78	75.21	3.88	35.32	32.74	32.33	40.36
CSN	85.10	75.49	77.50	4.07	34.03	30.91	30.34	35.03
PS-MCNN	84.67	76.31	77.12	3.81	35.32	32.00	30.92	49.79
CMCNN	**85.22**	**77.03**	**77.97**	**3.71**	**37.95**	**34.95**	**34.39**	27.81

Table 4
(%) Results for FER on CK+ and Oulu-CASIA with different multi-task learning methods. "BaseDCNN (FER)" and "BaseDCNN (FLD)" are the single-task baselines for FER and FLD tasks, respectively.

Method	CK+				Oulu-CASIA			
	ACC_{total}	$ACC_{average}$	F1	NRMSE	ACC_{total}	$ACC_{average}$	F1	NRMSE
BaseDCNN (FER)	92.75	94.10	93.64	-	83.46	83.46	83.46	-
BaseDCNN (FLD)	-	-	-	**6.37**	-	-	-	**3.32**
HPS	94.31	93.05	92.21	29.35	80.74	80.74	80.64	10.12
CSN	95.59	94.37	93.74	23.82	83.54	83.54	83.46	9.61
PS-MCNN	96.16	94.92	94.42	48.73	83.49	83.49	83.41	8.19
CMCNN	**96.71**	**96.02**	**95.48**	14.87	**85.04**	**85.04**	**85.35**	4.64

$$\text{precision}_{C_i} = \frac{TP_{C_i}}{TP_{C_i} + FP_{C_i}}, \text{recall}_{C_i} = \frac{TP_{C_i}}{TP_{C_i} + FN_{C_i}} \tag{21}$$

where N is the amount of samples, N_c is the number of categories and C_i denotes the ith category. $I(\cdot)$ is the indicator function, where y_i and $\hat{y}_i$ represent the true value and the predictive value of the ith sample, respectively.

For the FLD task, we choose Normalized Root Mean Square Error (NRMSE) as the evaluation indicator. Different from the above indicators, NME is the smaller the better.

$$\text{NRMSE} = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{q}\sum_{j=1}^{q}\frac{\sqrt{(x_i^j - \hat{x}_i^j)^2 + (y_i^j - \hat{y}_i^j)^2}}{d} \tag{22}$$

where M is the number of samples. q is the number of landmarks. x_i^j and y_i^j are the predicts. $\hat{x}_i^j$ and $\hat{y}_i^j$ are the labels. d is the distance between the eyes.

5. Experimental result and analysis

5.1. Comparisons with multi-task methods

In this section, we compare CMCNN with three multi-task learning methods and the single-task baseline, BaseDCNN. Results are shown in Tables 3 and 4.

For the FER task, firstly, we compare the results between the multi-task methods and the single-task baseline. We can see that all multi-task methods acquire a better performance than the single-task baseline on four benchmark databases. It validates the feasibility and significance of introducing FLD as the auxiliary task of FER. It is worth noting that algorithm tools extract supervisions used during the training process. We think the performance will be further improved after using more precise landmark annotations. Secondly, we compare the results between our method and three multi-task baselines. The results show that our method achieves the best performance on all evaluation metrics. In particular, our method achieves a significant improvement than the "CSN", which has a similar architecture to ours. Therefore, our Co-ATT module is helpful to improve the performance of FER. In summarization, multi-task learning methods are helpful for FER, and our approach achieves a significant improvement.

For the FLD task, we find that the performance of multi-task methods is not as good as the single-task model in most databases. This situation is closely related to the databases' size because the FLD task has more outputs and needs more training samples. When the database's size is smaller, the FER task is easier to generate negative interference for the FLD task. Therefore, we can see that the performance of the FLD task in the multi-task methods is significantly decreased on the SFEW and the CK+, which is consistent with databases' size. Moreover, compared with multi-task baselines, our method achieves the closest or even better performance.

5.2. Comparisons with state-of-the-arts

In this section, we compare our experimental results with the other state-of-the-arts. Some state-of-the-arts adopted deeper CNN and were pre-trained on the larger face database. Therefore, for a fair comparison, following SCN [41] and RAN [9], we improve our CMCNN with the ResNet18 [42] pre-trained on the MS-Celeb-1M face recognition dataset. In the following sections, we call this new model CMCNN-ResNet18. Besides, we reproduce experimental results on ResNet18 as the new single-task baseline.

RAF. The left half of Table 5 shows the comparison between our method and the other state-of-the-arts on the RAF database. We compare our method with DLP-CNN [18], DSAN-RES-AGE [43], SCN [41], and RAN [9]. The DLP-CNN [18] has the same DCNN backbone as CMCNN. However, the CMCNN exceeds it by nearly 3%. For the ACC_{total}, RAN [9] got the best performance. However, our new method, the CMCNN-ResNet18, achieved 90.36% performance, which surpasses RAN by 2%. For the $ACC_{average}$, DSAN-RES-AGE [43] acquired the best performance currently. It adopted a two-stage training strategy and not an end-to-end model. However, our method achieves better results and is an end-to-end model. Moreover, CMCNN-ResNet18 acquires significant improvement than the ResNet18, which is in line with the comparison between the CMCNN and the BaseDCNN.

The right half of Table 5 is the confusion matrix of the CMCNN evaluated on the RAF test set. We can see that the results on "Ha" is the highest while the results on "Di" and "Fe" are significantly lower than the other categories, which is consistent with the distribution of data categories.

Table 5
(%) Comparisons with state-of-the-arts on the RAF database. The right half part is the confusion matrix of the CMCNN evaluated on the RAF testing set."Ha": happiness, "Ne": neutral, "Sa": sadness, "Di": disgust, "Fe": fear, "Su": surprise, "An": angry.

Method	ACC_{total}	$ACC_{average}$		Ha	Ne	Sa	Di	Fe	Su	An
DLP-CNN [18]	-	74.20	Ha	**93.1**	3.78	1.13	0.62	0.14	0.73	0.56
DSAN-RES-AGE [43]	-	76.40	Ne	3.41	**84.7**	6.65	2.12	0.0	2.71	0.47
SCN [41]	88.14	-	Sa	3.43	8.12	**83.8**	2.68	0.33	0.67	0.96
RAN [9]	88.55	-	Di	8.87	13.5	11.0	**54.4**	1.87	2.37	8.00
ResNet18* [42]	86.29	78.33	Fe	4.59	5.41	10.8	3.78	**61.1**	9.73	4.59
CMCNN	85.22	77.03	Su	2.74	5.11	2.67	0.73	2.92	**84.1**	1.7
CMCNN-ResNet18	90.36	82.26	An	5.06	4.81	2.22	5.56	1.73	2.47	**78.2**

Table 6
(%) Comparisons with state-of-the-arts on the SFEW2 database. The right half part is the confusion matrix of the CMCNN evaluated on the SFEW2 validation set."Ha": happiness, "Ne": neutral, "Sa": sadness, "Di": disgust, "Fe": fear, "Su": surprise, "An": angry.

Method	ACC_{total}	$ACC_{average}$		Ha	Ne	Sa	Di	Fe	Su	An
STM-ExpLet [45]	-	31.73	Ha	**59.7**	10.6	11.1	1.39	1.39	1.67	14.2
IL-VGG [8]	-	44.95	Ne	4.52	**49.5**	20.7	3.10	6.90	9.52	5.71
SFEW best [46]	-	52.50	Sa	7.32	22.54	**31.8**	5.35	6.76	16.9	9.30
SCN [41]	54.19	-	Di	8.18	20.9	26.4	**20.9**	9.10	1.80	12.7
ResNet18* [42]	42.37	39.01	Fe	6.98	13.5	14.4	6.51	**18.6**	26.5	13.5
CMCNN	37.95	34.39	Su	12.7	13.1	10.4	1.15	10.0	**38.5**	14.2
CMCNN-ResNet18	45.78	41.56	An	21.6	13.6	13.9	7.50	3.70	14.1	**25.6**

Table 7
(%) Comparisons with state-of-the-arts on the CK+ database. The right half part is the confusion matrix of the CMCNN evaluated on the CK+ testing set."Ha": happiness, "Co": contempt, "Sa": sadness, "Di": disgust, "Fe": fear, "Su": surprise, "An": angry.

Method	ACC_{total}	$ACC_{average}$		Ha	Co	Sa	Di	Fe	Su	An
MSR [47]	-	91.40	Ha	**99.4**	0.3	0.0	0.0	0.3	0.0	0.0
F-Bases [48]	-	94.81	Co	0.0	**91.9**	6.7	0.7	0.4	0.4	0.0
IL-VGG [8]	-	91.64	Sa	0.0	2.5	**93.1**	0.0	0.0	0.0	3.3
IL-CNN [8]	-	94.39	Di	0.3	0.3	0.0	**97.6**	0.0	0.3	1.4
ResNet18* [42]	96.93	96.54	Fe	0.0	0.0	0.0	0.0	**96.0**	4.0	0.0
CMCNN	96.71	96.02	Su	0.0	1.4	0.2	0.2	0.1	**98.2**	0.0
CMCNN-ResNet18	98.33	97.52	An	0.0	1.0	2.8	1.2	0.0	0.0	**94.9**

SFEW2. Table 6 shows the comparison result and the confusion matrix on the SFEW2 database. We compare our method with AUDN [44], STM-ExpLet [45], IL-VGG [8], and SFEW best [46]. Similar to RAF, SFEW2 is also a real-world database but with small samples. Moreover, the SFEW2 is a competition database. Its performance has been greatly improved by using multiple competition strategies, including additional training data, multiple networks fusion, and carefully adjusting the parameters. In our method, we do not use the above strategies. The result in Table 6 shows that our method outperforms the single-task baselines and achieves comparable results with stat-of-the-arts.

CK+. On the CK+ database, we compare our method with MSR [47], F-Bases [48], IL-VGG [8], and IL-CNN [8]. The comparison result and the confusion matrix are shown in Table 7. We can see that our approach surpasses the other state-of-the-arts. The size of CK+ is close to the SFEW2 database but without head pose and lighting variations.

Oulu-CASIA. Table 8 shows the comparison result and the confusion matrix on the Oulu-CASIA database. We compare our approach with FN2EN [7], STM-ExpLet [45], IL-CNN [8], and IL-VGG [8]. In particular, the Oulu-CASIA is a category balanced database, so ACC_{total} is equal to $ACC_{average}$. We can see that our approach achieves comparable performance on the Oulu-CASIA database. It is worth noting that the SOTA method, FN2EN Ding et al. [7], used additional face recognition databases and adopted a two-stage training strategy, not an end-to-end network. Therefore, our method achieves comparable performance and is easier to train on the same data scale.

5.3. Transfer validation

In order to fully validate the generalizability of our method, we make three sets of transfer validation experiments between real-world and lab-controlled databases, including "Real & Lab", "Real & Real", and "Lab & Lab". In each set of experiments, we conduct mutual validation experiments. For example, in the experiments of "Real & Lab", we use RAF as the training set and CK+ as the testing set. And then, we use CK+ as the training set and RAF as the testing set. We report the average performance of five times on the "Test Set". In particular, since different databases have different emotional categories, as shown in Table 2, we remove the "Neural" category in RAF and SFEW databases and the "Contempt" category in the CK+ database. Therefore, in this section, we conduct six classification experiments on all scenarios.

Results are shown in Table 9. Firstly, we can see that our method achieves significant improvements in all data scenarios, which indicates that our approach has better generalizability and learns more migratory features. Secondly, comparing the results in "Real & Lab", we find that the model trained by real-world databases has better transferability than the model trained by lab-controlled databases. Thirdly, results in the "Real & Real" show that transfer learning of real-world facial expression recognition is still challenging. Finally, comparing the results in "Lab & Lab", we can see that the result of "from Oulu to CK+" are significantly better than that of "from CK+ to Oulu". It is because that the Oulu has more samples and larger face variations.

Table 8
(%) Comparisons with state-of-the-arts on the Oulu-CASIA database. The right half part is the confusion matrix of the CMCNN evaluated on the Oulu-CASIA testing set."Ha": happiness, "Sa": sadness, "Di": disgust, "Fe": fear, "Su": surprise, "An": angry.

Method	ACC_{total}	$ACC_{average}$		Ha	Sa	Di	Fe	Su	An
FN2EN [7]	87.71	87.71	Ha	**92.4**	2.4	0.6	3.4	0.2	1.0
STM-ExpLet [45]	74.59	74.59	Sa	1.8	**80.2**	2.9	2.1	0.0	13.1
IL-CNN [8]	77.29	77.29	Di	1.2	2.2	**79.8**	2.9	0.9	13.1
IL-VGG [8]	84.58	84.58	Fe	6.5	3.2	2.3	**83.8**	1.7	2.6
ResNet18* [42]	84.92	84.92	Su	0.1	0.0	0.3	2.1	**96.7**	0.8
CMCNN	85.04	85.04	An	0.2	7.2	10.3	2.2	0.1	**79.8**
CMCNN-ResNet18	87.32	87.32							

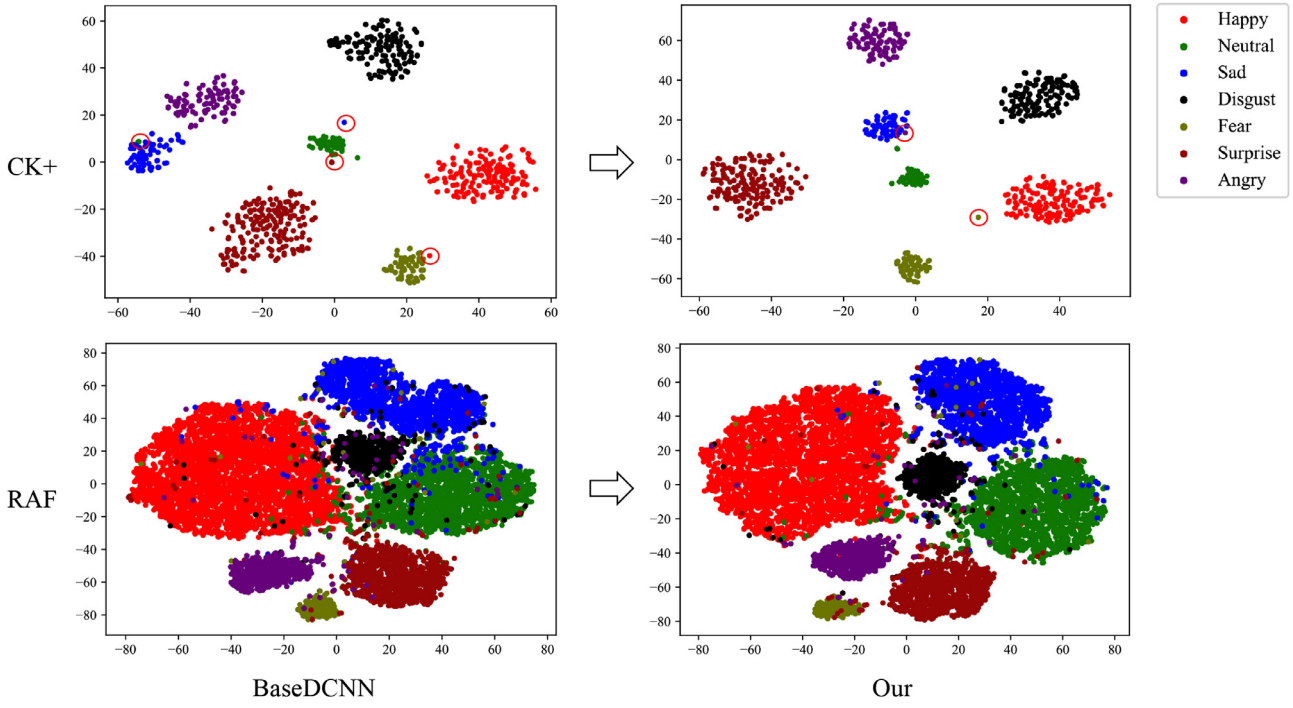

Fig. 2. T-SNE visualization of the outputs in the final hidden layer of the BaseDCNN and our approach. There are a total of 981 samples on the CK+ database. And there are a total of 15,339 samples, including 12,271 training data and 3068 testing data on the RAF database. Best viewed in color.

Table 9
(%) Transfer validation on different train sets and test sets. "Real" and "Lab" represent the characteristic of experimental data. RAF and SFEW2 are real-world databases. CK+ and Oulu ("Oulu-CASIA") are lab-controlled databases. Numbers below the name of database indicate the size of data set. "BaseDCNN" is the single-task model. All results are run in five times and we report the average values on the "Test Set".

Type	Train Set	Valid Set	Test Set	Model	$ACC_{average}$	F1
Real & Lab	RAF (12,771) CK+ (784)	RAF (3,068) CK+ (197)	CK+ (981) RAF (15,839)	BaseDCNN	56.90	54.63
				CMCNN	**63.72**	**64.16**
				BaseDCNN	27.10	24.00
				CMCNN	**29.86**	**26.68**
Real & Real	RAF (12,771) SFEW2 (958)	RAF (3,068) SFEW2 (436)	SFEW2 (1,394) RAF (15,839)	BaseDCNN	32.87	29.85
				CMCNN	**33.95**	**30.27**
				BaseDCNN	24.29	20.25
				CMCNN	**26.41**	**21.13**
Lab & Lab	CK+ (784) Oulu (1,152)	CK+ (197) Oulu (288)	Oulu (1,440) CK+ (981)	BaseDCNN	40.04	36.04
				CMCNN	**46.54**	**42.96**
				BaseDCNN	67.04	62.12
				CMCNN	**70.21**	**66.39**

5.4. Feature visualization

We use t-SNE [49] to perform a visualization study on the feature learned by the baseline and our approach. As illustrated in Fig. 2, the learned features are clustered according to 7 expressions. Results on CK+ can be viewed in the upper part, and results on RAF are in the lower part of Fig. 2. We can see that compared to the BaseDCNN, our method get more compact clusters, and the number of outliers is significantly reduced. It demonstrates that our method can learn more significant features, decreasing the intra-class features differences and increasing the variations of inter-class features.

5.5. Time cost analysis

In this section, we compare the time costs of different models in the inference stage, as shown in Table 10. All models are run on the single RTX2070 graphics card. The results show that the FLD task brings more time overhead than single-task baselines. However, our method's time cost is lower than RAN, which adopted the ensemble learning strategy and used multiple ResNet18. Therefore, although our model adds time cost, it brings a more significant performance improvement.

5.6. Parameter analysis

In this section, we aim to further explore the contributions of CCAM and SCAM using RAF database. In our multi-task learning approach, parameters $\beta = [\beta_1, \beta_2]$ are used to control the weights

Table 10
Time cost comparison between our method and baselines on the RAF database during the inference stage. Batch = 32.

Method	Time (ms) / Batch
SCN [41]	47.82
RAN [9]	134.19
CBAM [19]	42.67
BaseDCNN	25.43
ResNet18	38.17
CMCNN	42.18
CMCNN-ResNet18	73.53

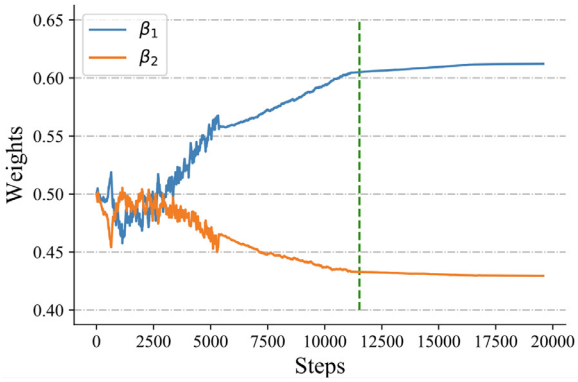

Fig. 3. The curve of the parameter β with the number of iterations. The green vertical dotted line is the moment when the model reaches the highest performance on the validation set. A step means a parameter update process. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

of CCAM and SCAM for the task FER, respectively. We assume that β_1 and β_2 are the mean of corresponding parameters in all Co-ATT components:

$$\beta_1 = \frac{1}{L}\sum_{k=1}^{L}\beta_1^k; \qquad \beta_2 = \frac{1}{L}\sum_{k=1}^{L}\beta_2^k; \tag{23}$$

where $L = 6$ is the number of Co-ATT components.

Firstly, we assume $\beta_1 = 0.5$ and $\beta_2 = 0.5$ as the initial values. As the number of network iterations increases, we observe the changes of β_1 and β_2, shown as Fig. 3. It indicates that the CCAM makes a greater contribution to our approach, which is consistent with previous experimental results.

Then, for a controlled analysis, we fix the parameters β_1 and β_2. In this way, we can explore the roles of the CCAM and the SCAM by setting different β_1 and β_2 values. The results are shown in Table 11. Comparing the results on different β_1 and β_2, we also find that the performance is higher when β_1 is relatively larger. Moreover, with the addition of the SCAM, the performance of our model is further improved.

Table 11
(%) Results for FER on RAF database with different module weights.

(β_1, β_2)	$ACC_{average}$	F1
(1.0, 0.0)	76.28	**77.68**
(0.0, 1.0)	75.83	76.55
(0.2, 0.8)	75.91	76.61
(0.8, 0.2)	**76.77**	77.25
(0.5, 0.5)	75.82	77.57

Table 12
(%) Results for channel attention module comparison on RAF database. CBAM-C and CMCNN-C mean the CBAM and CMCNN with channel attention only, respectively.

Model	ACC_{total}	$ACC_{average}$	F1
BaseDCNN	82.73	73.19	74.83
SE-Net [50]	83.95	75.43	76.11
CBAM-C [19]	84.15	75.59	76.37
CMCNN-C	**84.77**	**76.28**	**77.68**

5.7. Ablation study

To independently explore CCAM and SCAM modules' roles, we make an ablation study with different baselines.

CCAM. For the channel attention module, we compare our method (CMCNN-C) with SE-Net [50] and CBAM-C [19]. Results are presented in Table 12. The results show that our method achieves a better performance, which indicates that the co-attention between FER and FLD tasks is more effective than the self-attention of channels in the single task. **SCAM.** For the spatial attention module, we compare our method (CMCNN-S) with DAM-CNN [51] and CBAM-S [19]. Results are presented in Table 13. Specifically, CMCNN-C is derived from CBAM-S. From the results, CMCNN-C achieves better results than CBAM-C, which validates that the FLD task's introduction can help capture more effective spatial attention areas. In order to further explore what the SCAM learns, we use heating maps to visualize the attention area of the last convolutional components in the BaseDCNN and the last SCAM components in the CMCNN, as shown in Fig. 4. It is worth noting that the same test samples are used in both subfigures. The results show that the activation

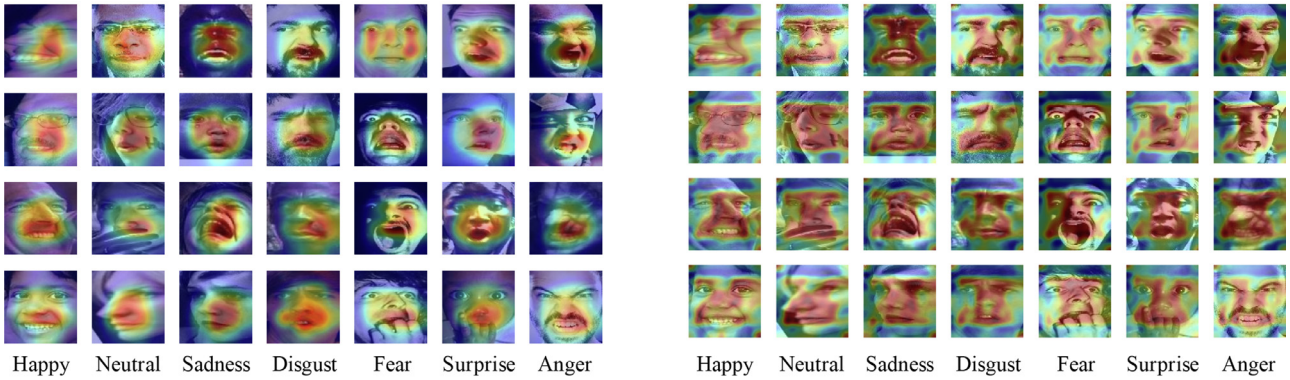

Fig. 4. Facial attention visualization comparison between baseline (BaseDCNN) and the SCAM of our proposed method (CMCNN). The true expressions in each column are the same. In each image, the redder, the greater the attention score. Conversely, the greener, the smaller the attention score.

Table 13
(%) Results for spatial attention module comparison on RAF database. CBAM-S and CMCNN-S mean the CBAM and CMCNN with spatial attention only, respectively.

Model	ACC_{total}	$ACC_{average}$	F1
BaseDCNN	82.73	73.19	74.83
DAM-CNN [51]	82.98	74.88	75.09
CBAM-S [19]	84.07	75.01	75.32
CMCNN-S	**84.35**	**75.83**	**76.55**

area obtained by BaseDCNN is mainly concentrated in the middle part of the face, and there is no significant difference in different face poses. However, the activation area obtained by our proposed method can focus on more effective parts, such as eyes, mouth, and other subtle distortions of the human face. It is consistent with our empirical observation. Moreover, the CMCNN can better cater to changes in head pose and hand occlusion. Thus, our approach is effective and robust in real-world facial expression recognition.

6. Conclusion

In this paper, we introduce facial landmarks detection as the auxiliary task of facial expression recognition. Inspired by the cross-stitch network, we propose a new end-to-end co-attentive multi-task convolutional neural network, mainly composed of the CCAM and the SCAM. Extensive experimental results on RAF, SFEW2, CK+, and Oulu-CASIA databases demonstrate the effectiveness and robustness of our approach. Multiple sets of transfer validation experiments validate the transferability of our method. Furthermore, we make a detailed model analysis to explore the different roles of CCAM and SCAM. We find that the CCAM makes a more significant contribution to our approach. Moreover, with the introduction of the SCAM, our model's performance is further improved. Our method can bring new possibilities to facial expression recognition and be easily used in other tasks.

In future work, we will further explore the interactivity between face-related tasks, including facial landmarks detection, facial expression recognition, and facial action units detection. Moreover, we will also focus on the occluded facial problem and further improve the model's robustness.

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.

Acknowledgment

This paper is funded by Project (Grant No. 62173195) supported by National Natural Science Foundation of China.

References

[1] P. Ekman, Strong Evidence for Universals in Facial Expressions: A Reply to Russell'S Mistaken critique, American Psychological Association, 1994.
[2] C. Liu, H. Wechsler, Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition, IEEE Trans. Image Process. 11 (4) (2002) 467–476.
[3] 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, IEEE, 2005, pp. 886–893.
[4] 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.
[5] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.
[7] H. Ding, S.K. Zhou, R. Chellappa, Facenet2expnet: regularizing a deep face recognition net for expression recognition, in: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), IEEE, 2017, pp. 118–126.
[8] J. Cai, Z. Meng, A.S. Khan, Z. Li, J. O'Reilly, Y. Tong, Island loss for learning discriminative features in facial expression recognition, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), IEEE, 2018, pp. 302–309.
[9] 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.
[10] Y. Wu, Q. Ji, Facial landmark detection: a literature survey, Int. J. Comput. Vis. 127 (2) (2019) 115–142.
[11] H. Jung, S. Lee, J. Yim, S. Park, J. Kim, Joint fine-tuning in deep neural networks for facial expression recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2983–2991.
[12] K. Zhang, Y. Huang, Y. Du, L. Wang, Facial expression recognition based on deep evolutional spatial-temporal networks, IEEE Trans. Image Process. 26 (9) (2017) 4193–4203.
[13] Y. Zhang, Q. Yang, A survey on multi-task learning, (2017) arXiv preprint arXiv: 1707.08114
[14] G. Pons, D. Masip, Multi-task, multi-label and multi-domain learning with residual convolutional networks for emotion recognition, (2018) arXiv preprint arXiv:1802.06664
[15] R. Ekman, What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS), Oxford University Press, USA, 1997.
[16] I. Misra, A. Shrivastava, A. Gupta, M. Hebert, Cross-stitch networks for multi–task learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3994–4003.
[17] J. Cao, Y. Li, Z. Zhang, Partially shared multi-task convolutional neural network with local constraint for face attribute learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4290–4299.
[18] S. Li, W. Deng, Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition, IEEE Trans. Image Process. 28 (1) (2018) 356–370.
[19] S. Woo, J. Park, J.-Y. Lee, I. So Kweon, CBAM: convolutional block attention module, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3–19.
[20] A. Dhall, R. Goecke, S. Lucey, T. Gedeon, Static facial expressions in tough conditions: data, evaluation protocol and benchmark, 1st IEEE International Workshop on Benchmarking Facial Image Analysis Technologies BeFIT, ICCV2011, 2011.
[21] P. Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, I. Matthews, The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, IEEE, 2010, pp. 94–101.
[22] G. Zhao, X. Huang, M. Taini, S.Z. Li, M. Pietikälnen, Facial expression recognition from near-infrared videos, Image Vis. Comput. 29 (9) (2011) 607–619.
[23] S.A. Bargal, E. Barsoum, C.C. Ferrer, C. Zhang, Emotion recognition in the wild from videos using images, in: Proceedings of the 18th ACM International Conference on Multimodal Interaction, 2016, pp. 433–436.
[24] A. Dhall, O.V. Ramana Murthy, R. Goecke, J. Joshi, T. Gedeon, Video and image based emotion recognition challenges in the wild: EmotiW 2015, in: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, 2015, pp. 423–426.
[25] B.-K. Kim, H. Lee, J. Roh, S.-Y. Lee, Hierarchical committee of deep CNNs with exponentially-weighted decision fusion for static facial expression recognition, in: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, 2015, pp. 427–434.
[26] Z. Yu, C. Zhang, Image based static facial expression recognition with multiple deep network learning, in: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, 2015, pp. 435–442.
[27] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: European Conference on Computer Vision, Springer, 2016, pp. 499–515.
[28] Y. Sun, X. Wang, X. Tang, Deep convolutional network cascade for facial point detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3476–3483.
[29] A. Zadeh, Y. Chong Lim, T. Baltrusaitis, L.-P. Morency, Convolutional experts constrained local model for 3D facial landmark detection, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2017, pp. 2519–2528.
[30] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, X.-J. Wu, Wing loss for robust facial landmark localisation with convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2235–2245.
[31] F. Xia, P. Wang, X. Chen, A.L. Yuille, Joint multi-person pose estimation and semantic part segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6769–6778.
[32] D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2650–2658.

[33] Y. Yang, C. Deng, S. Gao, W. Liu, D. Tao, X. Gao, Discriminative multi-instance multitask learning for 3D action recognition, IEEE Trans. Multimedia 19 (3) (2016) 519–529.
[34] Y. Yang, C. Deng, D. Tao, S. Zhang, W. Liu, X. Gao, Latent max-margin multitask learning with skelets for 3-D action recognition, IEEE Trans. Cybern. 47 (2) (2016) 439–448.
[35] S. Li, Z.-Q. Liu, A.B. Chan, Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 482–489.
[36] W. Liu, T. Mei, Y. Zhang, C. Che, J. Luo, Multi-task deep visual-semantic embedding for video thumbnail selection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3707–3715.
[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[38] T. Kanade, J.F. Cohn, Y. Tian, Comprehensive database for facial expression analysis, in: Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580), IEEE, 2000, pp. 46–53.
[39] T. Baltrusaitis, A. Zadeh, Y.C. Lim, L.-P. Morency, OpenFace 2.0: facial behavior analysis toolkit, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), IEEE, 2018, pp. 59–66.
[40] 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.
[41] 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.
[42] 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.
[43] Y. Fan, V. Li, J.C.K. Lam, Facial expression recognition with deeply-supervised attention network, IEEE Trans. Affect. Comput. (2020).
[44] M. Liu, S. Li, S. Shan, X. Chen, Au-aware deep networks for facial expression recognition, in: 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, FG 2013, Shanghai, China, 22–26 April, 2013, IEEE Computer Society, 2013, pp. 1–6, doi:10.1109/FG.2013.6553734.
[45] M. Liu, S. Shan, R. Wang, X. Chen, Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23–28, 2014, IEEE Computer Society, 2014, pp. 1749–1756, doi:10.1109/CVPR.2014.226.
[46] B.-K. Kim, J. Roh, S.-Y. Dong, S.-Y. Lee, Hierarchical committee of deep convolutional neural networks for robust facial expression recognition, J. Multimodal User Interfaces 10 (2) (2016) 173–189, doi:10.1007/s12193-015-0209-0.
[47] R. Ptucha, G. Tsagkatakis, A. Savakis, Manifold based sparse representation for robust expression recognition without neutral subtraction, in: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), IEEE, 2011, pp. 2136–2143.
[48] E. Sariyanidi, H. Gunes, A. Cavallaro, Learning bases of activity for facial expression recognition, IEEE Trans. Image Process. 26 (4) (2017) 1965–1978.
[49] M.L. van der, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (Nov) (2008) 2579–2605.
[50] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[51] S. Xie, H. Hu, Y. Wu, Deep multi-path convolutional neural network joint with salient region attention for facial expression recognition, Pattern Recognit. 92 (2019) 177–191.

Wenmeng Yu received his B.S. from Huazhong University of Science and Technology in 2018. Now he is a postgraduate at the State Key Laboratory of Intelligent Technology and Systems, in Dept. of C.S., Tsinghua University. His research fields include the following aspects: Facial Expression Recognition, Multi-Task Learning, Multi-Modal Learning.

Prof. & Dr. Hua Xu received his B.S. from Xi'an Jiaotong University in 1998. He received his M.S. and P.H.D from Tsinghua University in 2000 and 2003. Now he is a tenured associate professor in Dept. of C.S., Tsinghua University. His research fields include the following aspects: Data Mining, Dialogue System, Multi-modal Learning, and Evolutionary Computation. He has published over 100 academic papers, received 20 invention patents of China and is also the copyright owner of 16 software systems. He has achieved the 2nd Prize of National Science and Technology Progress of China and the 1st Prize of Beijing Science and Technology.