Cross-Domain Facial Expression Recognition: A Unified Evaluation Benchmark and Adversarial Graph Learning

Tianshui Chen, Tao Pu, Hefeng Wu, Yuan Xie, Lingbo Liu, Liang Lin

Abstract—Facial expression recognition (FER) has received significant attention in the past decade with witnessed progress, but data inconsistencies among different FER datasets greatly hinder the generalization ability of the models learned on one dataset to another. Recently, a series of cross-domain FER algorithms (CD-FERs) have been extensively developed to address this issue. Although each declares to achieve superior performance, comprehensive and fair comparisons are lacking due to inconsistent choices of the source/target datasets and feature extractors. In this work, we first propose to construct a unified CD-FER evaluation benchmark, in which we re-implement the well-performing CD-FER and recently published general domain adaptation algorithms and ensure that all these algorithms adopt the same source/target datasets and feature extractors for fair CD-FER evaluations. Based on the analysis, we find that most of the current state-of-the-art algorithms use adversarial learning mechanisms that aim to learn holistic domain-invariant features to mitigate domain shifts. However, these algorithms ignore local features, which are more transferable across different datasets and carry more detailed content for fine-grained adaptation. Therefore, we develop a novel adversarial graph representation adaptation (AGRA) framework that integrates graph representation propagation with adversarial learning to realize effective cross-domain holistic-local feature co-adaptation. Specifically, our framework first builds two graphs to correlate holistic and local regions within each domain and across different domains, respectively. Then, it extracts holistic-local features from the input image and uses learnable per-class statistical distributions to initialize the corresponding graph nodes. Finally, two stacked graph convolution networks (GCNs) are adopted to propagate holistic-local features within each domain to explore their interaction and across different domains for holistic-local feature co-adaptation. In this way, the AGRA framework can adaptively learn fine-grained domain-invariant features and thus facilitate cross-domain expression recognition. We conduct extensive and fair comparisons on the unified evaluation benchmark and show that the proposed AGRA framework outperforms previous state-of-the-art methods.

Index Terms—Facial expression recognition, Domain adaptation, Graph representation learning, Adversarial learning, Fair evaluation

1 Introduction

Automatically recognizing facial expressions helps understand human emotion states and behaviors, benefiting a wide range of applications such as human-computer interactions [1], medicine [2], and security monitoring [3], [4]. Over the last decade, much effort has been dedicated to collecting various facial expression recognition (FER) datasets, namely, lab-controlled datasets (e.g., the Extended Cohn-Kanade (CK+) [5], Japanese Female Facial Expressions (JAFFE) [6], MMI [7], and Oulu-CASIA [8] datasets) and in-the-wild datasets (e.g., the Real-world Affective Faces (RAF) [9], [10], Static Facial Expressions in the Wild (SFEW2.0) [11], Expression in-the-Wild (ExpW) [12], SEWA DB [13], and Facial Expression Recognition 2013 (FER2013) [14] datasets), which greatly facilitates FER performance. However, because humans' understanding of facial expressions varies with their experiences and cultures, their annotations are inevitably subjective, leading to obvious domain shifts across different datasets [15], [16] (see Figure 3). In addition, facial images of different datasets are usually collected in different environments (lab-controlled or in-the-wild environments) and from humans of different races, further enlarging the domain shift [17]. Consequently, current best-performing methods may achieve satisfactory performance in intra-dataset protocols, but they suffer from dramatic performance deterioration in inter-dataset settings [15].

Recently, much effort has been dedicated to the cross-domain FER (CD-FER) task by learning transferable features. Although each newly proposed method claims to achieve superior performance, it is difficult to assess the actual improvement of each method due to the inconsistent choice of the source/target datasets and feature extractors (see Table 1). On the one hand, the source datasets carry basic supervision for feature extractor and classifier learning, and the distribution similarity between the source and target datasets plays a key role in the final performance. Take the ICID algorithm [24] in Table 1 for example, using different source datasets leads to a performance disparity of more than 8.0% on the CK+ dataset. On the other hand, features extracted using different backbone networks in-

- *Tianshui Chen is with The Guangdong University of Technology, Guangzhou, China. (E-mail: tianshuichen@gmail.com)*
- *Tao Pu, Hefeng Wu, Yuan Xie, and Liang Lin are with Sun Yat-Sen University, Guangzhou, China. (Email: putao3@mail2.sysu.edu.cn, wuhefeng@mail.sysu.edu.cn, phoenixsysu@gmail.com, linliang@ieee.org)*
- *Lingbo Liu is with The Hong Kong Polytechnic University. (Email: lingbo.liu@polyu.edu.hk)*
- *Tianshui Chen and Tao Pu contribute equally to this paper and share first authorship. Corresponding author: Hefeng Wu*

- *This work was supported in part by National Natural Science Foundation of China (NSFC) under Grant No. 61876045, 61836012, and 62002069, in part by the Natural Science Foundation of Guangdong Province under Grant No. 2017A030312006, and in part by Guangdong Provincial Basic Research Program under Grant No. 102020369.*

Method	Source set	Backbone	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Da et al. [18]	BOSPHORUS	HOG & Gabor filters	57.60	36.2	-	-	-	-
E3DCNN [19]	MMI&FERA&DISFA	Inception-ResNet	67.52	-	-	-	-	-
STCNN [20]	MMI&FERA	Inception-ResNet	73.91	-	-	-	-	-
GDFER [21]	Six datasets	Inception	64.20	-	39.80	34.00	-	-
AIDN [22]	CK+	Manually designed network	-	-	29.43	-	-	-
DFA [23]	CK+	Manually designed network	-	63.38	-	-	-	-
DETN [17]	RAF-DB	Manually designed network	78.83	57.75	47.55	52.37	-	-
ICID [24]	RAF-DB	DarkNet-19	84.50	-	-	-	-	-
ICID [24]	MMI	DarkNet-19	76.10	-	-	-	-	-
FTDNN [25]	Six datasets	VGGNet	88.58	44.32	-	-	-	-
ECAN [16]	RAF-DB 2.0	VGGNet	86.49	61.94	54.34	58.21	-	-
DT	RAF-DB	ResNet-50	71.32	50.23	50.46	54.49	67.45	58.79
PLFT	RAF-DB	ResNet-50	77.52	53.99	48.62	56.46	69.81	61.28
DFA [23]	RAF-DB	ResNet-50	64.26	44.44	43.07	45.79	56.86	50.88
DETN [17]	RAF-DB	ResNet-50	78.22	55.89	49.40	52.29	47.58	56.68
ICID [24]	RAF-DB	ResNet-50	74.42	50.70	48.85	53.70	69.54	59.44
FTDNN [25]	RAF-DB	ResNet-50	79.07	52.11	47.48	55.98	67.72	60.47
ECAN [16]	RAF-DB	ResNet-50	79.77	57.28	52.29	56.46	47.37	58.63
LPL [9]	RAF-DB	ResNet-50	74.42	53.05	48.85	55.89	66.90	59.82
CADA [26]	RAF-DB	ResNet-50	72.09	52.11	53.44	57.61	63.15	59.68
SAFN [27]	RAF-DB	ResNet-50	75.97	61.03	52.98	55.64	64.91	62.11
SWD [28]	RAF-DB	ResNet-50	75.19	54.93	52.06	55.84	68.35	61.27
JUMBOT [29]	RAF-DB	ResNet-50	79.46	54.13	51.97	53.56	63.69	60.56
ETD [30]	RAF-DB	ResNet-50	75.16	51.19	52.77	50.41	67.82	59.47
Ours	RAF-DB	ResNet-50	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 1
Evaluation settings and accuracies of current leading CD-FER methods on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets. The settings and results in the upper part are taken from the corresponding papers, and they generally use different source datasets and backbones for comparison. The results of the bottom part are generated by our implementation with ResNet-50 as the backbone and the RAF-DB dataset as the source dataset. Reference [21] selects one dataset (i.e., CK+, SFEW2.0 or FER2013) from the seven datasets CK+, MultiPIE, MMI, DISFA, FERA, SFEW2.0, and FER2013 as the target domain and uses the rest six as the source domain; reference [25] selects one dataset (i.e., CK+ or JAFFE) from the seven datasets CK+, JAFFE, MMI, RaFD, KDEF, BU3DFE and ARFace as the target domain and uses the rest six as the source domain. "-" denotes the corresponding result is not provided.

herently have different discrimination and generalization abilities. These inconsistent choices hinder fair comparisons among CD-FER algorithms.

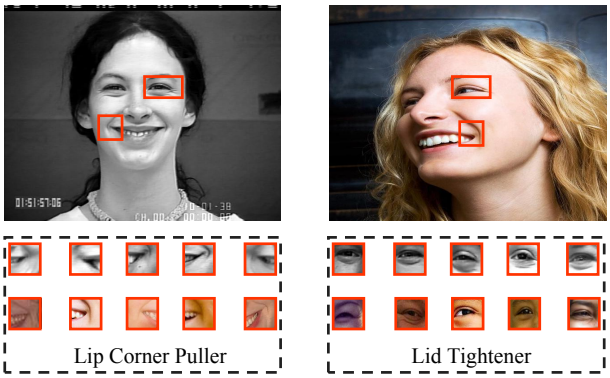


Fig. 1. The upper part presents two examples from CK+ [5] and RAF-DB [10], and the lower part presents the local mouth-corner and eye regions of the images from CK+ and RAF-DB.

Therefore, our first goal of this work is to construct a unified CD-FER evaluation benchmark for fair comparison and promote research in this field. To this end, we first analyze the performance gap caused by these inconsistent choices and re-implement several of the best-performing options [9], [16], [17], [23], [24], [25] according to the corresponding papers. In addition, many general domain adaptation methods exist, and we apply some of the best-performing ones [26], [27], [28] to CD-FER. To ensure fair comparisons, we use the same backbone network for feature extraction and the same source/target datasets for all the algorithms. For example, the lower part of Table 1 reports the results that are obtained by using the same RAF-DB [10] source dataset and ResNet-50 backbone, and it can better show the actual performance comparisons among different algorithms. On the other hand, current FER datasets mainly feature Western individuals. To facilitate more comprehensive evaluations, we further build a large-scale FER dataset (namely, the Asian Face Expression (AFE) dataset) that contains 54,901 well-labeled samples captured from Asian individuals. This dataset is used as the source dataset to evaluate the cross-culture FER performance.

Our second goal of this work is to propose a new adversarial learning framework that realizes effective cross-domain holistic-local feature co-adaptation for CD-FER, motivated by the following observations. To address the CD-FER task, most of the current state-of-the-art algorithms adopt adversarial learning mechanisms [26], [31], [32], [33], [34], [35] to learn domain-invariant features, but they just focus on extracting holistic features for domain adaptation. We believe that incorporating local features can benefit CD-FER because they are more transferable across different datasets and carry more detailed content for fine-grained adaptation. As exhibited in Figure 1, the lip-corner-puller and lid tightener actions can be used to distinguish happy expressions, and they are similar for samples from different datasets. To give a more in-depth analysis, we conduct an experiment on the generalization abilities of using holistic, local, and holistic-local features (see Sec. 4.1 for details). As shown in Figure 5, using the local or holistic-local features can greatly reduce the distribution discrepancies between learned features of the source and target datasets compared

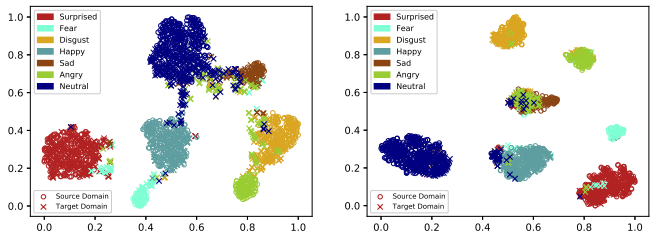


Fig. 2. Illustration of a feature distribution learned by the baseline adversarial learning [31] method that merely uses holistic features (left) and our proposed AGRA framework (right). It is obvious that the AGRA framework can better gather the samples of the same category and from different domains together than the baseline method, suggesting that our framework can learn more discriminative domain-invariant features for CD-FER.

with using the holistic features. Thus, it is worthy to model the correlation of holistic-local features for addressing CD-FER.

In this work, we show that the correlation of holistic-local features within each domain and across the source and target domains can be explicitly represented by structured graphs, and their interplay and adaptation can be captured by adaptive message propagation through the graphs. To achieve the aforementioned goal, we develop a novel adversarial graph representation adaptation (AGRA) framework that integrates graph representation propagation with adversarial learning for cross-domain holistic-local feature interplay and co-adaptation. Specifically, we first extract several discriminative local regions based on facial landmarks (e.g., eyes, nose, and mouth corner) [36], [37] and build two graphs to correlate holistic images and local regions within each domain and across different domains, respectively. Given an input image from one domain, we extract features of the holistic image and the local regions to initialize the corresponding nodes of this domain. The nodes of the other domain are initialized by the corresponding per-class learnable statistical feature distributions. Then, we introduce two stacked graph convolutional networks (GCNs) to propagate node messages within each domain to explore holistic-local feature interactions and across the two different domains to enable holistic-local feature co-adaptation. In this way, our method can progressively mitigate the shift in the holistic-local features between the source and target domains, enabling learning discriminative and domain-invariant features to facilitate CD-FER. Figure 2 shows the feature distributions learned by the baseline adversarial learning method [31] that merely uses holistic features for domain adaptation, and our proposed AGRA framework. The figure shows that our framework can better gather the features of samples that belong to the same category and are taken from different domains together than the baseline method. This phenomenon suggests that our framework can better learn domain-invariant features while improving their discriminative ability.

A preliminary version of this work was presented as a conference paper [38]. In this version, we strengthen the work from four aspects. First, we construct a unified and comprehensive CD-FER evaluation benchmark by re-implementing more best-performing algorithms and providing a variety of unified evaluation protocols (i.e., using the same source/target datasets and feature extractors) for comparing these algorithms fairly. We follow the evaluation benchmark to conduct extensive performance analysis and comparison under various evaluation protocols (i.e., different combination of source/target datasets and feature extractors). This is the first attempt to construct such a benchmark and it can doubtlessly facilitate the progress of CD-FER. Second, we build a new large-scale FER dataset that contains tens of thousands of samples captured mainly from Asian individuals. This dataset is incorporated into our benchmark and can be used to compare the cross-culture FER performance. Third, the motivation and details of the proposed AGRA framework are better described and enriched. Finally, substantially more experiments are conducted to demonstrate the effectiveness of the proposed framework and verify the contribution of each component.

The contributions of this work can be summarized as follows: 1) We construct a fair and comprehensive CD-FER evaluation benchmark by unifying the source/target datasets and feature extractors for different well-performing algorithms. To the best of our knowledge, this is the first attempt to construct such a unified evaluation benchmark. 2) We construct a large-scale FER dataset that contains 54,901 well-labeled samples mainly captured from Asian individuals. This dataset can be used to promote cross-culture FER task. 3) We propose to integrate graph representation propagation with the adversarial learning mechanism for holistic-local feature co-adaptation across different domains. This method can learn fine-grained and domain-invariant features to improve CD-FER performance. 4) We develop a class-aware two-stage updating mechanism to iteratively learn the statistical feature distribution of each domain for graph node initialization. This mechanism is a key factor in mitigating domain shifting to facilitate learning domain-invariant features. 5) We conduct comprehensive experiments using the unified evaluation benchmark to compare the leading CD-FER algorithms fairly and verify the effectiveness of the proposed framework. When using RAF-DB as the source dataset and ResNet-50 as the backbone, the AGRA framework improves the accuracy averaging over the CK+ [5], JAFFE [6], FER2013 [14], SFEW2.0 [11], and ExpW [39] datasets by 4.02% compared with the previously identified best performers. The unified evaluation benchmark, including the implementation codes of our proposed AGRA framework and current leading methods, the trained models and the newly built AFE dataset, is available at **https://github.com/HCPLab-SYSU/CD-FER-Benchmark.**

The remainder of this work is organized as follows. We review the most related works in Sec. 2. Sec. 3 presents the unified evaluation benchmark, and Sec. 4 introduces the proposed AGRA framework in detail. We provide extensive experimental comparison and evaluation in Sec. 5 and conclude the work in Sec. 6.

2 Related Works

In this section, we mainly review three streams of related works: cross-domain FER, adversarial domain adaptation, and graph representation learning.

2.1 Cross-Domain Facial Expression Recognition

Due to the subjective annotation process and inconsistent collection conditions, distribution divergences commonly exist among different FER datasets. To maintain the performance of cross-dataset validation, many CD-FER algorithms have been proposed [16], [23], [40], [41], [42], [43], [44], [45], [46], [47]. For instance, Yan et al. [42] used subspace learning to transfer the knowledge extracted from the source dataset to the target dataset. However, this method still requires annotating some samples from the target dataset, which is unexpected in unsupervised CD-FER scenarios. In contrast, Zheng et al. [44] proposed combining the labeled samples from the source domain and unlabeled auxiliary data from the target domain to jointly learn a discriminative subspace. This algorithm does not require any annotated samples from the target domain and thus facilitates CD-FER in an unsupervised manner. In contrast, Zong et al. [46] generated additional samples that share the same or similar feature distribution for both the source and target domains. Wang et al. [35] further introduced generative adversarial networks [33] to generate more subtle samples to facilitate CD-FER. More recently, Li et al. [16] observed that the conditional probability distributions between the source and target datasets are different. Based on this observation, they developed a deep emotion-conditional adaptation network (ECAN) that simultaneously considers conditional distribution bias and the expression class imbalance problem in CD-FER.

Though each work claims to achieve superior performance to previous algorithms, the comparisons are somewhat unfair, as these works often select completely different feature extractors and are evaluated on different source/target datasets. Thus, it is difficult to assess the actual improvement of each algorithm. To promote a fair comparison, we build a unified and comprehensive CD-FER evaluation benchmark by unifying the choices of the source/target datasets and feature extractor for the competing algorithms. This benchmark is novel and crucial for the CD-FER community. In addition, we propose a novel adversarial graph representation adaptation framework, which integrates graph propagation networks with adversarial learning mechanisms for adaptive holistic-local feature co-adaptation to facilitate CD-FER.

2.2 Adversarial Domain Adaptation

Obvious domain discrepancies commonly exist among different datasets. Recently, variant domain adaptation methods [26], [27], [31] have been intensively proposed to learn domain-invariant features; thus, classifiers/predictors learned using source datasets can be generalized to target test datasets. Motivated by generative adversarial networks [33] that aim to generate samples that are indistinguishable from real samples, recent domain adaptation methods [26], [31] also resort to adversarial learning to mitigate domain shifts. Specifically, adversarial learning involves a two-player game in which a feature extractor aims to learn transferable domain-invariant features while a domain discriminator struggles to distinguish samples from the source domain from those from the target domain. As a pioneering work, Tzeng et al. [31] propose a generalized adversarial adaptation framework by combining discriminative modeling, untied weight sharing, and an adversarial loss. Long et al. further designed a conditional adversarial domain adaptation method that further introduces two strategies of multilinear conditioning and entropy conditioning to improve the discriminability and control the uncertainty of the classifier. Despite achieving impressive progress for cross-domain general image classification, these methods [26], [31] mainly focus on holistic features for adaptation and ignore local content, which carries more transferable and fine-grained features. Different from these works, we propose to represent the correlation of holistic and local features by structured graphs and integrate graph propagation networks with adversarial learning to learn domain-invariant holistic-local features.

2.3 Graph Representation Learning

Deep convolutional neural networks (CNNs) [48], [49] have achieved impressive performance in visual recognition tasks [49], [50], [51], [52], [53], but these networks deal with only gridded data and are difficult to adapt to graph-structured data. To solve this issue, recent efforts have been dedicated to devising a series of graph neural networks [54], [55] that can learn the representation of graph-structured data via iterative message propagation. Recently, these graph neural networks have also been adapted to model visual feature interactions to facilitate variant tasks [56], [57], [58], [59], [60], [61], [62], ranging from object classification and detection [56], [58], [62], [63] to visual relationship reasoning [60], [61] and visual navigation [64] to traffic forecasting [65], [66]. For example, references [56], [58] modeled label dependencies with structured graphs and used graphs to guide feature and classifier learning to facilitate multilabel image recognition. Jiang et al. [62] introduced semantic label/attribute relationships and spatial relationships to help learn contextualized features and applied them to boost the large-scale object detection performance. Chen et al. [60] further extended graph neural networks to capture the interactions between candidate objects and their relationships, which can improve the visual relationship detection performance and alleviate the performance degradation caused by the long-tail distribution issue. Wang et al. [61] introduced a graph to model the correlations between social relationships and related semantic objects and applied graph neural networks to perform message propagation through the graph to capture their interactions. Inspired by these works, we introduce graph neural networks to capture the interactions among holistic-local features within each domain and across different domains and integrate them with adversarial learning to facilitate learning fine-grained domain-invariant features.

3 Unified Evaluation Benchmark

In this section, we present the datasets, feature extractors and competing algorithms involved in constructing the unified CD-FER evaluation benchmark, along with the analysis of performance gap caused by the inconsistent choices of the source/target datasets and feature extractors. Then, we present the unified evaluation protocols of the benchmark for fair comparison.

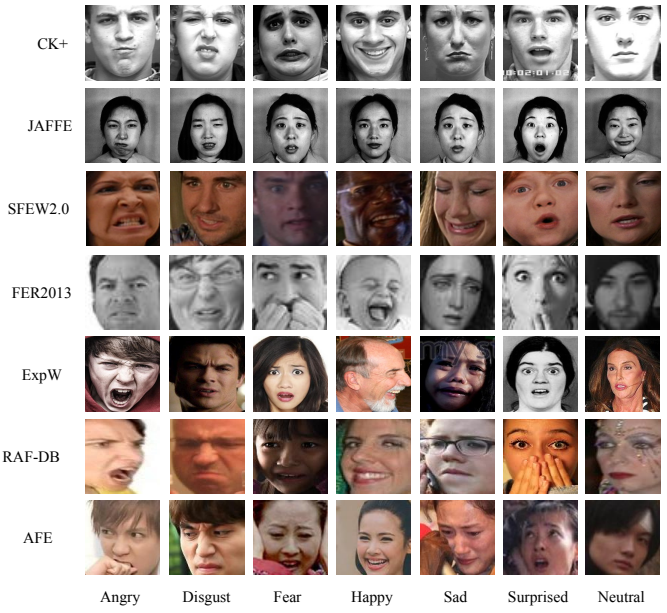


Fig. 3. Visualization of samples from the CK+, JAFFE, SFEW2.0, FER2013, ExpW, and RAF-DB datasets. The examples from different datasets differ in appearance, color, and view point.

3.1 Datasets for CD-FER

There exist enormous FER datasets, and they also serve as the source/target datasets for CD-FER. Here, we mainly discuss several publicly available datasets that are widely used as source/target datasets in CD-FER works, including lab-controlled (e.g., the CK+ [5] and JAFFE [6]) and in-the-wild (e.g., the FER2013 [14], SFEW2.0 [11], ExpW [39], RAF-DB [10]) datasets. These datasets cover only the seven basic expressions. Additionally, we introduce the newly built AFE dataset here.

CK+ [5] is a lab-controlled dataset that is mostly used for FER. It contains 593 videos from 123 subjects, among which 309 sequences are labeled with six basic expressions based on the Facial Action Coding System (FACS). We follow previous work [17] to select the three frames with peak formation from each sequence and the first frame (neutral expression) of each sequence, resulting in 1,236 images for evaluation. The dataset is divided into a training set of 1,125 and a test set of 129 images.

JAFFE [6] is another lab-controlled dataset that contains 213 images from 10 Japanese females. Approximately 3-4 images of each person are annotated with one of the six basic expressions and 1 image annotated with a neutral expression. This dataset covers only Asian people and could be used for cross-culture evaluation. As this dataset merely contains 213 images, we follow previous works [10], [17] to use the whole dataset for training and test sets.

SFEW2.0 [11] is an in-the-wild dataset collected from different films with spontaneous expressions, various head poses, age ranges, occlusions and illuminations. This dataset is divided into training, validation, and test sets, with 958, 436, and 372 samples, respectively.

FER2013 [14] is a large-scale uncontrolled dataset that was automatically collected by the Google Image Search application programming interface (API). It contains 35,887 images of size 48×48 pixels, and each image is annotated with the seven basic expressions. The dataset is further divided into a training set of 28,709 images, a validation set of 3,589 images, and a test set of 3,589 images.

ExpW [39] is an in-the-wild dataset with images that have been downloaded from Google Image searches. This dataset contains 91,793 face images, and each image is manually annotated with one of the seven expressions. In the experiments, we divide the dataset into a training set of 28,848 images, a validation set of 28,848 images, and a test set of 34,097 images.

RAF-DB [10] contains 29,672 highly diverse facial images from thousands of individuals that were also collected from the Internet. Among these images, 15,339 images are annotated with the seven basic expressions, which are divided into 12,271 training samples and 3,068 testing samples for evaluation.

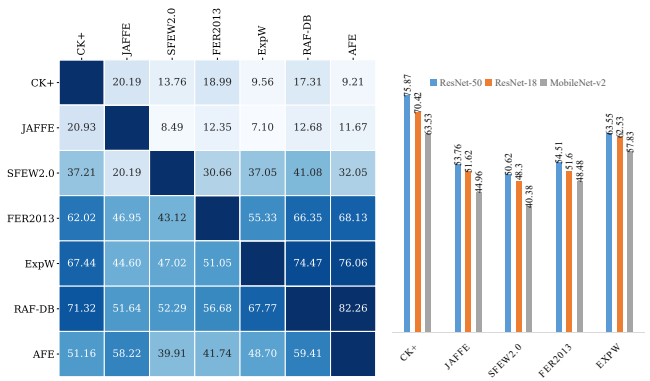
AFE is a new dataset constructed in this work that covers thousands of Asian individuals. To collect this dataset, we first downloaded approximately 500,000 images of faces from the film *DouBan*[1]. Then, each image was annotated by 3-4 annotators, and only the image that all annotators set to the same expression are kept, leading to 54,901 well-labeled samples. The dataset is further divided into 32,757 images for training, 16,380 images for validation, and 5,464 images for testing. As most images from existing datasets are of Americans and Europeans, this dataset can be used for cross-culture domain adaptation. As described above, JAFFE [6] also covers Asian people, but it contains merely 213 images. In contrast, AFE contains much more images (i.e., more than 50,000), and it can better facilitate cross-culture FER.

As mentioned above, these datasets are different from each other (see Figure 3). Thus, the different choices of either the source or the target datasets may inevitably lead to a performance gap. As exhibited in Table 1, current algorithms select different datasets or even combine multiple datasets as the source, and these algorithms are also evaluated on different target datasets, leading to extremely unfair comparisons. To quantitatively analyze this point, we conduct an experiment that trains a ResNet-50 baseline [49] on each dataset and tests the network on all the datasets without fine-tuning. As shown in Figure 4(a), we find that testing on the same target dataset but selecting different source datasets may result in a more than 70% accuracy gap, while using the same source dataset but testing on different target datasets leads to a more than 30% accuracy gap.

3.2 Feature Extractors for CD-FER

Features play a key role in visual recognition tasks, and current algorithms mainly use deep neural networks to extract learnable visual features. However, the features extracted by different deep networks inherently have different discrimination and generalization abilities. Current methods use different feature extractors, ranging from classical Visual Geometry Group (VGG), Inception and ResNet models [48], [49], [67] to different manually designed networks (see Table 1), further aggravating the unfairness of comparisons.

1. https://movie.douban.com/



(a) (b)

Fig. 4. (a) Accuracies of the cross-dataset evaluation using the ResNet-50 baseline and (b) the accuracies of cross-dataset evaluation using the ResNet-50, ResNet-18, and MobileNet-v2 baselines.

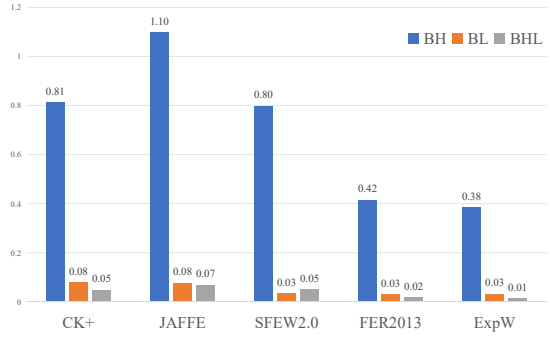


Fig. 5. The MMD comparisons between features of the source RAF-DB and those of each target CK+, JAFFE , FER2013, SFEW2.0, ExpW for the baseline with holistic features (BH), baseline with local features (BL), and baseline with holistic-local features (BHL).

To clearly show the effect of the feature extractor choice, we conduct an experiment that uses the widely used ResNet-50, ResNet-18, and lightweight MobileNet-v2 [68] networks as backbone networks for feature extraction and perform cross-dataset validation by training on the RAF-DB dataset and testing on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets. As shown in Figure 4(b), adopting different backbone networks leads to an accuracy gap of approximately 10% when using the same source and target datasets.

3.3 Algorithms for CD-FER

Another issue that prevents fair evaluation is that most CD-FER algorithms do not release the codes, and thus, it is difficult to evaluate different algorithms with the same source/target datasets and feature extractors. To address these issues, we re-implement several fruitful CD-FER methods according to the detailed descriptions in each paper, including the ICID algorithm [24], discriminative feature adaptation (DFA) [23], the locality-preserving loss (LPL) [9], a deep emotion transfer network (DETN) [17], a fine-tuned deep convolutional network (FTDNN) [25], and an ECAN [16]. In addition, many general domain adaptation algorithms exist, and we adapt some algorithms to address the CD-FER task. To this end, we select several recently published and advanced algorithms, i.e., conditional adversarial domain adaptation (CADA) [26], the stepwise adaptive feature norm (SAFN) [27], the sliced Wasserstein discrepancy (SWD) [28], Joint Unbalanced MiniBatch OT (JUMBOT) [29], Enhanced Transport Distance (ETD) [30], and use the codes released by the authors for implementation. We also implement two more baselines that can be easily adapted to address the CD-FER task, namely, direct transferring (DT) that directly transfers the model trained on the source domain to test samples of the target domain, and pseudo-label fine-tuning (PLFT) that uses the model to annotate the target domain (pseudo labels) and then conducts fine-tuning on the target domain.

3.4 Unified Evaluation Protocols

The evaluation benchmark currently contains fourteen competing algorithms, including our AGRA method. For fair comparisons, we provide unified evaluation protocols to ensure that all the algorithms can be evaluated with the same source/target datasets and the same feature extractor. Through the provided protocols, the competing algorithms can feasibly select different options of source/target datasets and feature extractors. More details are described as follows.

Dataset choice. In the evaluation benchmark, all the seven datasets presented in Sec. 3.1 can be used as source or target datasets. Due to limited space, we present in the paper the comparison results of all the methods using two representative datasets (i.e., RAF-DB and AFE) as the source datasets. The reason is that RAF-DB can achieve the overall best cross-dataset testing performance (see Figure 4(a)) and the AFE dataset can be used to evaluate the cross-culture FER ability of the methods. The corresponding evaluation results are exhibited in Table 2. For more comprehensive evaluation, we have conducted experiments using each of seven datasets as the source domain and the remaining ones as the target domain. The comparison results are reported in the supplementary materials.

Feature extractor choice. We unify the feature extractors for all competing methods to eliminate the effect of feature inconsistency. Here, we adopt ResNet-50, ResNet-18, and MobileNet-v2 as the backbone networks for the feature extractors. The reason is that ResNet-50 and ResNet-18 are most widely used for visual recognition and MobileNet-v2 is lightweight and can be adapted for mobile device applications.

In addition, our proposed AGRA method aggregates both holistic and local features to facilitate CD-FER, while the above-mentioned competing methods just use holistic features. To ensure fair comparison between our method and the competing methods, we follow the same process as described in Sec. 4.4.1 to extract holistic and local features, and concatenate them as the input for all these competing methods. The results that use ResNet-50 as the backbone network and RAF-DB as the source dataset are presented in Table 1, and more results are shown in Sec. 5 and in the supplemental materials.

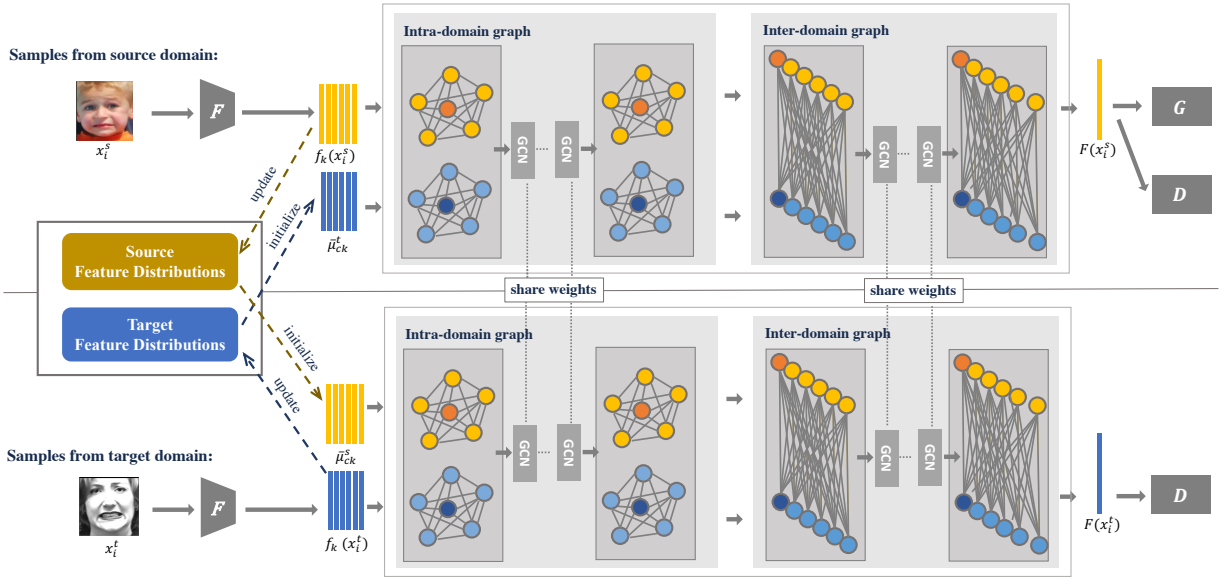


Fig. 6. An illustration of the proposed AGRA framework. The framework builds two graphs to correlate holistic-local features within each domain and across different domains, initializes the graph nodes with input image features of a certain domain and the learnable statistical distribution of the other domain, and introduces two stacked GCNs to propagate node information with each domain and transfer node messages across different domains for holistic-local feature co-adaptation. Note that the nodes in the intra-domain and inter-domain graphs are the same, and we arrange them in different layouts for a clearer illustration of the connections. The feature extractor F and domain discriminator D are also the same for the source and target domains, and we present two weight-sharing branches merely for simple illustration.

4 AGRA Framework

4.1 Overview

In the CD-FER task, a source domain FER dataset $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ and a target domain FER dataset $\mathcal{D}_t = \{(x_i^t)\}_{j=1}^{n_t}$ are provided. The two datasets are sampled from two different distributions: $p_s(X, Y)$ and $p_t(X, Y)$. Each sample from the source data x_i^s has a label y_i^s, while the samples from the target dataset do not have labels. A learned model is required to perform well on the target dataset.

To address this task, the proposed AGRA framework builds on the adversarial cross-domain mechanism that learns domain-invariant features via two-player games

$$\min_{D} \mathcal{L}(F, G, D) \tag{1}$$

$$\min_{G,F} \mathcal{L}(F, G) - \mathcal{L}(F, G, D) \tag{2}$$

where

$$\begin{aligned} \mathcal{L}(F, G) &= -\mathbb{E}_{(x^s, y^s)\sim\mathcal{D}_s} \ell(G(F(x^s)), y^s) \\ \mathcal{L}(F, G, D) &= -\mathbb{E}_{(x^s, y^s)\sim\mathcal{D}_s} \log\left[D(F(x^s))\right] \\ &\quad -\mathbb{E}_{x^t\sim\mathcal{D}_t} \log\left[1 - D(F(x^t))\right] \end{aligned} \tag{3}$$

Here, F is the feature extractor, G is the classifier, and D is the domain discriminator. As suggested in the above two objectives, the feature extractor targets on generating transferable features that can foolish the domain discriminator, while the domain discriminator aims to distinguish the samples of the source domain from those of the target domain. In this way, it can gradually reduce the domain shift and learn domain-invariant image features that are transferable across both the source and target domains. Thus, the classifier trained with only the labeled samples from the source domain can be used to classify samples from both domains.

Many works have applied the above adversarial mechanism to domain adaptation tasks, but they mainly extract holistic features for domain adaptation and usually ignore local patterns that are more transferable. With regard to the CD-FER task, these local features are also valuable, as this task requires a fine-grained and detailed understanding of the face images. For a more direct and in-depth analysis, we conduct experiments on the generalization abilities of different types of features. Specifically, we design three baselines with the same ResNet-50 backbone and use three different heads to extract holistic, local, and holistic-local features, namely BH, BL, BHL. We use the same process as described in Sec. 4.4.1 to extract the holistic and local feature vectors. For the BH baseline, we directly use the 64-dimensional holistic feature vector. For the BL baseline, we concatenate the five local feature vectors and use a fully-connected layer followed by an rectified linear unit (ReLU) function to map the concatenated feature vectors to a 64-dimensional vector. For the BHL baseline, we concatenate the holistic and five local feature vectors and use a fully-connected layer followed by an ReLU function to map the concatenated feature vector to a 64-dimensional vector. Then, we train the three baselines on RAF-DB and use the trained models to extract features for the samples of both the source RAF-DB and target CK+ [5], JAFFE [6], FER2013 [14], SFEW2.0 [11], and ExpW [39]. Finally, we compute the maximum mean discrepancy (MMD) [69] between features of the source RAF-DB dataset and those of each target dataset. As shown in Figure 5, using the local features can greatly decrease the MMDs compared with using the holistic features, and integrating holistic-local features can generally further decrease the MMDs. These comparisons suggest that local features have better generalization abilities than holistic features and that integrating holistic-local features can further strengthen these abilities. The way to effectively integrating the two for

CD-FER is worthy investigating.

In this work, we propose to represent correlation of holistic-local features in structured graphs and integrate graph propagation networks with adversarial learning mechanisms to learn domain-invariant holistic-local features for CD-FER. To this end, we extract several discriminative regions based on facial landmarks and build an intra-domain graph to correlate holistic-local regions within each domain and an inter-domain graph to correlate these regions across different domains. We develop a class-aware two-stage updating mechanism to iteratively learn the per-class statistical feature distributions for both the holistic and local regions from both domains. Given an input image from one domain, we extract the holistic-local features from corresponding regions to initialize the graph nodes of this domain and apply the statistical feature distribution to initialize the graph nodes of the other domain. Finally, we use two stacked GCNs to propagate messages through the intra-domain graph to explore holistic-local feature interactions and transfer information across the inter-domain graph to enable holistic-local co-adaptation. An overall pipeline of the proposed AGRA framework is illustrated in Figure 6.

4.2 Graph Construction

In this section, we introduce the constructions of the intra-domain and inter-domain graphs. According to the FACS [70], [71], facial expression can be interpreted as facial action units, and most of these action units are defined at the regions centered on the left eye (*le*), right eye (*re*), nose (*no*), left mouth corner (*lm*), and right mouth corner (*rm*). Thus, these regions contain the most detailed information for FER, and we extract the holistic face and further crop the five local regions. We then build the two graphs $\mathcal{G}_{intra} = (\mathbf{V}, \mathbf{A}_{intra})$ and $\mathcal{G}_{inter} = (\mathbf{V}, \mathbf{A}_{inter})$. $\mathbf{V} = \{v_h^s, v_{le}^s, v_{re}^s, v_{no}^s, v_{lm}^s, v_{rm}^s, v_h^t, v_{le}^t, v_{re}^t, v_{no}^t, v_{lm}^t, v_{rm}^t\}$ is the node set denoting the holistic image and five local regions of the source and target domains, and it is the same for both graphs. $\mathbf{A}_{intra}$ is the prior intra-domain adjacency matrix denoting the connections among nodes within each domain. It contains two types of connections; the first type is holistic-to-local connections, and the second type is local-to-local connections. $\mathbf{A}_{inter}$ is the prior inter-domain adjacency matrix denoting the connections between nodes from the different domains. Similarly, it contains three types of connections: holistic-to-holistic connections, holistic-to-local connections, and local-to-local connections. We use different values to denote different connections.

4.3 Graph Representation Adaptation

Once the two graphs are constructed, message propagations are performed through the intra-domain graph to explore holistic-local feature interactions with each domain and through the inter-domain graph to enable holistic-local feature co-adaptation. As suggested in previous works [54], GCNs [54] can effectively update node features of graph-structured data by iteratively propagating node massages to the neighborhood nodes. In this work, we apply two stacked GCNs to propagate messages through the two graphs.

As discussed above, the graphs contain nodes from two domains. Given an input sample of one domain d ($d \in \{s, t\}$), we extract the features of the corresponding regions to initialize the nodes of domain d. It is expected that these features can interact with the feature distributions of the other domain, and thus, the model can gradually reduce the domain shift. In addition, motivated by a previous work [72], it is essential to integrate class information to enable finer-grained intraclass interaction and adaptation. To this end, we estimate the per-class statistical feature distributions of each domain, i.e., $\bar{\mu}_{ck}^s$ and $\bar{\mu}_{ck}^t$, where $c \in \{0, 1, \ldots, C-1\}$ is the class label and $k \in \{h, le, re, no, lm, rm\}$ is the node type. This estimation is implemented by a class-aware two-stage updating mechanism as follows.

4.3.1 Class-aware two-stage updating mechanism

Here, we update the statistical distribution by epoch-level clustering that reclusters the samples to obtain the distribution every E epochs and iteration-level updating that updates the distribution every iteration. Specifically, we first extract features for all the samples from both the source and target datasets using the backbone network pretrained using the labeled source samples. For each domain, we divide the samples into C clusters using the K-means clustering algorithm and compute the means for each cluster to obtain the initial statistical distribution, which is formulated as

$$\begin{aligned} \bar{\mu}_{ck}^s &= \frac{1}{n_c^s} \sum_{i=1}^{n_c^s} f_k(x_{ci}^s) \\ \bar{\mu}_{ck}^t &= \frac{1}{n_c^t} \sum_{j=1}^{n_c^t} f_k(x_{ci}^t) \end{aligned} \tag{4}$$

where $f_k(\cdot)$ is the feature extractor for region k; $n_c^{s/t}$ is the number of samples in cluster c of domain s/t; and $x_{ci}^{s/t}$ is the i-th sample of cluster c. During training, we further use the moving average to iteratively update these statistical distributions in a progressive manner. For each batch iteration, we compute the distances between each sample and the distributions of each cluster. These samples are grouped into the cluster with the smallest distance. Then, we compute the mean features (i.e., μ_{ck}^s and μ_{ck}^t) over the samples in the same cluster and update the statistical distribution by

$$\begin{aligned} \bar{\mu}_{ck}^s &= (1-\alpha)\bar{\mu}_{ck}^s + \alpha\mu_{ck}^s \\ \bar{\mu}_{ck}^t &= (1-\alpha)\bar{\mu}_{ck}^t + \alpha\mu_{ck}^t \end{aligned} \tag{5}$$

where α is a balance parameter, which is set to 0.1 in our experiments. To avoid distribution shifting, this process is repeated every E epochs. Then, we recluster the samples to obtain new distributions for each cluster according to equation 4. Epoch-level reclustering and iteration-level updating are iteratively performed along with the training process to obtain the final statistical distributions.

4.3.2 Stacked graph convolutional networks

As discussed above, we use two stacked GCNs: one GCN propagates messages through the intra-domain graph to explore holistic-local feature interactions within each domain, and the GCN transfers messages through the inter-domain GCN to enable holistic-local feature co-adaptation. In this section, we describe the two GCNs in detail.

Given an input sample x_i^s from the source domain, we can extract features of the holistic image and the corresponding local regions to initialize the corresponding node of the source domain

$$h_{intra,k}^{s,0} = f_k(x_i^s). \tag{6}$$

Then, we compute the distance between this sample and the feature distributions of all clusters of the target domain and obtain the cluster c with the smallest distance. Then, each node of the target domain is initialized by the corresponding feature distribution

$$h_{intra,k}^{t,0} = \bar{\mu}_{ck}^t. \tag{7}$$

The initial features are then rearranged to obtain feature matrix $\mathbf{H}_{intra}^0 \in \mathcal{R}^{n \times d_{intra}^0}$, where $n = 12$ is the number of nodes. Then, we perform a graph convolution operation on the input feature matrix to iteratively propagate and update the node features, which is formulated as

$$\mathbf{H}_{intra}^l = \sigma(\widehat{\mathbf{A}}_{intra}\mathbf{H}_{intra}^{l-1}\mathbf{W}_{intra}^{l-1}), \tag{8}$$

By stacking L_{intra} graph convolutional layers, the node messages are fully explored within the intra-domain graph, and the feature matrix $\mathbf{H}_{intra}$ is obtained. This feature matrix is then used to initialize the nodes of the inter-domain graph

$$\mathbf{H}_{inter}^0 = \mathbf{H}_{intra}. \tag{9}$$

The graph convolution operation is performed to iteratively update the node features

$$\mathbf{H}_{inter}^l = \sigma(\widehat{\mathbf{A}}_{inter}\mathbf{H}_{inter}^{l-1}\mathbf{W}_{inter}^{l-1}), \tag{10}$$

Similarly, the graph convolution operation is repeated L_{inter} times, and the final feature matrix $\mathbf{H}$ is generated. We concatenate the features of nodes from the source domain as the final feature $F(x_i^s)$, which is fed into the classifier to predict the expression label and domain discriminator to estimate its domain. The two matrices $\widehat{\mathbf{A}}_{intra}$ and $\widehat{\mathbf{A}}_{inter}$ are initialized by the prior matrices $\mathbf{A}_{intra}$ and $\mathbf{A}_{inter}$ and jointly fine-tuned to learn better relationships during the training process.

Similarly, given a sample from the target domain, the nodes of the source domain are initialized by the corresponding extracted feature, and those of the target domain are initialized by the corresponding statistical feature distributions. Then, the same process is conducted to obtain the final feature $F(x_i^t)$. As the final feature does not have expression label annotation, it is merely fed into the domain discriminator for domain estimation.

4.4 Implementation Details

4.4.1 Network architecture

As stated in previous works, we use ResNet-50, ResNet-18, and MobileNet-v2 [49], [73] as backbone networks for feature extraction. All three networks consist of four block layers. Given an input image with a size of $112 \times 112 \times 3$, we can obtain feature maps with the size of $28 \times 28 \times 128$ from the second layer and feature maps with the size of $7 \times 7 \times 512$ from the fourth layer. For the holistic features, we perform a convolution operation to obtain feature maps with the size of $7 \times 7 \times 64$, which is followed by an average pooling layer to obtain a 64-dimensional vector. For the local features, we use a multi-task CNN (MT-CNN) [36] to locate the landmarks and use the feature maps from the second layer as they have a higher resolution. Specifically, we crop $7 \times 7 \times 128$ feature maps centered at the corresponding landmark and use similar convolution operations and average pooling to obtain a 64-dimensional vector for each region.

The intra-domain GCN consists of two graph convolutional layers with 128 and 64 output channels. Thus, the sizes of the parameter matrices $\mathbf{W}_{intra}^0$ and $\mathbf{W}_{intra}^1$ are 64×128 and 128×64, respectively. The inter-domain GCN contains only one graph convolutional layer, and the number of output channels is also set to 64. The parameter matrix $\mathbf{W}_{inter}^0$ has a size of 64×64. We perform ablation studies to analyze the effect of the number of layers of the two GCNs and find setting them to 2 and 1 obtains the best results.

The classifier is simply implemented by a fully connected layer that maps the 384-dimensional (i.e., 64×6) feature vector to seven scores that indicate the confidence of each expression label. The domain discriminator is implemented by two fully connected layers with an ReLU nonlinear function, followed by another fully connected layer to one score to indicate its domain.

4.4.2 Training details

The AGRA framework is trained with the objectives of equations 1 and 2 to optimize the feature extractor, classifier, and domain discriminator. Here, we follow previous domain adaptation works [74] to adopt a two-stage training process. We initialize the parameters of the backbone networks with those pretrained on the MS-Celeb-1M [75] dataset and the parameters of the newly added layers with the Xavier algorithm [76]. In the first stage, we train the feature extractor and classifier with the cross-entropy loss using stochastic gradient descent (SGD) with an initial learning rate of 0.0001, a momentum of 0.9, and a weight decay of 0.0005. It is trained for approximately 15 epochs. In the second stage, we use the objective loss in equation 1 to train the domain discriminator and the objective loss in equation 2 to fine-tune the feature extractor and the classifier. It is also trained using SGD with the same momentum and weight decay as the first stage. The learning rate for the feature extractor and the source classifier is initialized at 0.0001, and it is divided by 10 after approximately 10 epochs. As the domain discriminator is trained from scratch, we initialize it at 0.001 and divide it by 10 when the error saturates.

4.4.3 Inference details

Given an input image, we extract holistic and local images to initialize the corresponding nodes of the target domain. Then, we compute the distances between the given image and all the per-class feature distributions of the source domain. We select the feature distributions with the smallest distance to initialize the nodes of the source domain. After GCN message propagation, we can obtain its feature and feed it into the classifier to predict the final score vector.

5 Experiments

In this section, we present the results of all the re-implemented algorithms and the proposed AGRA frame-

Method	Source set	Backbone	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
DT	RAF-DB	ResNet-50	71.32	50.23	50.46	54.49	67.45	58.79
PLFT	RAF-DB	ResNet-50	77.52	53.99	48.62	56.46	69.81	61.28
ICID [24]	RAF-DB	ResNet-50	74.42	50.70	48.85	53.70	**69.54**	59.44
DFA [23]	RAF-DB	ResNet-50	64.26	44.44	43.07	45.79	56.86	50.88
LPL [9]	RAF-DB	ResNet-50	74.42	53.05	48.85	55.89	66.90	59.82
DETN [17]	RAF-DB	ResNet-50	78.22	55.89	49.40	52.29	47.58	56.68
FTDNN [25]	RAF-DB	ResNet-50	79.07	52.11	47.48	55.98	67.72	60.47
ECAN [16]	RAF-DB	ResNet-50	79.77	57.28	52.29	56.46	47.37	58.63
CADA [26]	RAF-DB	ResNet-50	72.09	52.11	53.44	57.61	63.15	59.68
SAFN [27]	RAF-DB	ResNet-50	75.97	61.03	52.98	55.64	64.91	62.11
SWD [28]	RAF-DB	ResNet-50	75.19	54.93	52.06	55.84	68.35	61.27
JUMBOT [29]	RAF-DB	ResNet-50	79.46	54.13	51.97	53.56	63.69	60.56
ETD [30]	RAF-DB	ResNet-50	75.16	51.19	52.77	50.41	67.82	59.47
Ours	RAF-DB	ResNet-50	**85.27**	**61.50**	**56.43**	**58.95**	68.50	**66.13**
DT	RAF-DB	ResNet-18	68.22	49.30	49.31	52.71	67.63	57.34
PLFT	RAF-DB	ResNet-18	74.42	53.05	48.62	54.21	69.06	59.67
ICID [24]	RAF-DB	ResNet-18	67.44	48.83	47.02	53.00	68.52	56.96
DFA [23]	RAF-DB	ResNet-18	54.26	42.25	38.30	47.88	47.42	46.02
LPL [9]	RAF-DB	ResNet-18	72.87	53.99	49.31	53.61	68.35	59.63
DETN [17]	RAF-DB	ResNet-18	64.19	52.11	42.25	42.01	43.92	48.90
FTDNN [25]	RAF-DB	ResNet-18	76.74	50.23	49.54	53.28	68.08	59.57
ECAN [16]	RAF-DB	ResNet-18	66.51	52.11	48.21	50.76	48.73	53.26
CADA [26]	RAF-DB	ResNet-18	73.64	55.40	52.29	54.71	63.74	59.96
SAFN [27]	RAF-DB	ResNet-18	68.99	49.30	50.46	53.31	68.32	58.08
SWD [28]	RAF-DB	ResNet-18	72.09	53.52	49.31	53.70	65.85	58.89
JUMBOT [29]	RAF-DB	ResNet-18	76.67	52.10	49.19	50.58	61.45	58.00
ETD [30]	RAF-DB	ResNet-18	72.34	49.44	49.67	47.66	64.62	56.75
Ours	RAF-DB	ResNet-18	**77.52**	**61.03**	**52.75**	**54.94**	**69.70**	**63.19**
DT	RAF-DB	MobileNet-V2	66.67	38.97	41.74	49.99	63.08	52.09
PLFT	RAF-DB	MobileNet-V2	72.09	38.97	41.97	**51.11**	**64.12**	53.65
ICID [24]	RAF-DB	MobileNet-v2	57.36	37.56	38.30	44.47	60.64	47.67
DFA [23]	RAF-DB	MobileNet-v2	41.86	35.21	29.36	42.36	43.66	38.49
LPL [9]	RAF-DB	MobileNet-v2	59.69	40.38	40.14	50.13	62.26	50.52
DETN [17]	RAF-DB	MobileNet-v2	53.49	40.38	35.09	45.88	45.26	44.02
FTDNN [25]	RAF-DB	MobileNet-v2	71.32	46.01	45.41	49.96	62.87	55.11
ECAN [16]	RAF-DB	MobileNet-v2	53.49	43.08	35.09	45.77	45.09	44.50
CADA [26]	RAF-DB	MobileNet-v2	62.79	53.05	43.12	49.34	59.40	53.54
SAFN [27]	RAF-DB	MobileNet-v2	66.67	45.07	40.14	49.90	61.40	52.64
SWD [28]	RAF-DB	MobileNet-v2	68.22	55.40	43.58	50.30	60.04	55.51
JUMBOT [29]	RAF-DB	MobileNet-v2	**73.64**	51.35	44.41	49.05	60.84	55.86
ETD [30]	RAF-DB	MobileNet-v2	69.27	48.57	41.34	49.43	57.05	53.13
Ours	RAF-DB	MobileNet-v2	72.87	**55.40**	**45.64**	51.05	63.94	**57.78**
DT	AFE	ResNet-50	61.24	57.28	46.79	47.79	52.03	53.03
PLFT	AFE	ResNet-50	68.22	58.22	45.41	48.97	53.72	54.91
ICID [24]	AFE	ResNet-50	56.59	57.28	44.27	46.92	52.91	51.59
DFA [23]	AFE	ResNet-50	51.86	52.70	38.03	41.93	60.12	48.93
LPL [9]	AFE	ResNet-50	73.64	61.03	49.77	49.54	55.26	57.85
DETN [17]	AFE	ResNet-50	56.27	52.11	44.72	42.17	59.80	51.01
FTDNN [25]	AFE	ResNet-50	61.24	57.75	47.25	46.36	52.89	53.10
ECAN [16]	AFE	ResNet-50	58.14	56.91	46.33	46.30	61.44	53.82
CADA [26]	AFE	ResNet-50	72.09	49.77	50.92	50.32	61.70	56.96
SAFN [27]	AFE	ResNet-50	73.64	64.79	49.08	48.89	55.69	58.42
SWD [28]	AFE	ResNet-50	72.09	61.50	48.85	48.83	56.22	57.50
JUMBOT [29]	AFE	ResNet-50	75.36	54.38	48.38	50.75	60.74	57.92
ETD [30]	AFE	ResNet-50	73.12	51.43	49.71	50.34	62.37	57.39
Ours	AFE	ResNet-50	**78.57**	**65.43**	**51.18**	**51.31**	**62.71**	**61.84**
DT	AFE	ResNet-18	71.32	63.38	52.52	51.03	54.65	58.58
PLFT	AFE	ResNet-18	78.29	59.62	51.15	**51.98**	57.85	59.78
ICID [24]	AFE	ResNet-18	54.26	51.17	47.48	46.44	54.85	50.84
DFA [23]	AFE	ResNet-18	35.66	45.82	34.63	36.88	62.53	43.10
LPL [9]	AFE	ResNet-18	67.44	**62.91**	48.39	49.82	54.51	56.61
DETN [17]	AFE	ResNet-18	44.19	47.23	45.46	45.39	58.41	48.14
FTDNN [25]	AFE	ResNet-18	58.91	59.15	47.02	48.58	55.29	53.79
ECAN [16]	AFE	ResNet-18	44.19	60.56	43.26	46.15	62.52	51.34
CADA [26]	AFE	ResNet-18	72.09	53.99	48.39	48.61	58.50	56.32
SAFN [27]	AFE	ResNet-18	68.22	61.50	50.46	50.07	55.17	57.08
SWD [28]	AFE	ResNet-18	77.52	59.15	50.69	51.84	56.56	59.15
JUMBOT [29]	AFE	ResNet-18	68.06	53.62	47.81	48.89	60.37	55.75
ETD [30]	AFE	ResNet-18	72.08	50.79	48.46	49.38	61.79	56.50
Ours	AFE	ResNet-18	**79.84**	61.03	**51.15**	51.95	**65.03**	**61.80**
DT	AFE	MobileNet-V2	68.22	49.30	45.41	46.64	53.71	52.66
PLFT	AFE	MobileNet-V2	69.77	52.58	40.37	46.36	53.19	52.45
ICID [24]	AFE	MobileNet-v2	55.04	42.72	34.86	39.94	44.34	43.38
DFA [23]	AFE	MobileNet-v2	44.19	27.70	31.88	35.95	61.55	40.25
LPL [9]	AFE	MobileNet-v2	69.77	50.23	43.35	45.57	51.63	52.11
DETN [17]	AFE	MobileNet-v2	57.36	54.46	32.80	44.11	**64.36**	50.62
FTDNN [25]	AFE	MobileNet-v2	65.12	46.01	46.10	46.69	53.02	51.39
ECAN [16]	AFE	MobileNet-v2	71.32	**56.40**	37.61	45.34	64.00	54.93
CADA [26]	AFE	MobileNet-v2	70.54	45.07	40.14	46.72	54.93	51.48
SAFN [27]	AFE	MobileNet-v2	62.79	53.99	42.66	46.61	52.65	51.74
SWD [28]	AFE	MobileNet-v2	64.34	53.52	44.72	**50.24**	55.85	53.73
JUMBOT [29]	AFE	MobileNet-v2	67.29	54.07	46.88	47.06	54.26	53.91
ETD [30]	AFE	MobileNet-v2	66.67	49.42	46.11	46.37	52.50	52.21
Ours	AFE	MobileNet-v2	**75.19**	54.46	**47.25**	47.88	61.10	**57.18**

TABLE 2
Accuracies of our proposed framework with current leading methods on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets. The results are generated by our implementation with exactly the same source dataset and backbone network.

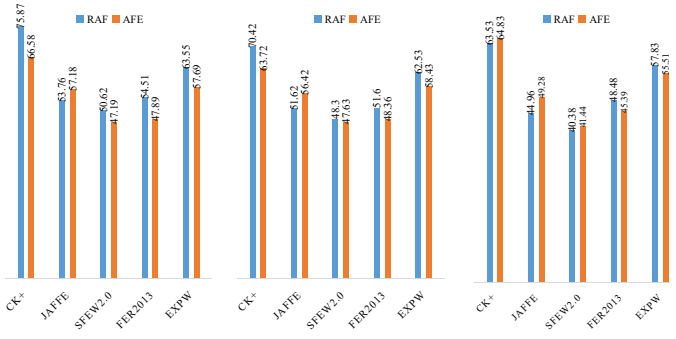


Fig. 7. The average accuracy comparisons using the RAF-DB and AFE datasets as the source dataset. We average the accuracies of all the methods using ResNet-50 (left), ResNet-18 (middle), and MobileNet-v2 (right) as the backbone.

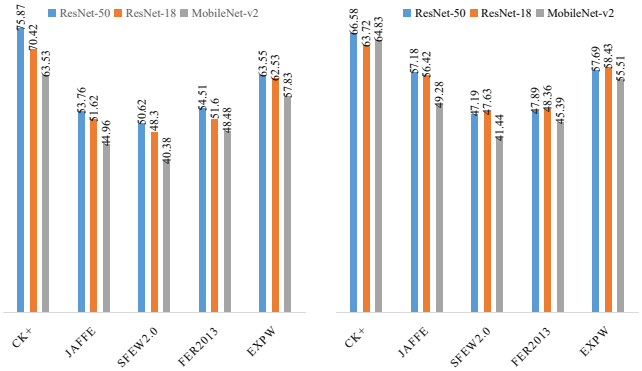


Fig. 8. The average accuracy comparisons using ResNet-50, ResNet-18, and MobileNet-v2 as the backbone. We average the accuracies of all the methods using the RAF-DB (left) and AFE (right) datasets as the source datasets.

work under the fair evaluation setting and perform an ablation study to analyze the actual contribution of each component.

5.1 Performance Evaluation and Analysis

5.1.1 Effect of the inconsistent choice

In this part, we use the unified evaluation benchmark to analyze the performance impact of the inconsistent choices of the source/target datasets and backbone networks. The performances of all the methods are presented in Table 2.

Effect of dataset inconsistency: The source dataset provides basic supervision for recognition, and the distribution similarity with the target dataset is key for the final performance. To compare the performance using different source datasets, we compute the average value over all the methods that use the same backbone and test on the same target dataset, as shown in Figure 7. We find that using the AFE dataset as the source dataset performs better on the JAFFE dataset, while using the RAF dataset performs well on the remaining CK+, SFEW2.0, FER2013, and ExpW datasets. One possible reason for this phenomenon is that the JAFFE dataset is collected from Japanese people, and its distribution is more similar to that of the AFE dataset. In contrast, the remaining datasets are mainly captured from Western areas, and their distributions are more similar to those of the RAF dataset. On the other hand, when using the same source dataset and the same backbone, the performances of the different target datasets are also different. For example, the accuracies of the SWD vary from 52.06% to 75.19% if using the RAF-DB source dataset and the ResNet-50 backbone. This phenomenon is natural because different target datasets have different difficulties and different similarities with the source dataset.

Effect of feature extractor inconsistency: Feature extractors with different backbones can learn features that have inherently different discrimination and generalization abilities. Similarly, to compare the performance using different backbones, we compute the average value over all the methods that use the same source and target datasets. The results are presented in Figure 8. We find that the performances decrease with the backbone choice from ResNet-50 to ResNet-18 to MobileNet-v2 because their discrimination and generalization abilities successively weaken.

5.1.2 Comparison of the AGRA with current state-of-the-art algorithms

In this part, we use the unified evaluation benchmark to compare the proposed AGRA approach with the current competing methods. As shown in Table 2, the proposed AGRA method consistently outperforms all the current methods for almost all the source/target datasets and backbones. Specifically, when using the RAF source dataset and ResNet-18 backbone, our AGRA approach obtains accuracies of 77.52%, 61.03%, 52.75%, 54.94%, 69.70% on the CK+, JAFFE, SFEW2.0, FER2013, ExpW datasets, outperforming all of the current best-performing methods. For a comprehensive comparison, we average the accuracies of all the target datasets to obtain the mean accuracy for each source dataset and backbone network choice. As shown, our AGRA approach achieves the best mean accuracies for all the RAF-DB/ResNet-50, RAF-DB/ResNet-18, RAF-DB/MobileNet-v2, AFE/ResNet-50, AFE/ResNet-18, and AFE/MobileNet-v2 choices.

5.2 Ablation Study

In this subsection, we conduct ablation studies to discuss and analyze the actual contribution of each component and obtain a more thorough understanding of the framework. To ensure a fair comparison and evaluation, the experiments are conducted with the same ResNet-50 model as the backbone and RAF-DB dataset as the source domain. We eliminate the backbone and source dataset information for convenient illustration.

Method	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Ours HFs	72.09	52.11	53.44	57.61	63.15	59.68
Ours HLFs	72.09	56.34	50.23	57.30	64.00	59.99
Ours	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 3
Accuracies of our approach using holistic features (HFs), concatenating holistic-local features (HLFs) and ours for adaptation on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

5.2.1 Analysis of holistic-local feature co-adaptation

The core contribution of the proposed framework is the holistic-local feature co-adaptation module that jointly

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2021.3131222, IEEE Transactions on Pattern Analysis and Machine Intelligence

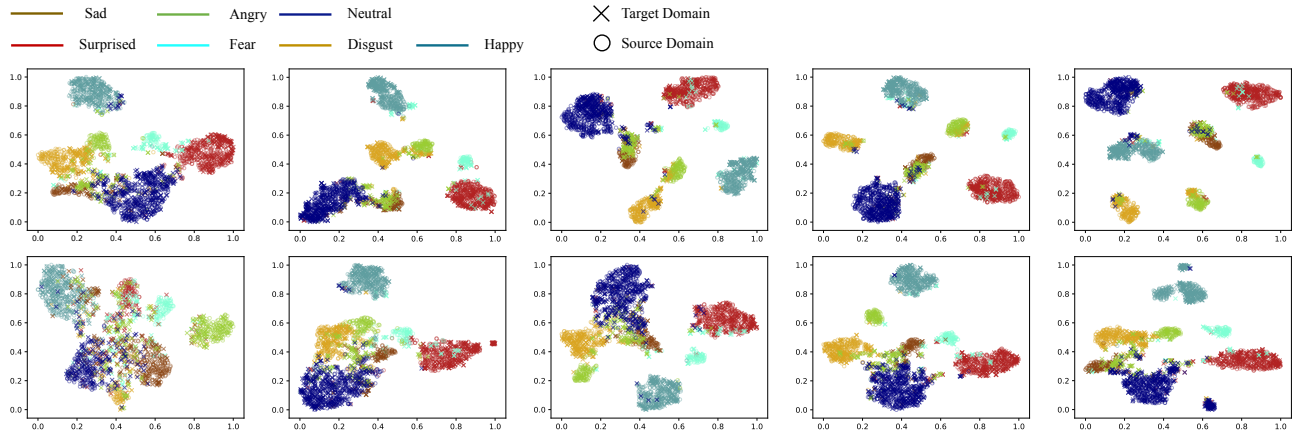


Fig. 9. Illustration of the feature distributions learned by our proposed approach at epochs 0, 5, 10, 15, and 20 (from left to right) on the CK+ (upper) and SFEW2.0 (lower) datasets.

learns domain-invariant holistic-local features. To analyze its contribution, we remove this module while keeping the others unchanged. Thus, it merely uses holistic features for adaptation (namely, Ours HF). As shown in Table 3, removing this module leads to an obvious performance drop on all the datasets. Specifically, the accuracies drop from 85.27% to 72.09%, from 61.50% to 52.11%, from 56.43% to 53.44%, from 58.95% to 57.61%, and from 68.50% to 63.15% on the five datasets, respectively. The mean accuracy drops from 66.13% to 59.68%, with a decrease of 6.45%.

These obvious performance drops well demonstrate the contribution of the co-adaptation module for CD-FER well. It is also crucial that we introduce the two stacked GCNs for holistic-local feature co-adaptations. To verify their contribution, we remove the two GCNs and simply concatenate the holistic-local features for adaptation (namely, Ours HLFs). The results are presented in Table 3. We find that concatenating the local features can improve the performance, e.g., an improvement of 0.31% in the mean accuracy. However, this approach still is outperformed by our AGRA approach on all five datasets, reducing the mean accuracy by 6.14%.

Method	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Ours intra-GCN	77.52	**61.97**	55.28	57.95	66.99	63.94
Ours inter-GCN	77.52	57.75	49.77	55.64	66.00	61.34
Ours single GCN	74.42	56.34	52.06	57.33	67.30	61.49
Ours	**85.27**	61.50	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 4
Accuracies of our approach using only the intra-domain GCN (Ours intra-GCN), using only the inter-domain GCN (Ours inter-GCN), using only one GCN (Ours single GCN), and our original approach (Ours) on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

Note that we use two stacked GCNs, in which an intra-domain GCN propagates messages within each domain to capture holistic-local feature interactions and an inter-domain GCN transfers messages across different domains to ensure domain adaptation. To demonstrate the effectiveness of this point, we conduct an experiment that uses one single GCN for message propagation within and across the source and target domains. As shown in Table 4, we find dramatic performance drops on all the datasets, e.g., a decrease in the mean accuracy by 4.64%. The reason is mainly because message propagations within each domain and across different domains are different, and using only one GCN cannot model two types of propagation well. To further analyze the actual contribution of each GCN, we conduct two more experiments. The first experiment removes the inter-domain GCN and merely performs message propagation within each domain, while the second experiment removes the intra-domain GCN, and message propagation is only carried out across different domains. We find that both experiments show obvious performance drops, i.e., a decrease in the mean accuracy by 2.19% if the inter-domain GCN is removed and a decrease in the mean accuracy by 4.79% if the intra-domain GCN is removed, as shown in Table 4.

Method	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Ours mean	82.95	52.58	55.96	58.45	65.23	63.03
Ours iter	82.17	58.28	52.98	56.40	68.32	63.63
Ours epoch	80.62	56.81	53.67	55.58	66.59	62.65
Ours	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 5
Accuracies of our approach with the mean statistical distribution (Ours mean), our approach and updating the per-class statistical distributions every iteration (Ours iter), our approach and updating the per-class statistical distributions every ten epochs (Ours epoch) and our original approach (Ours) on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

5.2.2 *Analysis of the per-class statistical distributions*

To ensure meaningful initializations for the nodes of each domain when the input image comes from the other domain, we learn the per-class statistical feature distributions. Here, we first illustrate the feature distributions of samples from the lab-controlled CK+ and in-the-wild SFEW2.0 datasets during different training stages. As shown in Figure 9, it can be observed that the proposed model can gather the samples of the same category and from different domains together, which suggests that it can learn discriminative and domain-variant features. To quantitatively analyze its contribution, we learn the dataset-level statistical feature distributions and replace the per-class statistical feature

distributions for node initialization. We find that the mean accuracy drops from 66.13% to 63.03%, as shown in Table 5.

As stated above, we learn the per-class statistical distributions by updating every iteration and reclustering every ten epochs. To analyze the effect of the updating mechanism, we conduct experiments that merely update every iteration or merely recluster every ten epochs and present the results in Table 5. We find that both experiments exhibit an obvious drop in performance; i.e., the mean accuracy decreases by 2.50% if the per-class statistical distributions are updated every iteration and by 3.48% if reclustering is performed every ten epochs.

Method	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Ours RM	68.99	50.70	54.36	55.47	67.88	59.48
Ours OM	79.07	57.28	53.90	57.07	66.71	62.81
Ours FM	68.99	47.42	54.13	53.28	56.25	56.01
Ours	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 6
Accuracies of our approach where the matrices are initialized with randomly initialized matrices (Ours RM), our approach where the matrices are initialized with all-one matrices (Ours OM), our approach where the matrices are initialized with fixed matrices (Ours FM), and our original approach (Ours) on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

5.2.3 Analysis of the adjacency matrix

We initialize the two adjacency matrices of the intra-domain and inter-domain graphs with manually defined connections, which can provide prior guidance to regularize message propagation. In this part, we replace the adjacency matrices with two randomly initialized matrices (denoted by Ours RM) and with two all-one matrices (denoted by Ours OM) to verify the effectiveness of this point. We present the results in Table 6. We observe that both experiments show severe performance degradation on all the datasets; i.e., the mean accuracies are degraded by 6.65% and 3.32%, respectively. It is noteworthy that the experiment with randomly initialized matrices exhibits more obvious performance degradation than the experiment with the all-ones matrices. One possible reason is that the randomly initialized matrices may provide misleading guidance for message propagation, which further indicates the importance of the prior adjacency matrices.

To adjust the adjacency matrices to better guide message propagation, they are also jointly fine-tuned during the training process. In this part, we verify the effectiveness by fixing the prior matrix training. We present the results in Table 6. The mean accuracy drops from 66.13% to 56.01%, which suggests that jointly adjusting the adjacency matrices can learn dataset-specified matrices, which is crucial to promoting CD-FER.

5.2.4 Analysis of GCN

In this work, we use GCN for message propagation. Increasing the number of layers of the GCN can promote deeper feature interaction, but it may lead to message smoothing and hurt the discriminative ability. Here, we present experimental studies to analyze the effect of the number of iterations (i.e., T_{intra} and T_{inter}) of both GCNs on CD-FER. To this end, we first fix T_{inter} as 1 and vary T_{intra} from 1 to 3. As shown in Table 7, the performance can be boosted by increasing T_{intra} from 1 to 2, but the performance drops when further increasing it to 3. Thus, we set the number of layers of the intra-domain GCN to 2 and conduct an experiment that varies T_{inter} from 1 to 3. We find that setting T_2 to 1 achieves the best performance and increasing this number results in performance degradation, as depicted in Table 7. Thus, we set T_{intra} to 2 and T_{inter} to 1 for all the experiments.

T_{intra}	T_{inter}	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
1	1	75.19	52.11	55.28	57.22	67.32	61.42
2	1	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**
3	1	80.62	53.06	50.46	56.82	64.41	61.07
2	2	74.42	54.46	54.59	58.31	66.94	61.74
2	3	79.07	49.77	51.61	56.85	67.14	60.89

TABLE 7
Accuracies of our approach with different numbers of iterations for the intra-domain GCN and inter-domain GCN on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

Method	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Ours TR	75.19	51.64	54.82	57.33	66.51	61.10
Ours NT	74.42	59.62	53.21	56.93	67.26	62.29
Ours GCN	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 8
Accuracies of our approach that uses GCN-based (Ours GCN), neural tensor (Ours NT), and transformer-based (Ours TR) fusion algorithms on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

On the other hand, other message propagation and fusion algorithms exist, such as neural tensor and transformer-based fusion. To verify the effectiveness of the GCN-based message propagation and fusion, we further implement two more baselines that use the neural tensor and transformer-based fusion algorithms, respectively. 1) For the neural tensor fusion based algorithm, we replace the two GCNs with two neural tensor fusion operations, in which an intra-domain neural tensor fusion operation is used to fuse the holistic and local features of the source and target domain respectively, and then an inter-domain neural tensor fusion operation is used to fuse features from both domains. 2) For the transformer-based algorithm, we similarly replace the two GCNs with two transformers, in which an intra-domain transformer is used to sequentially fuse the holistic and local features of the source and target domain respectively, and then an inter-domain transformer is used to sequentially fuse features from both domains. As shown in Table 8, the results obtained by using GCN-based fusion algorithm obviously outperforms those obtained by using these two algorithms.

5.2.5 Contribution of adversarial learning

Adversarial learning is also key to facilitating learning domain-invariant features. To validate its contribution, we remove the adversarial loss and use only the classification loss to train the whole model. As shown in Table 9, the accuracies drop to 68.22%, 49.30%, 52.98%, 56.46% and 63.93% on the five datasets, much worse than that using adversarial learning.

Method	CK+	JAFFE	SFEW2.0	FER2013	ExpW	Mean
Ours w/o AL	68.22	49.30	52.98	56.46	63.93	58.18
Ours w/ AL	**85.27**	**61.50**	**56.43**	**58.95**	**68.50**	**66.13**

TABLE 9
Accuracies of our approach with and without adversarial learning (Ours w/ AL and Ours w/o AL, respectively) on the CK+, JAFFE, SFEW2.0, FER2013, and ExpW datasets.

5.2.6 Efficiency analysis

During inference stage, the proposed framework introduces two graphs for within-domain and cross-domain message propagation. Here, we further compare the computational overhead. As suggested in Sec. 4.4.1, the intra-domain GCN consists of two layers with dimensions of 64×128 and 128×64, while the inter-domain GCN contains merely one layer with dimensions of 64×64, which introduces negligible computational overhead. As shown in Table 10, the baseline with ResNet-50 backbone has 6.7700G multiply-accumulate operations (MACs) and adding the two GCNs increases the MACs to 6.7702G, with an relative increase of 0.00295%. For more practical analysis, we further compare the running time for our method with and without the GCNs. Specifically, the experiment is conducted with a batch size of 1 and on a desktop with a single NVIDIA GTX 1070 Ti. As shown, when using ResNet-50 as backbone, the proposed framework with the GCNs needs 2.54 ms more time than that without the GCNs.

For a comprehensive analysis on this point, we also analyze the training times of our method with and without the GCNs. Specifically, we use RAF-DB as the source dataset and CK+ as the target dataset and conduct the experiment on a desktop with a single NVIDIA GTX 1070 Ti. As described in Sec. 4.4.2, it consists of one training stage on the source dataset (first training stage, shorted as FTS) and another training stage on both the source and target datasets (second training stage, shorted as STS). And we present the training time for each epoch on these two stages in Table 11. As shown, adding the GCNs increases about 5% to 20% training time for each epoch.

	ResNet-50		ResNet-18		MobileNet-v2	
Method	MACs	Time	MACs	Time	MACs	Time
Our w/o GCNs	6.7700	18.30	3.0665	15.88	2.9481	20.58
Our w/ GCNs	6.7702	20.84	3.0667	16.86	2.9483	21.87

TABLE 10
The multiply-accumulate operations (MACs) and running time of the proposed framework with and without the GCNs (Ours w/ GCN and Ours w/o GCN). The units are giga (G) for MACs and millisecond (ms) for running time.

	ResNet-50		ResNet-18		MobileNet-v2	
Method	FTS	STS	FTS	STS	FTS	STS
Our w/o GCNs	327	648	242	435	283	537
Our w/ GCNs	386	713	277	468	334	579

TABLE 11
The training time (seconds) for each epoch of the proposed framework with and without the GCNs (Ours w/ GCN and Ours w/o GCN). We select RAF-DB and CK+ as the source and target datasets.

6 Conclusion

In this work, we first analyze the inconsistent choices of the source/target datasets and feature extractors and their performance effect on the CD-FER task. Then, we construct a unified evaluation CD-FER benchmark, in which all the competing methods are compared fairly with unified source/target datasets and feature extractors. A new AFE dataset is also built and added to the benchmark. In addition, based on the observation that current leading CD-FER methods mainly focus on learning holistic domain-invariant features but ignore local features that are more transferable and carry more detailed content, we develop a novel AGRA framework that integrates the graph propagation mechanism with adversarial learning for effective holistic-local representation co-adaptation across different domains. In the experiments, we use the unified evaluation benchmark to compare the proposed AGRA framework with current state-of-the-art methods, which demonstrates the effectiveness of the proposed framework.

References

[1] N. Fragopanagos and J. G. Taylor, "Emotion recognition in human–computer interaction," *Neural Networks*, vol. 18, no. 4, pp. 389–405, 2005.

[2] J. Edwards, H. J. Jackson, and P. E. Pattison, "Emotion recognition via facial expression and affective prosody in schizophrenia: a methodological review," *Clinical psychology review*, vol. 22, no. 6, pp. 789–832, 2002.

[3] C. Clavel, I. Vasilescu, L. Devillers, G. Richard, and T. Ehrette, "Fear-type emotion recognition for future audio-based surveillance systems," *Speech Communication*, vol. 50, no. 6, pp. 487–503, 2008.

[4] S. T. Saste and S. Jagdale, "Emotion recognition from speech using mfcc and dwt for security system," in *International Conference of Electronics, Communication and Aerospace Technology*, vol. 1. IEEE, 2017, pp. 701–704.

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

[6] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE, 1998, pp. 200–205.

[7] M. Valstar and M. Pantic, "Induced disgust, happiness and surprise: an addition to the mmi facial expression database," in *Intern. Workshop on EMOTION (satellite of LREC): Corpora for Research on Emotion and Affect*. Paris, France, 2010, p. 65.

[8] G. Zhao, X. Huang, M. Taini, S. Z. Li, and M. PietikäInen, "Facial expression recognition from near-infrared videos," *Image and Vision Computing*, vol. 29, no. 9, pp. 607–619, 2011.

[9] S. Li, W. Deng, and J. Du, "Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2852–2861.

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

[11] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon, "Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark," in *IEEE International Conference on Computer Vision Workshops*. IEEE, 2011, pp. 2106–2112.

[12] Z. Zhang, P. Luo, C.-C. Loy, and X. Tang, "Learning social relation traits from face images," in *IEEE International Conference on Computer Vision*, 2015, pp. 3631–3639.

[13] J. Kossaifi, R. Walecki, Y. Panagakis, J. Shen, M. Schmitt, F. Ringeval, J. Han, V. Pandit, A. Toisoul, B. W. Schuller *et al.*, "Sewa db: A rich database for audio-visual emotion and sentiment research in the wild," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

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

[15] J. Zeng, S. Shan, and X. Chen, "Facial expression recognition with inconsistently annotated datasets," in *The European Conference on Computer Vision*, September 2018.

[16] S. Li and W. Deng, "A deeper look at facial expression dataset bias," *IEEE Transactions on Affective Computing*, 2020.

[17] ——, "Deep emotion transfer network for cross-database facial expression recognition," in *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 3092–3099.

[18] F. A. M. da Silva and H. Pedrini, "Effects of cultural characteristics on building an emotion classifier through facial expression analysis," *Journal of Electronic Imaging*, vol. 24, no. 2, p. 023015, 2015.

[19] B. Hasani and M. H. Mahoor, "Facial expression recognition using enhanced deep 3d convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 30–40.

[20] B. Hasani and M. Mahoor, "Spatio-temporal facial expression recognition using convolutional neural networks and conditional random fields," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, 2017, pp. 790–795.

[21] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2016, pp. 1–10.

[22] M. Liu, S. Li, S. Shan, and X. Chen, "Au-inspired deep networks for facial expression feature learning," *Neurocomputing*, vol. 159, pp. 126–136, 2015.

[23] R. Zhu, G. Sang, and Q. Zhao, "Discriminative feature adaptation for cross-domain facial expression recognition," in *2016 International Conference on Biometrics (ICB)*. IEEE, 2016, pp. 1–7.

[24] Y. Ji, Y. Hu, Y. Yang, F. Shen, and H. T. Shen, "Cross-domain facial expression recognition via an intra-category common feature and inter-category distinction feature fusion network," *Neurocomputing*, vol. 333, pp. 231–239, 2019.

[25] M. V. Zavarez, R. F. Berriel, and T. Oliveira-Santos, "Cross-database facial expression recognition based on fine-tuned deep convolutional network," in *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. IEEE, 2017, pp. 405–412.

[26] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems*, 2018, pp. 1640–1650.

[27] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1426–1435.

[28] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 285–10 295.

[29] K. Fatras, T. Séjourné, R. Flamary, and N. Courty, "Unbalanced minibatch optimal transport; applications to domain adaptation," in *International Conference on Machine Learning*. PMLR, 2021, pp. 3186–3197.

[30] M. Li, Y.-M. Zhai, Y.-W. Luo, P.-F. Ge, and C.-X. Ren, "Enhanced transport distance for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 936–13 944.

[31] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7167–7176.

[32] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[33] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[34] X. Wei, H. Li, J. Sun, and L. Chen, "Unsupervised domain adaptation with regularized optimal transport for multimodal 2d+ 3d facial expression recognition," in *IEEE International Conference on Automatic Face & Gesture Recognition*. IEEE, 2018, pp. 31–37.

[35] X. Wang, X. Wang, and Y. Ni, "Unsupervised domain adaptation for facial expression recognition using generative adversarial networks," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[36] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[37] L. Liu, G. Li, Y. Xie, Y. Yu, Q. Wang, and L. Lin, "Facial landmark machines: A backbone-branches architecture with progressive representation learning," *IEEE Transactions on Multimedia*, vol. 21, no. 9, pp. 2248–2262, 2019.

[38] Y. Xie, T. Chen, T. Pu, H. Wu, and L. Lin, "Adversarial graph representation adaptation for cross-domain facial expression recognition," in *Proceedings of the 28th ACM international conference on Multimedia*, 2020.

[39] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "From facial expression recognition to interpersonal relation prediction," *International Journal of Computer Vision*, vol. 126, no. 5, pp. 550–569, 2018.

[40] Y.-Q. Miao, R. Araujo, and M. S. Kamel, "Cross-domain facial expression recognition using supervised kernel mean matching," in *International Conference on Machine Learning and Applications*, vol. 2. IEEE, 2012, pp. 326–332.

[41] E. Sangineto, G. Zen, E. Ricci, and N. Sebe, "We are not all equal: Personalizing models for facial expression analysis with transductive parameter transfer," in *ACM international conference on Multimedia*. ACM, 2014, pp. 357–366.

[42] H. Yan, "Transfer subspace learning for cross-dataset facial expression recognition," *Neurocomputing*, vol. 208, pp. 165–173, 2016.

[43] K. Yan, W. Zheng, Z. Cui, and Y. Zong, "Cross-database facial expression recognition via unsupervised domain adaptive dictionary learning," in *International Conference on Neural Information Processing*. Springer, 2016, pp. 427–434.

[44] W. Zheng, Y. Zong, X. Zhou, and M. Xin, "Cross-domain color facial expression recognition using transductive transfer subspace learning," *IEEE transactions on Affective Computing*, vol. 9, no. 1, pp. 21–37, 2016.

[45] W.-S. Chu, F. De la Torre, and J. F. Cohn, "Selective transfer machine for personalized facial expression analysis," *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 529–545, 2016.

[46] Y. Zong, W. Zheng, X. Huang, J. Shi, Z. Cui, and G. Zhao, "Domain regeneration for cross-database micro-expression recognition," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2484–2498, 2018.

[47] K. Yan, W. Zheng, T. Zhang, Y. Zong, C. Tang, C. Lu, and Z. Cui, "Cross-domain facial expression recognition based on transductive deep transfer learning," *IEEE Access*, vol. 7, pp. 108 906–108 915, 2019.

[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, Y. Bengio and Y. LeCun, Eds., 2015.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[50] T. Chen, L. Lin, L. Liu, X. Luo, and X. Li, "Disc: Deep image saliency computing via progressive representation learning," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 6, pp. 1135–1149, 2016.

[51] Z. Wang, T. Chen, G. Li, R. Xu, and L. Lin, "Multi-label image recognition by recurrently discovering attentional regions," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 464–472.

[52] T. Chen, Z. Wang, G. Li, and L. Lin, "Recurrent attentional reinforcement learning for multi-label image recognition," in *AAAI Conference on Artificial Intelligence*, 2017.

[53] T. Chen, L. Lin, X. Wu, N. Xiao, and X. Luo, "Learning to segment object candidates via recursive neural networks," *IEEE Transactions on Image Processing*, vol. 27, no. 12, pp. 5827–5839, 2018.

[54] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, 2017.

[55] Y. Li, D. Tarlow, M. Brockschmidt, and R. S. Zemel, "Gated graph sequence neural networks," in *International Conference on Learning Representations*, 2016.

[56] Z. Chen, X. Wei, P. Wang, and Y. Guo, "Multi-label image recognition with graph convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 5177–5186.

[57] T. Chen, L. Lin, R. Chen, Y. Wu, and X. Luo, "Knowledge-embedded representation learning for fine-grained image recog-

nition," in *Proc. of International Joint Conference on Artificial Intelligence*, 2018, pp. 627–634.
[58] T. Chen, M. Xu, X. Hui, H. Wu, and L. Lin, "Learning semantic-specific graph representation for multi-label image recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 522–531.
[59] R. Chen, T. Chen, X. Hui, H. Wu, G. Li, and L. Lin, "Knowledge graph transfer network for few-shot recognition," in *AAAI Conference on Artificial Intelligence*, 2020.
[60] T. Chen, W. Yu, R. Chen, and L. Lin, "Knowledge-embedded routing network for scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6163–6171.
[61] Z. Wang, T. Chen, J. Ren, W. Yu, H. Cheng, and L. Lin, "Deep reasoning with knowledge graph for social relationship understanding," in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 2018, pp. 2021–2018.
[62] C. Jiang, H. Xu, X. Liang, and L. Lin, "Hybrid knowledge routed modules for large-scale object detection," in *Advances in Neural Information Processing Systems*, 2018, pp. 1552–1563.
[63] T. Chen, L. Lin, X. Hui, R. Chen, and H. Wu, "Knowledge-guided multi-label few-shot learning for general image recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
[64] W. Yang, X. Wang, A. Farhadi, A. Gupta, and R. Mottaghi, "Visual semantic navigation using scene priors," in *International Conference on Learning Representations*, 2019.
[65] L. Liu, J. Chen, H. Wu, J. Zhen, G. Li, and L. Lin, "Physical-virtual collaboration modeling for intra-and inter-station metro ridership prediction," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
[66] L. Liu, Y. Zhu, G. Li, Z. Wu, L. Bai, M. Mao, and L. Lin, "Online metro origin-destination prediction via heterogeneous information aggregation," *arXiv preprint arXiv:2107.00946*, 2021.
[67] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
[68] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
[69] M. L. Junguang Jiang, Bo Fu, "Transfer-learning-library," https://github.com/thuml/Transfer-Learning-Library, 2020.
[70] E. Friesen and P. Ekman, "Facial action coding system: a technique for the measurement of facial movement," *Palo Alto*, vol. 3, 1978.
[71] Y.-I. Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 2, pp. 97–115, 2001.
[72] J. Wang, Y. Chen, L. Hu, X. Peng, and S. Y. Philip, "Stratified transfer learning for cross-domain activity recognition," in *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2018, pp. 1–10.
[73] J. Zhao, J. Li, X. Tu, F. Zhao, Y. Xin, J. Xing, H. Liu, S. Yan, and J. Feng, "Multi-prototype networks for unconstrained set-based face recognition," in *International Joint Conferences on Artificial Intelligence*, 2019.
[74] J. Wen, R. Liu, N. Zheng, Q. Zheng, Z. Gong, and J. Yuan, "Exploiting local feature patterns for unsupervised domain adaptation," in *AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 5401–5408.
[75] Y. Guo, L. Zhang, Y. Hu, X. He, and 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.
[76] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

Tianshui Chen received a Ph.D. degree in computer science at the School of Data and Computer Science Sun Yat-sen University, Guangzhou, China, in 2018. Prior to earning his Ph.D, he received a B.E. degree from the School of Information and Science Technology in 2013. He is currently the lecturer in the Guangdong University of Technology. His current research interests include computer vision and machine learning. He has authored and coauthored approximately 30 papers published in top-tier academic journals and conferences, including T-PAMI, T-NNLS, T-IP, T-MM, CVPR, ICCV, AAAI, IJCAI, ACM MM, etc. He has served as a reviewer for numerous academic journals and conferences. He was the recipient of the Best Paper Diamond Award at IEEE ICME 2017.

Tao Pu received a B.E. degree from the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China, in 2020, where he is currently pursuing a master's degree in computer science. His current research interests include computer vision and machine learning.

Hefeng Wu received a B.S. degree in computer science and technology and a Ph.D. degree in computer application technology from Sun Yat-sen University, China, in 2008 and 2013, respectively. He is currently a full research scientist at the School of Computer Science and Engineering, Sun Yat-sen University, China. His research interests include computer vision, multimedia, and machine learning. He has published works in and served as reviewers for many top-tier academic journals and conferences, including T-PAMI, T-IP, T-MM, CVPR, ICCV, AAAI, ACM MM, etc.

Yuan Xie received a B.E. degree in software engineering and a master's degree in computer science and technology from the School of Data and Computer Science Sun Yat-sen University, Guangzhou, China, in 2016 and 2019, respectively. She is currently a senior researcher at DMAI. Her research interests are in computer vision and human behavior analysis and their applications to human behavior analysis, human–robot interaction, and personalized learning.

Lingbo Liu is currently a postdoctoral fellow in the Department of Computing at the Hong Kong Polytechnic University. He received his Ph.D. degree in computer science from Sun Yat-Sen University in 2020. He was a research assistant at the University of Sydney, Australia. His current research interests include machine learning and intelligent transportation systems. He has authored and coauthored more than 15 papers in top-tier academic journals and conferences.

Liang Lin (M'09, SM'15) is a full professor at Sun Yat-sen University. From 2008 to 2010, he was a postdoctoral fellow at the University of California, Los Angeles. From 2016–2018, he led the SenseTime R&D teams to develop cutting-edge and deliverable solutions for computer vision, data analysis and mining, and intelligent robotic systems. He has authored and coauthored more than 100 papers in top-tier academic journals and conferences (e.g., 15 papers in TPAMI and IJCV and 60+ papers in CVPR, ICCV, NIPS, and IJCAI). He has served as an associate editor of IEEE Trans. Human-Machine Systems, The Visual Computer, and Neurocomputing and as an area/session chair for numerous conferences, such as CVPR, ICME, ACCV, and ICMR. He was the recipient of the Annual Best Paper Award by Pattern Recognition (Elsevier) in 2018, the Best Paper Diamond Award at IEEE ICME 2017, the Best Paper Runner-Up Award at ACM NPAR 2010, Google Faculty Award in 2012, the Best Student Paper Award at IEEE ICME 2014, and the Hong Kong Scholars Award in 2014. He is an IET Fellow.