IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 1, 2021, accepted October 12, 2021, date of publication October 15, 2021, date of current version October 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3120542

Cross-Domain Few-Shot Micro-Expression Recognition Incorporating Action Units

YI DAI, (Member, IEEE), AND LING FENG, (Senior Member, IEEE)
Centre for Computational Mental Healthcare, Department of Computer Science and Technology, Research Institute of Data Science, Tsinghua University, Beijing 100084, China

Corresponding authors: Yi Dai (daiy17@mails.tsinghua.edu.cn) and Ling Feng (fengling@mail.tsinghua.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61872214 and Grant 61521002.

ABSTRACT Micro-expression, different from ordinary facial expressions, is an involuntary, spontaneous, and subtle facial movement that reveals true emotions which people intend to conceal. As it usually occurs within a fraction of a second (less than 1/2 second) with a low action intensity, capturing micro-expressions among facial movements in a video is difficult. Moreover, when a micro-expression recognition system works in cold-start conditions, it has to recognize novel classes of micro-expressions in a new scenario, suffering from the lack of sufficient labeled samples. Inconsistency in micro-expression labeling criteria makes it difficult to use existing labeled datasets in other scenarios. To tackle the challenges, we present a micro-expression recognizer, which on one hand leverages the knowledge of facial action units (AU) to enhance facial representations, and on the other hand performs cross-domain few-shot learning to transfer knowledge acquired from other domains with different data labeling protocols and feature distribution to overcome the scarcity of labeled samples in the cold-starting scenario. In particular, we draw inspirations from the correlation between micro-expression and facial action units (AUs), and design an action unit module, aiming to extract subtle AU-related features from videos. We then fuse AU-related features and general features extracted by optical-flow facial images. Through fine-tuning, we transfer knowledge from datasets in different domains to the target domain. The experimental results on two datasets show that: (1) the proposed recognizer can effectively learn to recognize new categories of micro-expressions in different domains with a very few labeled samples with the UF1 score of 0.544 on CASME dataset, outperforming the state-of-the-art methods by 0.089; (2) the performance of the recognizer is more competitive when it distinguishes micro-expression videos of more categories; and (3) the action unit module enables to improve the recognition performance by 0.072 and 0.047 on CASME and SMIC, respectively.

INDEX TERMS Cross-domain few-shot learning, facial action unit, micro-expression recognition, multimodal feature fusion.

I. INTRODUCTION

Face is an ideal site to transmit information among different parts of the body, attributed to diverse, obvious, and quick facial muscle movements. Facial expressions refer to these facial movements that convey emotions and intentions of human [1]. Unlike ordinary facial expressions, facial micro-expressions occur within a fraction of a second with a low action intensity. Their involuntary emotional leakage usually expose true emotions and feelings, which people tend to hide. In some cases, even though people can deliberately pose false and misleading facial expressions, they could hardly hide their micro-expressions, which reveal their real emotional states [2]. Haggard and Isaacs [3] once played video records of conversation between a patient and a psychotherapist at a slow rate, spotting transient micro-expression of grimace between patient's smiles.

Due to the true emotions revealed by natural and involuntary micro-expressions, micro-expression recognition technologies have a wide scope of applications in the fields, such as psychological and clinical diagnosis, criminal investigation, judicial judgment, etc.

In the literature, substantial efforts have been made on micro-expression recognition. Majority of the work pay attention to the process of feature extraction. Given a facial image or a video, the micro-expression recognition system

The associate editor coordinating the review of this manuscript and approving it for publication was Kathiravan Srinivasan.

needs to generate feature representations with a specific feature extraction method. Feature representations help the recognition system summarize features of raw data, discarding irrelevant information to the recognition task. Based on the generated representations, the data samples can further be classified into several micro-expression categories by the system.

Traditional micro-expression recognition methods (e.g., Local Binary Pattern histograms from Three Orthogonal Planes (LBP-TOP) [4] and Bi-Weighted Oriented Optical Flow (Bi-WOOF) [5]) extract hand-crafted features from raw data, mapping micro-expression videos to a feature space. These features are then classified by a classifier like support vector machine (SVM). With the recent development of various deep neural networks (e.g., Visual Geometry Group (VGG) [6], AlexNet [7] and ResNet [8]), deep neural networks are adopted in micro-expression recognition [9]–[16]. As deep neural networks (DNN) can acquire deep features from input samples, DNN-based micro-expression recognition methods significantly outperform the traditional methods. However, DNN-based techniques need large-scale annotated datasets to train feature extractors and classifiers. Otherwise, the recognition models tend to overfit the samples provided, resulting in poor classification performance.

Despite much progress in micro-expression recognition mentioned above, few applications have been implemented up till now. Confronted with two main challenges, it is difficult to widely use micro-expression recognition technologies in real-life scenarios. First, as the duration of a micro-expression is short, and its occurrence is relatively rare, we could capture a very limited number of micro-expression samples from a large amount of facial videos. Hence, features learnt by micro-expression recognition systems are limited, and it is challenging to provide a method of high recognition accuracy. Second, in real-life scenarios, micro-expression recognition systems are under a cold-start problem. They have to recognize micro-expression videos of unseen classes with a few labeled samples to learn from. As large-scale micro-expression datasets only contain samples of basic emotion categories for the sake of universality, recognition system trained with such datasets cannot recognize task-specific emotion categories. For example, panic, anger and anxiety may appear in prisons, which are unseen when training a recognition system. Meanwhile, as it is labor-intensive and time-consuming to build a micro-expression dataset of these new micro-expression categories from scratch, training a recognition system with task-specific datasets is not feasible, only few labeled samples can be available. In other tasks under cold-start conditions, transfer learning [17] methods are introduced, using datasets of other scenarios to augment training datasets. However, it is also challenging to use datasets available. Since the protocols of micro-expression data collection are not unified [18], [19], the datasets introduced could be quite different from micro-expression samples the model needs to recognize (e.g., things eliciting micro-expressions). Most important of all, the categories of micro-expressions could be quite different in datasets introduced in the application scenario. For example, the auxiliary diagnostic system for psychotherapists needs to recognize *repression*, *despair*, and *anxiety*, while extreme emotions like *anger* could be prioritized for prison management systems.

To tackle the first challenge, we leverage knowledge of facial action units (AUs) to strengthen facial representations for micro-expression recognition. It is inspired by the significant correlation between micro-expressions and facial action units, as well as established research on facial action units. AUs are a set of objective labels describing facial muscle movements, and they are related to different facial regions. Micro-expressions of a specific emotion category are corresponding to certain groups of AUs. For example, facial expression of happiness includes cheek raiser (AU6) or lip corner puller (AU12) [20], while sadness includes inner brow raiser (AU1) [21]. Therefore, we can enrich features learnt from raw data by incorporating AU-related features. Furthermore, as AUs are region-specific, AU-related features can guide the model to place more emphasis on local regions posing significant influence on emotion expression. Previous studies [20]–[25] have shown that a certain group of AUs cannot determine micro-expression category solely, i.e., two different micro-expression may share the same group of AUs. Thus, the feature extractor in our proposed model not only extracts AU-related features from raw data, but also considers general features extracted from optical-flow facial images.

For the cold-start challenge, we perform cross-domain few-shot learning, which is getting researchers' attention recent years. There are areas related to cross-domain few-shot learning, which also learn unseen classes from streaming data. Continual learning [26]–[29] requires a model to learn new tasks sequentially and avoid forgetting former knowledge catastrophically. Active learning enables models to interactively select samples to be labeled by specialists or other sources [30]–[32]. Different from these studies, cross-domain few-shot learning focuses on the data scarcity problem of the new task. Two methods (fine-tuning and metric-based few-shot learning method) are adopted to enable the model to acquire knowledge from datasets available in other scenarios (source domain), and then transfer the knowledge to the scenario where it works (target domain), recognizing novel classes with only a few labeled samples.

A. KEY CONTRIBUTIONS

The main contributions of the study are summarized as the following:

- We propose Micro-expression Recognizer incorporating Action Units (MERAU), which incorporates knowledge of facial action units, and effectively learns to recognize new categories of micro-expressions in a different domain with a few labeled samples. We are the first to combine AU-related features with general features extracted from optical-flow in micro-expression recognition.

- To handle cold-start problems in potential applications of micro-expression recognition, we perform cross-domain few-shot learning for micro-expression recognition.
- We propose that incorporating AU-related features in feature extraction can help the model better differentiate samples of different categories at the representation level.

The remainder of this paper is structured as follows. In Section II, we summarize studies related to our work. In Section III, our proposed micro-expression recognition framework is introduced, along with two different learning methods applied to this framework. Section IV describes the details of our experiments and experimental results. We conclude our work and point out potential research directions in Section V.

II. RELATED WORK

In this section, we review relevant literature on micro-expression recognition. Additionally, studies on facial action unit detection, cross-domain few-shot learning, and multi-modal fusion techniques are also summarized.

A. MICRO-EXPRESSION RECOGNITION

Micro-expression recognition methods can be categorized into *statistical methods* and *deep learning methods* by forms of feature extraction. Statistical methods adopt hand-crafted feature extraction to describe the characteristics of micro-expression videos, aiming to transform original data into statistic features. A representative statistical method, LBP-TOP [33], is used as baseline in Facial Micro-Expressions Grand Challenge (MEGC) 2018 [34] and MEGC 2019 [18].

In comparison, deep learning methods utilize deep neural networks to extract features from micro-expression videos. As deep convolutional networks are powerful for extracting discriminative features from original data, deep learning methods outperform statistical methods in most micro-expression recognition scenarios. So far, a lot of deep learning methods have been developed for recognizing micro-expressions with deep neural networks [9]–[16].

Some studies such as [9] considered all frames in a video when extracting features, increasing the computational complexity at the same time. Nevertheless, Liong *et al.* [5] found out that not all frames are necessary for providing adequate information, and prompted the use of only onset and apex frames of a video instead.

Previous studies also considered recognizing micro-expression with the aid of facial action units. After extracting features with 3D ConvNet, Xie *et al.* [35] transformed those features into a feature map for building an AU graph. A Graph Convolutional Network (GCN) was then used to process AU node features and provide information for micro-expression recognition. Unlike this work which only relied on AU-related features for micro-expression recognition, we integrate AU-related features with general features extracted from optical-flow images in micro-expression recognition based on the previous studies [20]–[25], which showed that a certain group of AUs cannot distinguish micro-expression categories well.

Recent work has considered practicality of micro-expression recognition systems in the real world. Li *et al.* [36] handled small training dataset of micro-expression by using neighbouring frames of apex frame, Lai *et al.* [37] and Hashmi *et al.* [38] focused on real-time micro-expression recognition, proposing end-to-end micro-expression recognition systems.

B. FACIAL ACTION UNIT DETECTION

Studies on facial action units detection can also be divided into two main categories: *AUs occurrence detection* [39]–[43] and *AUs intensity estimation* [44]–[48]. AUs occurrence detection intends to recognize the occurrence of each AU, transforming AUs detection into a multi-labeled binary classification problem. In comparison, AUs intensity estimation considers not only the presence but also exact intensity levels of AUs, i.e., from 1 to 5.

Early research on facial action unit detection used features of the whole face with hand-crafted feature extraction methods [49]. Since each AU is related to a certain facial region, sparsity-induced methods were then introduced into AU detection, reducing interference from irrelevant regions. For instance, Zhao *et al.* [41] proposed a region layer. Instead of sharing weights across the entire image, the region layer has local convolution components for different facial regions, thus enabling the model to capture local appearance changes. Li *et al.* [42] attached E-Net and C-Net to a conventional deep convolutional network. E-Net places more emphasis on active regions related to AUs with an attention mechanism, and C-Net crops AU areas of interest.

Due to the difficulty in AU labeling, some studies intend to reduce dependence on manual annotation, focusing on weak-supervised or self-supervised AUs detection to reduce dependence on labeled samples. Weak-supervised studies do not need correct and exact labels from human annotation. Zeng *et al.* [40] proposed a weak-supervised learning method based on confidence. Zhang *et al.* [47] used prior knowledge that AU intensity increases monotonically between the onset frame and apex frame during a facial action. Self-supervised learning generates supervisory information from unlabeled data, using its own structure. Twin-Cycle Autoencoder [39] disentangled AU related movements from head motion related ones in videos. This model was trained with facial image pairs of the same person in videos. With the absence of manual annotation, the model learned to recognize displacements of pixels between the source image and the target image. Thus, the model can be optimized with the reconstruction loss.

C. CROSS-DOMAIN FEW-SHOT LEARNING

Few-shot learning is an important subproblem of machine learning. It aims to improve performance of models on a

specific task with knowledge acquired from a few labeled samples [50]. In many real-life scenarios, due to the lack of labeled samples for training, models are likely to overfit and perform poorly on testing sets. Therefore, researchers have proposed a series of methods to tackle this problem. For example, ProtoNet [51] computes the mean of samples in each category as prototypes in the feature space. In this way, a test sample can be classified by computing its distance to each prototype, and a closer distance indicates higher possibility of belonging. Siamese Network [52] embeds a sample pair into the feature space with an identical neural network, and applies a binary classifier to indicate whether the pair of samples belong to the same category. Unlabeled test samples can thus be classified by comparing them with the labeled samples in each category.

Cross-domain learning, also known as domain adaptation, requires models to solve problems in a target domain, only utilizing knowledge learnt from a source domain. However, samples in these two domains have different feature distribution. Since few-shot learning methods tend to use knowledge from other domains as supplementary knowledge, few-shot learning and cross-domain learning tasks are highly correlated and should be considered together [53]. A number of recent studies [54]–[57] try to address cross-domain few-shot problems, incorporating knowledge learnt from source domains. Chen *et al.* [57] addressed cross-domain few-shot problem in generic object recognition and fine-grained image classification. Two different fine-tuning methods are implemented, as well as several metric-based few-shot learning methods. Their experimental results show surprisingly competitive performance of fine-tuning methods. Inspired by their work, we introduce fine-tuning and metric-based few-shot learning methods into micro-expression recognition.

D. MULTIMODAL FUSION

Modalities are information presented in different forms or collected from different sources. To explore human emotions, studies have been conducted about different modalities, including audio [58]–[60], electroencephalography [61], [62], and imagery [9]–[14], [16], [33], [35]. By combining multimodal information, models can obtain richer features and have better understanding of the samples. Hence, multimodal fusion approaches are widely used, like fusing features extracted from texts and images [63], [64], or images presenting different information [65]–[67].

Multimodal fusion can be classified based on the fusion time [68]. Late fusion methods fuses multimodal features at the decision level, providing independent models for different modalities that do not interfere with each other [69]. Early fusion fuses at the feature level. Li *et al.* [66] concatenated three channels of a RGB image with two channels of the optical flow image before feature extraction. In this study, we take the early fusion strategy to integrate AU features with the ones extracted from optical flow images as the final embedding of the raw video in the feature space.

III. PROBLEM DEFINITION AND METHODOLOGY

A. PROBLEM DEFINITION

Given a user's frame sequence containing an onset frame and an apex frame, denoted as $x = (s_{onset}, s_{apex})$, our task is to identify his/her micro-expression y in the category set E_t based on x. In the study, we consider two different category sets, i.e., $E_t = \{Tense, Repression, Disgust, Surprise\}$ | $\{Positive, Negative, Surprise\}$. Let X_t be the set of onset and apex frame pairs.

Assume we only have a limited number of K labeled samples for each target class among E_t, while the ramaining samples in X_t are left unlabeled. If the model is merely trained on these samples, it can hardly obtain knowledge of micro-expression, which will result in the poor performance when testing. Thus we intend to acquire knowledge from labeled samples in datasets available from other scenarios, referred to as *source domain*, and the samples in the scenario we cold-start are referred to as *target domain*, we cast the problem definition to a cross-domain few-shot learning setting.

Let E_s denote the set of source categories, and E_t is the set of target categories, where $(E_s \neq E_t)$ and $(|E_s| \neq |E_t|)$. X_s and X_t denote frame pairs in *source domain* and *target domain* respectively. Labeled samples in *target domain* are denoted as $D_{support} = \{(x_1^s, y_1^s), \cdots, (x_{ns}^s, y_{ns}^s)\}$, where $(ns = K \cdot |E_t|)$, $(x_i^s \in X_t)$, and $(y_i^s \in E_t)$ (for $i = 1, \cdots, ns$). This few-shot problem is also known as an $|E_t|$-Way-K-Shot problem. The remaining unlabeled samples in *target domain* are denoted as $D_{test} = \{(x_1^q, y_1^q), (x_2^q, y_2^q), \cdots, (x_{nq}^q, y_{nq}^q)\}$, which are to be queried and detected.

We use $D_{train} = \{(x_1^t, y_1^t), (x_2^t, y_2^t), \cdots, (x_{nt}^t, y_{nt}^t)\}$ to denote nt labeled samples in the *source domain*, where $(nt \gg ns)$, $(x_i^t \in X_s)$, and $(y_i^t \in E_s)$ (for $i = 1, 2, \cdots, nt$).

Furthermore, to incorporate knowledge of AUs, we utilize another AU-labeled dataset D_{au}. It shares the same set of frame pairs X_s with D_{train}, yet has a different annotation format from the micro-expression datasets D_{train}, $D_{support}$, and D_{test}. Let $D_{au} = \{(x_1^a, \overline{y}_1^a), (x_2^a, \overline{y}_2^a), \cdots, (x_{na}^a, \overline{y}_{na}^a)\}$, where na is the number of samples in the AU dataset, and for each $(x_i^a, \overline{y}_i^a) \in D_{au}$, $\overline{y}_i^a$ is a 10-dimensional scalar value vector, signifying the existence of ten typical action units (Inner Brow Raiser, Outer Brow Raiser, Brow Lower, Lid Tightener, Nose Wrinkler, Upper Lip Raiser, Lip Corner Puller, Dimpler, Lip Corner Depressor, Chin Raiser) in $x_i^a \in X_s$. Here, value 1 represents the existence, and 0 otherwise.

B. OVERALL FRAMEWORK

The presented Micro-Expression Recognition framework incorporates Action Units (MERAU) to cross-domain few-shot micro-expression classification. As shown in Figure 1, MERAU consists of two modality feature extractors (named AU module and Optical-flow module) and a classifier. Optical-flow module aims to acquire optical flow information from the onset and apex frames of a video with an encoder, and maps it to low-dimensional feature space with a projection layer. AU module extracts AU-related information

from the apex frame of the video, and transforms it into two different feature embeddings. The three feature embeddings generated by Optical-flow module and AU module are then concatenated as the final embedding of the raw video in the feature space. MERAU implants two different ways of learning (fine-tuning and metric-based few-shot learning) to project the final embedding of the raw video into the label space, detecting the micro-expression category of the facial video given.

1) AU MODULE

We utilize Twin-Cycle Autoencoder (TCAE) [39] as the encoder of AU module. For a frame sequence, we only feed its apex frame s_{apex} into this encoder. The output of TCAE encoder $\boldsymbol{x}_{au}$ is then fed into an AU detector P pretrained with an AU dataset D_{au}. The detector transforms the AU-related features into AU prediction $\boldsymbol{p} = [\omega_1, \omega_2, \cdots, \omega_A]$ with $\frac{1}{1+e^{-\omega_i}}$ as the possibility that s_{apex} has the i-th action unit.

Since AU prediction feature $\boldsymbol{p}$ is obtained with additional supervision (AU pretraining), it may have different distribution from AU-related feature $\boldsymbol{x}_{au}$. In order to fuse $\boldsymbol{x}_{au}$ and $\boldsymbol{p}$, We use two projection layers with ReLU activation function to project them into the same feature space, separately. Meanwhile, the projection layer for $\boldsymbol{x}_{au}$ transforms it into low-dimensional vectors, extracting task-related information. The projections in the feature space are denoted by $\boldsymbol{e}_{a1}$ and $\boldsymbol{e}_{a2}$.

To incorporate knowledge of AU detection into our model, the AU detector $\mathcal{P}$ needs to be pretrained with the AU-labeled samples in D_{au}. For a sample x in D_{au}, $\mathcal{P}_\phi(x) \in [0, 1]^A$ represents possibilities of occurrence of all AU labels. We thus compute AU loss $\mathcal{L}_{au}$ as follows:

$$\mathcal{L}_{au} = -\frac{1}{A}\Sigma_{i=1}^{A} y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot (1 - \log \hat{y}_i) \quad (1)$$

Then we can achieve optimized parameters of $\mathcal{P}$, denoted by

$$\phi = argmin_\phi(\mathcal{L}_{au}(D_{ext}; \phi)) \quad (2)$$

Note that ϕ will be frozen in the follow-up micro-expression training and testing.

2) OPTICAL-FLOW MODULE

As the category of micro-expression is not determined by AU-related information solely, we compute optical flow images using the onset and apex frames, which describe geometric deformations of facial videos, and then feed them into an Optical-flow module. We take ResNet18 [8] as the encoder of Optical-flow module, and use Gunnar Farneback Algorithm [70] to generate the dense optical flow as the input of Optical-flow module. We intend to map the high-dimensional feature embedding $\boldsymbol{x}_{of}$ obtained by Optical-flow module, to the same feature space of $\boldsymbol{e}_{a1}$ and $\boldsymbol{e}_{a2}$, and fuse three features.

Hence, we use a projection layer with ReLU activation function to transform $\boldsymbol{x}_{of}$ into a low-dimensional feature embedding $\boldsymbol{e}_{of}$ as optical-flow feature of the video.

3) CLASSIFIER

We use $\mathcal{M}$ to denote the combination of AU module and Optical-flow module. For each sample input, the feature embedding given by $\mathcal{M}$ is the concatenation of three feature vectors, which can be denoted as:

$$\boldsymbol{e} = \boldsymbol{e}_{of} \oplus \boldsymbol{e}_{a1} \oplus \boldsymbol{e}_{a2} \quad (3)$$

The classifier $\mathcal{C}$ then projects e into the label space, predicting the micro-expression category of facial videos given. Here, we adopt two different learning methods (i.e., fine-tuning and metric-based few-shot learning) to perform classification.

We use D_{train} to train the feature embedding model $\mathcal{M}$. $\mathcal{M}$ transforms samples into low-dimensional feature embeddings $\boldsymbol{e}$, the process can be denoted by $\boldsymbol{e} = \mathcal{M}_{\theta,\phi}(x)$, where ϕ is a freezed parameter of AU detector P, and θ is a trainable parameter of $\mathcal{M}$. Based on the label space of the dataset, classifier $\mathcal{C}$ transforms e into a category label, represented by $\boldsymbol{p} = C(\boldsymbol{e})$.

We can denote the combination of feature embedding model $\mathcal{M}$ and classifier $\mathcal{C}$ by a function $f_{\theta,\phi}(\boldsymbol{x}) = \boldsymbol{y}$, use loss function $\mathcal{L}_{exp}$ to train $\mathcal{M}$, and obtain:

$$\theta = argmin_\theta \, \mathcal{L}_{exp}(D_{train}; \theta; \phi) \quad (4)$$

Note that the detailed form of loss function $\mathcal{L}_{exp}$ depends on the learning method we use.

a: FINE-TUNING

Fine-tuning method uses a fully-connected layer as classifier. It has the weight of $\boldsymbol{W} \in \mathbb{R}^{d \times |E_s|}$ at the training stage, where d denotes the dimension of feature embedding $\boldsymbol{e}$, and E_s is the set of micro-expression categories in D_{train}. The classifier is trained together with Optical-flow module and AU module. While in $D_{support}$ and D_{test}, only the parameters of feature embedding model $\mathcal{M}$ are kept, and the weight matrix of $\mathcal{C}$ is re-initialized to $\boldsymbol{W} \in \mathbb{R}^{d \times |E_t|}$, where E_t is the set of micro-expression categories in $D_{support}$ and D_{test}.

The training and fine-tuning process are shown in Figure 2. For a basic classifier, when we feed feature embedding $\boldsymbol{e}$ into classifier $\mathcal{C}$, the output is

$$\hat{\boldsymbol{y}} = \boldsymbol{W}^T \boldsymbol{e} \quad (5)$$

Additionally, following the setting proposed by Chen *et al.* [57], we implement the fine-tuning method with a cosine-distance based classifier. The weight matrix of $\mathcal{C}$ is $\boldsymbol{W} \in \mathbb{R}^{d \times |E_s|}$, which is the concatenation of $|E_s|$ vectors, $[\boldsymbol{w_1}, \boldsymbol{w_2}, \cdots, \boldsymbol{w}_{|\boldsymbol{E_s}|}]$. When a feature embedding is fed into the classifier, the output is:

$$\hat{\boldsymbol{y}}_{cd} = [sim(\boldsymbol{e}, \boldsymbol{w_1}), \cdots, sim(\boldsymbol{e}, \boldsymbol{w_i}), \cdots, sim(\boldsymbol{e}, \boldsymbol{w}_{|\boldsymbol{E_s}|})] \quad (6)$$

where sim is a cosine distance function. Given two vectors $\boldsymbol{e}$ and $\boldsymbol{w}$, the output is computed as:

$$sim(\boldsymbol{e}, \boldsymbol{w}) = \frac{\boldsymbol{e}^T \boldsymbol{w}}{\|\boldsymbol{e}\| \, \|\boldsymbol{w}\|} \quad (7)$$

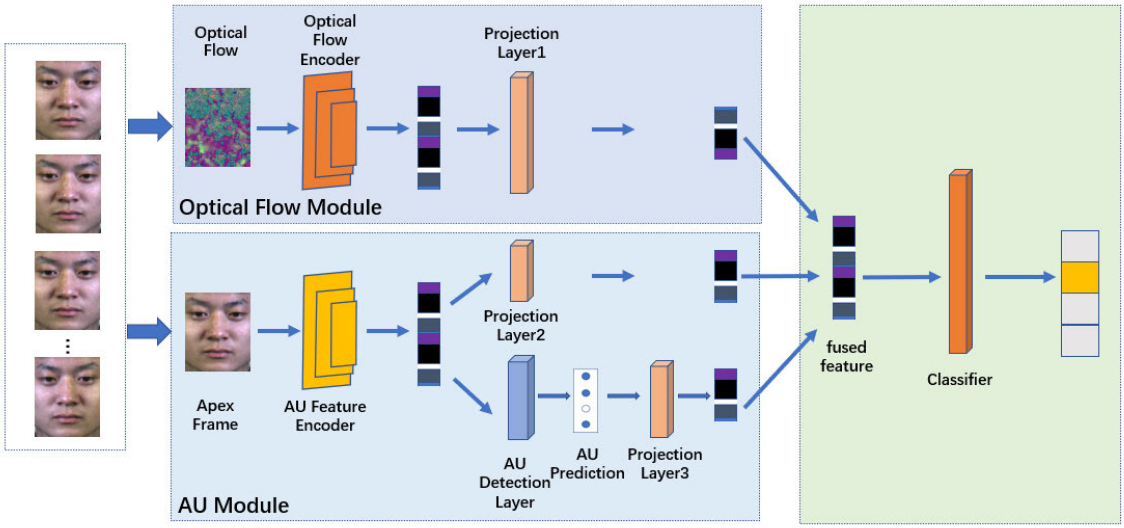


FIGURE 1. Micro-expression recognizer incorporating action units (MERAU).

Similar to the basic classifier, the cosine-distance based classifier is parameterized by $\boldsymbol{W} \in \mathbb{R}^{d\times|E_t|}$ at the testing stage.

For these two classifiers, we use the same cross entropy loss function as follows:

$$\mathcal{L}_{exp} = -\Sigma_{i=1}^{N} y_i log(\hat{y}_i) \tag{8}$$

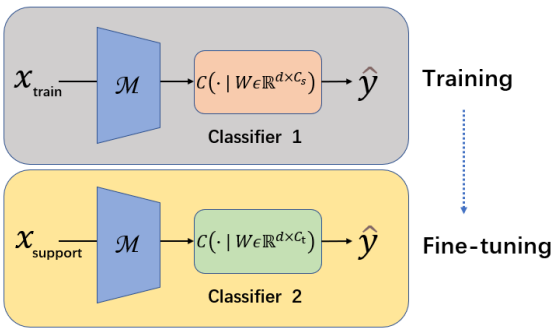


FIGURE 2. The process of training and fine-tuning.

b: METRIC-BASED FEW-SHOT LEARNING

Metric-based few-shot learning uses distance metrics to differentiate between samples in a dataset. We implement ProtoNet [51], a typical and effective metric-based few-shot learning method. It computes the mean of samples in each category as prototypes, and compares Euclidean distance between feature embeddings of query samples and prototypes. The core of ProtoNet method in micro-expression is to grasp the representative prototypes of each micro-expression category in the feature space. Despite the lack of labeled samples in target domain, the model learns how to generate micro-expression feature prototypes with samples in the source domain. In the target domain, the model directly generates prototypes without learning.

We assume that there are $|E_t|$ categories of micro-expressions in D_{test}, and each category has K labeled samples in $D_{support}$. At the training stage, instead of using all labeled samples in training dataset D_{train}, we pick samples in only categories E_t from D_{train}, and split them into Support Set $\{S_1, \cdots, S_{|E_t|}\}$ and Query Set $\{Q_1, \cdots, Q_{|E_t|}\}$, where S_i and Q_i denote support samples and query samples of category i, respectively.

We group these support and query samples into different episodes. For each episode, we select K labeled samples from support samples S_i for each category i, as episode support samples, and select T unlabeled samples from Q_i as episode query samples. Thus, an episode contains a total of $|E_t| \cdot (K + T)$ samples.

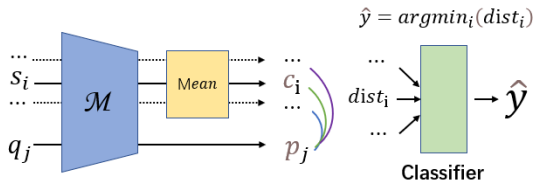


FIGURE 3. The process of classification in ProtoNet [51].

As shown in Figure 3, samples are first transformed into embeddings in the feature space by $\mathcal{M}$, for each category, embeddings of all episode support samples are averaged into prototypes of the category, denoted by $\boldsymbol{c}$.

The model classifies the category of each episode query sample q_j by comparing it with all prototypes:

$$\hat{y} = \underset{j\in\{1,\cdots,|E_t|\}}{argmin_j}\ dist(\boldsymbol{q_i}, \boldsymbol{c_j}) \tag{9}$$

where dist$(\cdot,\cdot)$ is the function to compute Euclidean distance between embeddings. Note that for two embeddings $\boldsymbol{x}$ and $\boldsymbol{y}$, $\text{dist}(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|^2$.

More details about the sample selection and parameter optimization can be found in Algorithm 1. To optimize parameters of feature embedding model $\mathcal{M}$, we use a cross entropy loss based on distance:

$$\mathcal{L}_{exp} = -d(\boldsymbol{c_j}, \boldsymbol{p_i}) - \log \Sigma_{k=1}^{|E_t|} e^{-d(\boldsymbol{c_k}, \boldsymbol{p_i})} \tag{10}$$

where $\boldsymbol{c_j}$ is the prototype of category j.

Algorithm 1 Training With ProtoNet. $|E_t|$ Is the Number of Categories and K Is the Number of Support Samples per Class We Select in an Episode. The Feature Embedding Model $\mathcal{M}$ Is Trained for N Episodes

Require: Support set $\{S_1, \cdots, S_{|E_t|}\}$
Require: Query set $\{Q_1, \cdots, Q_{|E_t|}\}$
Require: Feature embedding model $\mathcal{M}$
Initialize trainable parameters θ of feature embedding model $\mathcal{M}$
for $e \leftarrow 1$ to N **do**
  for $i \leftarrow 1$ to $|E_t|$ **do**
    $V_s \leftarrow sample(S_i, K)$
    $P \leftarrow \varnothing$
    for support sample $s \in V_s$ **do**
      Append $\mathcal{M}_{\theta,\phi}(s)$ to P
    end for
    $c_i \leftarrow \bar{P}$
  end for
  for $i \leftarrow 1$ to $|E_t|$ **do**
    for query sample $q \in Q_i$ **do**
      $p \leftarrow \mathcal{M}_{\theta,\phi}(q)$
      Calculating $\mathcal{L}_{exp}$ with Equation 10
      Update θ with $\nabla\mathcal{L}_{exp}$
    end for
  end for
end for

IV. EXPERIMENTS AND DISCUSSION

In this section, we present our experimental setting, including baseline methods, datasets we used, pre-processing methods, and evaluation protocols. Experimental results are reported and analyzed as well.

A. DATASETS AND PRE-PROCESSING

We conduct experiments on three micro-expression datasets, including SMIC [71], CASME [21], and CASMEII [20]. SMIC dataset has 164 micro-expression videos collected from 16 subjects. It contains three coarse-grained categories of emotion labels: *positive, negative*, and *surprise*. CASME dataset has 196 samples classified into 8 fine-grained classes. CASME II dataset is larger than CASME, containing 255 samples of 7 categories. We screen out categories of few samples from CASME and CASMEII. The remaining categories of the three datasets are listed in Table 1.

As CASME and CASME II have similar categories, as well as settings of dataset construction, we train our model on CASME II, and test it on CASME and SMIC in order to evaluate its performance with different scales of domain-shift.

Besides data augmentation strategies such as random cropping, resizing, and rotation, we expand our training dataset by using neighboring frames of the apex frames in micro-expression videos based on the previous studies [36], [47], which assert that neighboring frames have similar facial appearance and emotion expression. Thus, neighboring frames share the same micro-expression categories as apex frames.

Meanwhile, to balance the training samples of different categories, for each apex frame in videos labeled with micro-expression category i, we calculate the number of neighboring frames (aug_i) to be added to the training dataset as follows:

$$aug_i = max(round(\frac{5\dot{N}_{min}}{N_i}) - 1,\ 0) \tag{11}$$

where N_i is the number of samples in category i, and N_{min} is the minimum among all N_i.

B. IMPLEMENTATION DETAILS

1) COMPARISON METHODS

We implement two state-of-the-art methods: Quang's CapsuleNet [12] for micro-expression recognition, and Liu's micro-expression recognizer [11]. Additionally, several benchmarks methods are implemented, including LBP-TOP with uniform code [33] and VGG [6].

LBP-TOP is a hand-crafted micro-expression recognition method. Liu's work extracts features from optical-flow images, while Quang's CapsuleNet and VGG use apex frames as input images.

For a fair comparison, we apply the same data augmentation method to these baselines.

2) PRE-TRAINING

To incorporate knowledge of AUs into the model, we pre-train our AU module before the training stage. We adopt a pre-trained Twin-Cycle Autoencoder [39] as encoder in the AU module, and freeze its parameters. The learning rate of AU detector P in the AU module is set to 0.0012.

For each apex frame in CASME II dataset, there is a group of labels, indicating AUs that appear on the face. We use 10 AUs in CASME II dataset for pre-training. The process is shown in Figure 4.

3) CROSS-DOMAIN FEW-SHOT LEARNING

After the pre-training stage, we conduct cross-domain few-shot learning with fine-tuning and ProtoNet. The parameters of AU encoder and AU detector are frozen. We list the learning rates for other layers in MERAU and baseline models in Table 2.

TABLE 1. Label summary of the three datasets.

	Tense	Repression	Disgust	Surprise	Happiness	Others	Positive	Negative
CASME II	-	27	63	25	32	99	-	-
CASME	66	38	44	20	-	-	-	-
SMIC	-	-	-	43	-	-	51	70

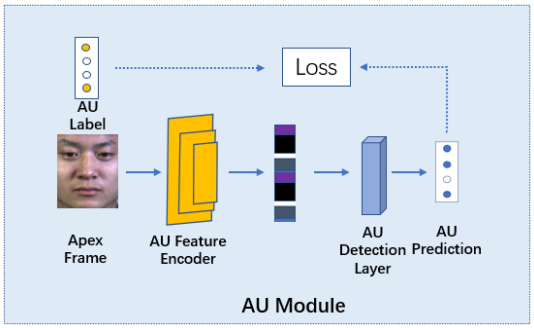


FIGURE 4. Pre-training of the AU module.

TABLE 2. Learning rates of MERAU and baseline models under fine-tuning and ProtoNet.

Layers		Fine-tune		ProtoNet
		Train	Fine-tune	Train
MERAU	OF encoder	1e-5	1e-5	1e-5
	projection layers	1.2e-3	1e-3	1e-4
	classifier	1.2e-3	2e-3	-
Liu's	ResNet18 encoder	1e-5	1e-5	1e-5
	classifier	1e-3	1e-3	-
VGG		1e-4	1e-4	5e-6
CapsuleNet		1e-4	1e-4	1e-5

C. EVALUATION

We evaluate the performance of our MERAU and baseline methods on CASME under the setting of 4-Way-5-Shot and 2-Way-5-Shot. As we adopt basic classifier and cosine-distance based classifier for fine-tuning, our method using these two classifiers are named MERAU and MERAU (CD), respectively.

For 4-Way-5-Shot setting, we select all four categories remaining in the CASME dataset. Meanwhile, for 2-Way-5-Shot, we divide samples in CASME into two groups: (1) Easy group and (2) Hard group. The Easy group contains micro-expression videos labeled with *Disgust* and *Surprise*, while the Hard group contains samples labeled with *Tense* and *Repression*. Samples in the Hard group are relatively more difficult to differentiate, since both *Tense* and *Repression* are negative feelings in coarse-grained classification. We will quantitatively verify this assumption in the next subsection. To avoid overfitting certain categories of samples, besides accuracy (ACC), unweighted F1-scores (UF1) and unweighted average recall (UAR) are chosen as performance metric of MERAU and baseline models, presented in Table 3. we use TP, TN, FP and FN to denote true positives, true negatives, false positives and false negatives, as there are N categories of samples, UF1, UAR and ACC can be calculated as:

$$UF1 = \Sigma_{i=1}^{N} \frac{2 \cdot TP_i}{2 \cdot TP_i + FP_i + FN_i} / N \tag{12}$$

$$UAR = \Sigma_{i=1}^{N} \frac{TP_i}{TP_i + FN_i} / N \tag{13}$$

$$ACC = \frac{\Sigma_{i=1}^{N}(TP_i + TN_i)}{\Sigma_{i=1}^{N}(TP_i + TN_i + FP_i + FN_i)} \tag{14}$$

In addition, to evaluate performance of MERAU with a larger domain-shift, we test our MERAU and all baseline methods on SMIC, which has only 3 coarse-grained categories. The results are shown in Table 4.

As illustrated, our MERAU outperforms all baseline methods confronted with both shallow domain-shift (CASME II to CASME) and large domain-shift (CASME II to SMIC) on all three metrics. Under the same setting (e.g., 4-way-5-shot on CASME), performance of MERAU and baselines are consistent when using different metrics. Despite the fact that micro-expression recognizer has to differentiate among samples of more categories on CASME than on SMIC, the recognition accuracies are significantly higher on CASME. This is because larger domain-shift increases the difficulty in transferring knowledge learnt from the source domain to a new task on the target domain. The *Positive* and *Negative* categories of SMIC dataset have never appeared in CASME II, while most of the categories are shared between CASME II and CASME. In addition, subjects and data collection criteria are quite different from CASME II to SMIC. This assumption can be verified through 2-Way-5-Shot experiments we conduct, where the performance of all methods on Easy group is significantly better.

Furthermore, to quantify domain-shifts, we generate features of samples from source domain (CASME II) and two different target domains (CASME and SMIC) with a pre-trained encoder. The encoder is a Resnet18, which has no prior knowledge about three datasets (SMIC, CASME II and CASME), in order to avoid interference. We then compute Maximum Mean Discrepancies (MMD) [72] between source domain and two different target domains, as follows.

$$MMD(\mathcal{F}, S, T) = \sup_{f \in \mathcal{F}} \left(\frac{\Sigma_{i=1}^{N_t} \boldsymbol{x_i^t}}{N_t} - \frac{\Sigma_{i=1}^{N_s} \boldsymbol{x_i^s}}{N_s} \right) \tag{15}$$

Here, $\mathcal{F}$ is the unit ball in a reproducing kernel Hilbert space, S and T denote source domain and target domain.

TABLE 3. Few-shot evaluation with CASME on ACC, UF1 and UAR.

Setting (CASME)		4-Way-5-Shot		2-Way-5-Shot (Easy)		2-Way-5-Shot (Hard)	
Metric	Method	Finetune	ProtoNet	Finetune	ProtoNet	Finetune	ProtoNet
ACC	LBP-TOP	0.3824	-	0.5763	-	0.4803	-
	VGG	0.4414	0.3488	0.7078	0.6338	0.5501	0.5606
	Quang's	0.4356	0.3531	0.6981	0.6750	0.5573	0.5988
	Liu's	0.4679	0.4581	0.8156	0.8013	0.5892	**0.6287**
	MERAU	0.5279	**0.5119**	0.8368	**0.8188**	0.6373	0.6213
	MERAU (CD)	**0.5528**	-	**0.8557**	-	**0.6512**	-
UF1	LBP-TOP	0.3356	-	0.5537	-	0.4578	-
	VGG	0.4192	0.3116	0.6730	0.6273	0.5458	0.5420
	Quang's	0.4237	0.3370	0.6871	0.6594	0.5543	0.5651
	Liu's	0.4546	0.4427	0.7806	0.7943	0.5763	**0.6099**
	MERAU	0.5232	**0.5039**	0.8064	**0.7975**	0.6038	0.6075
	MERAU (CD)	**0.5435**	-	**0.8482**	-	**0.6462**	-
UAR	LBP-TOP	0.3679	-	0.5916	-	0.5047	-
	VGG	0.4269	0.3381	0.7075	0.6316	0.5694	0.5628
	Quang's	0.4340	0.3637	0.7090	0.6458	0.5771	0.5694
	Liu's	0.4879	0.4756	0.7835	0.7820	0.5763	0.6072
	MERAU	0.5465	**0.5118**	0.8141	**0.7932**	0.6257	**0.6284**
	MERAU (CD)	**0.5733**	-	**0.8386**	-	**0.6709**	-

TABLE 4. Few-shot evaluation with SMIC on ACC, UF1 and UAR.

Setting (SMIC)	3-Way-5-Shot					
Method	Fine-tuning			ProtoNet		
	ACC	UF1	UAR	ACC	UF1	UAR
LBP-TOP	0.3921	0.3528	0.3842	-	-	-
VGG	0.4218	0.4035	0.4169	0.3508	0.3472	0.3691
Quang's	0.4124	0.3997	0.4190	0.3583	0.3501	0.3673
Liu's	0.3764	0.3562	0.3808	0.3900	0.3799	0.3923
MERAU	**0.4390**	**0.4216**	**0.4397**	**0.4117**	**0.4063**	**0.4112**
MERAU (CD)	0.4139	0.4087	0.4250	-	-	-

$\{\boldsymbol{x}_1^s, \cdots, \boldsymbol{x}_{N_s}^s\}$ and $\{\boldsymbol{x}_1^t, \cdots, \boldsymbol{x}_{N_t}^t\}$ are features of samples from source domain S and target domain T. MMD can effectively represent distances between distributions. As the results show, the MMD between SMIC and CASME II is 1.1588, while it between CASME and CASME II is only 0.0756. It verifies our claim that there is a larger domain-shift between CASME II and SMIC, than CASME II and CASME. As some cross-domain learning studies [73]–[75] constrained MMD or other domain-shift indicators between two domains to minimize domain-shift, achieving remarkable results of domain adaptation, it could further improve our system performance to incorporate these methods in the training process.

Comparing the performance of fine-tuning and ProtoNet, fine-tuning is superior to ProtoNet, because metric-based few-shot learning methods cannot fine-tune parameters of the model on the target domain. In other words, models trained with metric-based few-shot learning methods cannot acquire knowledge from the target domain well.

In fine-tuning, we replace basic classifier with cosine-distance based classifier, which improves the recognition accuracy of MERAU on CASME dataset. However, its performance on SMIC dataset is poorer. The reverse effect of cosine-distance based classifier on SMIC is attributed to the scale of domain-shift, since the cosine-distance based classifier is designed for reducing intra-class difference with the sacrifice of cross-domain adaptability.

According to the confusion matrix shown in Figure 8 and Figure 9, MERAU has better cognitive ability of recognizing familiar classes (*Disgust* and *Surprise*), but may confuse

samples of the *Tense* class with those of the *Repression* class.

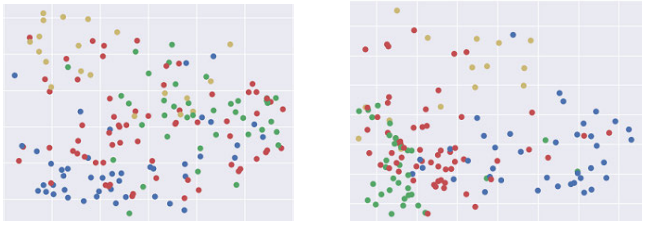


FIGURE 5. (a) Shows feature embeddings generated by Liu's model, and (b) shows feature embeddings generated by MERAU.

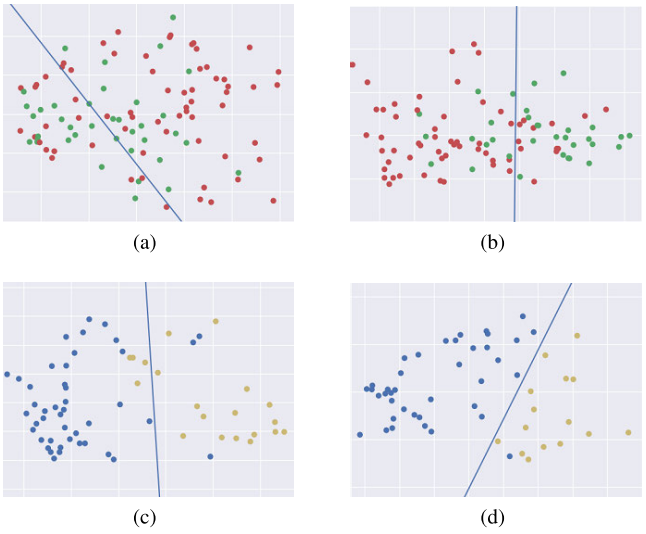


FIGURE 6. (a), (b) Show samples of hard group (*tense* and *repression*) from CASME. (c), (d) show samples of easy group (*disgust* and *surprise*) from CASME. (a) and (c) show features computed by Liu's method, (b) and (d) show features computed by our MERAU.

D. VISUALIZATION

In order to show the effectiveness of our proposed model, we give a case study, demonstrating the feature spaces learnt by Liu's model and MERAU. After training two models with D_{train} and fine-tuning with $D_{support}$, we feed all samples in CASME to both models. Feature embeddings generated by two models before classification are recorded. To visualize these feature embeddings, we conduct Principal Component Analysis (PCA) for dimension reduction, so that these samples can be presented in the same 2-dimensional space in Figure 5.

To quantify the effectiveness of the feature embeddings generated by MERAU and the baseline Liu's model, we split visualized samples into Easy group and Hard group. For each group of samples, we use SVM to acquire its best linear boundary, and then draw it as the blue line in Figure 6. Furthermore, we compute the classification accuracy of SVM for each image, and show it in Table 5. As the results shown, feature embeddings generated by MERAU have higher intra-cluster similarity and lower inter-cluster similarity, demonstrating that MERAU have better distinction among different categories of samples.

TABLE 5. Classification accuracy of Liu's method and our MERAU.

	Hard	Easy
Liu's	77.20%	95.81%
MERAU	**86.20%**	**99.40%**

E. ABLATION STUDY

In order to verify the effectiveness of incorporating action units, we eliminate the AU module in our framework, and only feed optical flow features into the classifier. Consistent with former experiments, we pre-train this model on CASME II and fine-tune it on CASME and SMIC. Figure 7 shows the classification result. It turns out that the UF1 scores of classification improve by 0.0722 and 0.0471, respectively, after we incorporate the AU module. When we replace the basic classifier with cosine-distance based classifier, incorporating knowledge of AUs has similar improvements.

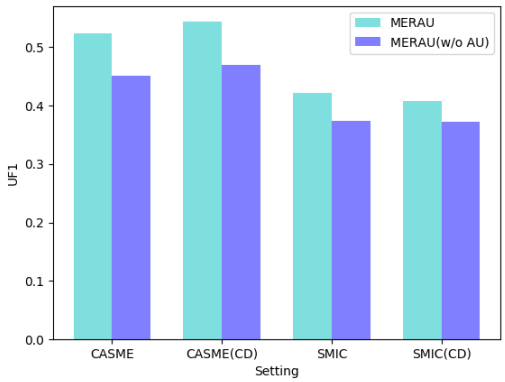


FIGURE 7. Effectiveness of the AU module in MERAU.

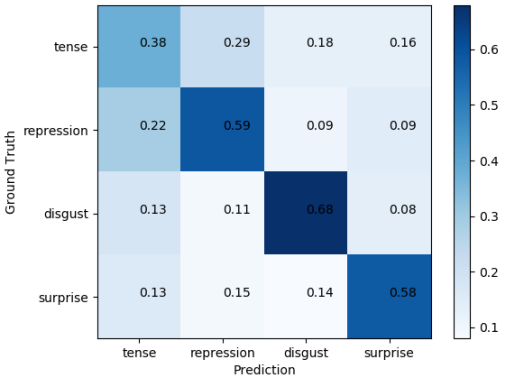


FIGURE 8. Confusion matrix of MERAU on CASME dataset.

FIGURE 9. Confusion matrix of MERAU on SMIC dataset.

V. CONCLUSION AND FUTURE WORK

Micro-expression recognition has a wide range of applications (e.g., psychological and clinical diagnosis, emotional analysis, criminal investigation, etc.). However, when a micro-expression recognition system works in cold-start conditions, it has to recognize novel classes of micro-expressions in a new scenario, suffering from the lack of sufficient labeled samples. Meanwhile, inconsistency in micro-expression labeling criteria makes it challenging to use existing labeled datasets in other scenarios.

To tackle these challenges, we present a micro-expression recognizer, which on one hand leverages the knowledge of facial action units (AU) to enhance facial representation, and on the other hand performs cross-domain few-shot learning to transfer knowledge acquired from labeled samples in datasets available from other scenarios to classify samples in the cold-starting scenario. The experimental results show that our recognizer has better distinction among samples of different micro-expression categories and achieves better recognition accuracies than state-of-art methods. On UF1 metric, our recognizer outperforms baseline methods by 0.089 on CASME dataset, and 0.022 on SMIC dataset.

For future work, we assert that the micro-expression recognition accuracy of our recognizer largely relies on the performance of facial action units detection. One future work is to incorporate cross-domain learning methods into the pre-training process of the AU detector in our framework, as it has to work in a different domain and predict possibilities of different AU's occurrences. In addition, after the cold-start period, more labeled samples will be available, and the micro-expression recognition model has to adapt to the new samples, also known as hot update. To avoid repeatedly learning from the old samples or forgetting knowledge learnt, continual learning technologies need to be further investigated.

ACKNOWLEDGMENT

The authors would like to thank all the anonymous reviewers' constructive comments, enabling them to improve the manuscript. The work was supported by the National Natural Science Foundation of China (61872214, 61521002).

REFERENCES

[1] Y.-I. Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 97–115, Feb. 2001.

[2] P. Ekman and W. Friesen, "Nonverbal leakage and clues to deception," *Psychiatry*, vol. 32, no. 1, pp. 88–106, 1969.

[3] E. A. Haggard and K. S. Isaacs, "Micromomentary facial expressions as indicators of ego mechanisms in psychotherapy," in *Methods of Research in Psychotherapy*. Boston, MA, USA: Springer, 1966, pp. 154–165.

[4] Y. Wang, J. See, R. C.-W. Phan, and Y.-H. Oh, "LBP with six intersection points: Reducing redundant information in LBP-TOP for micro-expression recognition," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 525–537.

[5] S. Liong, J. See, K. Wong, and R. C. Phan, "Less is more: Micro-expression recognition from video using apex frame," *Signal Process., Image Commun.*, vol. 62, pp. 82–92, Mar. 2018.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 2015, pp. 1–14.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.

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

[9] S. P. T. Reddy, S. T. Karri, S. R. Dubey, and S. Mukherjee, "Spontaneous facial micro-expression recognition using 3D spatiotemporal convolutional neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.

[10] Z. Xia, X. Hong, X. Gao, X. Feng, and G. Zhao, "Spatiotemporal recurrent convolutional networks for recognizing spontaneous micro-expressions," *IEEE Trans. Multimedia*, vol. 22, no. 3, pp. 626–640, Mar. 2020.

[11] Y. Liu, H. Du, L. Zheng, and T. Gedeon, "A neural micro-expression recognizer," in *Proc. 14th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2019, pp. 1–4.

[12] N. V. Quang, J. Chun, and T. Tokuyama, "CapsuleNet for micro-expression recognition," in *Proc. 14th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2019, pp. 1–7.

[13] A. J. R. Kumar and B. Bhanu, "Micro-expression classification based on landmark relations with graph attention convolutional network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 1511–1520.

[14] M. A. Takalkar, S. Thuseethan, S. Rajasegarar, Z. Chaczko, M. Xu, and J. Yearwood, "LGAttNet: Automatic micro-expression detection using dual-stream local and global attentions," *Knowl.-Based Syst.*, vol. 212, Jan. 2021, Art. no. 106566.

[15] T.-K. Tran, Q.-N. Vo, X. Hong, X. Li, and G. Zhao, "Micro-expression spotting: A new benchmark," *Neurocomputing*, vol. 443, pp. 356–368, Jul. 2021.

[16] S.-J. Wang, Y. He, J. Li, and X. Fu, "MESNet: A convolutional neural network for spotting multi-scale micro-expression intervals in long videos," *IEEE Trans. Image Process.*, vol. 30, pp. 3956–3969, 2021.

[17] M. Yan, J. Sang, T. Mei, and C. Xu, "Friend transfer: Cold-start friend recommendation with cross-platform transfer learning of social knowledge," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2013, pp. 1–6.

[18] J. See, M. H. Yap, J. Li, X. Hong, and S.-J. Wang, "MEGC 2019—The second facial micro-expressions grand challenge," in *Proc. 14th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2019, pp. 1–5.

[19] X. Ben, Y. Ren, J. Zhang, S.-J. Wang, K. Kpalma, W. Meng, and Y.-J. Liu, "Video-based facial micro-expression analysis: A survey of datasets, features and algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Mar. 19, 2021, doi: 10.1109/TPAMI.2021.3067464.

[20] W.-J. Yan, X. Li, S.-J. Wang, G. Zhao, Y.-J. Liu, Y.-H. Chen, and X. Fu, "CASME II: An improved spontaneous micro-expression database and the baseline evaluation," *PLoS ONE*, vol. 9, no. 1, Jan. 2014, Art. no. e86041.

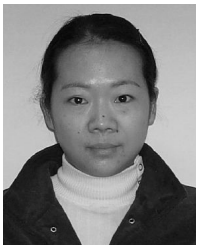
[21] W.-J. Yan, Q. Wu, Y.-J. Liu, S.-J. Wang, and X. Fu, "CASME database: A dataset of spontaneous micro-expressions collected from neutralized faces," in *Proc. 10th IEEE Int. Conf. Workshops Automat. Face Gesture Recognit. (FG)*, Apr. 2013, pp. 1–7.

[22] M. H. Yap, J. See, X. Hong, and S.-J. Wang, "Facial micro-expressions grand challenge 2018 summary," in *Proc. 13th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2018, pp. 675–678.

[23] P. Ekman and W. V. Friesen, *Manual for the Facial Action Coding System*. Palo Alto, CA, USA: Consulting Psychologists Press, 1978.
[24] F. Qu, S.-J. Wang, W.-J. Yan, H. Li, S. Wu, and X. Fu, "CAS(ME)^2: A database for spontaneous macro-expression and micro-expression spotting and recognition," *IEEE Trans. Affect. Comput.*, vol. 9, no. 4, pp. 424–436, Jan. 2017.
[25] A. K. Davison, C. Lansley, N. Costen, K. Tan, and M. H. Yap, "SAMM: A spontaneous micro-facial movement dataset," *IEEE Trans. Affective Comput.*, vol. 9, no. 1, pp. 116–129, Jun. 2018.
[26] R. Aljundi, K. Kelchtermans, and T. Tuytelaars, "Task-free continual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11254–11263.
[27] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3987–3995.
[28] M. B. Ring, "Continual learning in reinforcement environments," Ph.D. dissertation, Dept. Comput. Sci., Univ. Texas Austin, Austin, TX, USA, 1994.
[29] R. Hadsell, D. Rao, A. A. Rusu, and R. Pascanu, "Embracing change: Continual learning in deep neural networks," *Trends Cognit. Sci.*, vol. 24, no. 12, pp. 1028–1040, Dec. 2020.
[30] B. Ghai, Q. V. Liao, Y. Zhang, R. Bellamy, and K. Mueller, "Explainable active learning (XAL): Toward AI explanations as interfaces for machine teachers," *Proc. ACM Hum.-Comput. Interact.*, vol. 4, pp. 1–28, Jan. 2021, doi: 10.1145/3432934.
[31] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res.*, vol. 4, pp. 129–145, Mar. 1996.
[32] O. Borkowski, M. Koch, A. Zettor, A. Pandi, A. C. Batista, P. Soudier, and J.-L. Faulon, "Large scale active-learning-guided exploration for *in vitro* protein production optimization," *Nature Commun.*, vol. 11, no. 1, pp. 1–8, Dec. 2020.
[33] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 915–928, Jun. 2007.
[34] W. Merghani, A. Davison, and M. Yap, "Facial micro-expressions grand challenge 2018: Evaluating spatio-temporal features for classification of objective classes," in *Proc. 13th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2018, pp. 662–666.
[35] H.-X. Xie, L. Lo, H.-H. Shuai, and W.-H. Cheng, "AU-assisted graph attention convolutional network for micro-expression recognition," in *Proc. 28th ACM Int. Conf. Multimedia*, Seattle, WA, USA, Oct. 2020, pp. 2871–2880.
[36] Y. Li, X. Huang, and G. Zhao, "Joint local and global information learning with single apex frame detection for micro-expression recognition," *IEEE Trans. Image Process.*, vol. 30, pp. 249–263, 2021.
[37] Z. Lai, R. Chen, J. Jia, and Y. Qian, "Real-time micro-expression recognition based on ResNet and atrous convolutions," *J. Ambient Intell. Hum. Comput.*, vol. 546, pp. 1–12, Mar. 2020.
[38] M. F. Hashmi, B. K. K. Ashish, V. Sharma, A. G. Keskar, N. D. Bokde, J. H. Yoon, and Z. W. Geem, "LARNet: Real-time detection of facial micro expression using lossless attention residual network," *Sensors*, vol. 21, no. 4, p. 1098, Feb. 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/4/1098
[39] Y. Li, J. Zeng, S. Shan, and X. Chen, "Self-supervised representation learning from videos for facial action unit detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 10924–10933.
[40] J. Zeng, W. Chu, F. D. la Torre, J. F. Cohn, and Z. Xiong, "Confidence preserving machine for facial action unit detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, CL, USA, Dec. 2015, pp. 3622–3630.
[41] K. Zhao, W.-S. Chu, and H. Zhang, "Deep region and multi-label learning for facial action unit detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 3391–3399.
[42] W. Li, F. Abtahi, Z. Zhu, and L. Yin, "EAC-Net: A region-based deep enhancing and cropping approach for facial action unit detection," in *Proc. 12th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2017, pp. 103–110.
[43] J. Yang, T. Qian, F. Zhang, and S. U. Khan, "Real-time facial expression recognition based on edge computing," *IEEE Access*, vol. 9, pp. 76178–76190, 2021.
[44] A. Savran, B. Sankur, and M. T. Bilge, "Regression-based intensity estimation of facial action units," *Image Vis. Comput.*, vol. 30, no. 10, pp. 774–784, Oct. 2012.
[45] Y. Zhang, R. Zhao, W. Dong, B.-G. Hu, and Q. Ji, "Bilateral ordinal relevance multi-instance regression for facial action unit intensity estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7034–7043.
[46] R. Walecki, O. Rudovic, V. Pavlovic, B. Schuller, and M. Pantic, "Deep structured learning for facial action unit intensity estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5709–5718.
[47] Y. Zhang, W. Dong, B.-G. Hu, and Q. Ji, "Weakly-supervised deep convolutional neural network learning for facial action unit intensity estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2314–2323.
[48] Y. Zhang, Y. Fan, W. Dong, B.-G. Hu, and Q. Ji, "Semi-supervised deep neural network for joint intensity estimation of multiple facial action units," *IEEE Access*, vol. 7, pp. 150743–150756, 2019.
[49] B. Jiang, M. F. Valstar, and M. Pantic, "Action unit detection using sparse appearance descriptors in space-time video volumes," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Mar. 2011, pp. 314–321.
[50] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, no. 3, pp. 1–34, Jul. 2020.
[51] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds. Long Beach, CA, USA, Dec. 2017, pp. 4077–4087.
[52] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, Lille, France, vol. 2, 2015, pp. 1–8.
[53] J. Guan, M. Zhang, and Z. Lu, "Large-scale cross-domain few-shot learning," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, Nov. 2020, pp. 1–17.
[54] A.-N. Ciubotaru, A. Devos, B. Bozorgtabar, J.-P. Thiran, and M. Gabrani, "Revisiting few-shot learning for facial expression recognition," 2019, *arXiv:1912.02751*. [Online]. Available: https://arxiv.org/abs/1912.02751
[55] H. Tseng, H. Lee, J. Huang, and M. Yang, "Cross-domain few-shot classification via learned feature-wise transformation," in *Proc. 8th Int. Conf. Learn. Represent. (ICLR)*, Addis Ababa, Ethiopia, Apr. 2020, pp. 1–18.
[56] J. Cai and S. M. Shen, "Cross-domain few-shot learning with meta fine-tuning," 2020, *arXiv:2005.10544*. [Online]. Available: https://arxiv.org/abs/2005.10544
[57] W. Chen, Y. Liu, Z. Kira, Y. F. Wang, and J. Huang, "A closer look at few-shot classification," in *Proc. 7th Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, May 2019, pp. 1–17.
[58] Mustaqeem and S. Kwon, "MLT-DNet: Speech emotion recognition using 1D dilated CNN based on multi-learning trick approach," *Expert Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114177.
[59] D. Issa, M. F. Demirci, and A. Yazici, "Speech emotion recognition with deep convolutional neural networks," *Biomed. Signal Process. Control*, vol. 59, May 2020, Art. no. 101894. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809420300501
[60] N. Vryzas, L. Vrysis, M. Matsiola, R. Kotsakis, C. Dimoulas, and G. Kalliris, "Continuous speech emotion recognition with convolutional neural networks," *J. Audio Eng. Soc.*, vol. 68, no. 1/2, pp. 14–24, Feb. 2020.
[61] T. Tuncer, S. Dogan, and A. Subasi, "A new fractal pattern feature generation function based emotion recognition method using EEG," *Chaos, Solitons Fractals*, vol. 144, Mar. 2021, Art. no. 110671.
[62] H. Cui, A. Liu, X. Zhang, X. Chen, K. Wang, and X. Chen, "EEG-based emotion recognition using an end-to-end regional-asymmetric convolutional neural network," *Knowl.-Based Syst.*, vol. 205, Oct. 2020, Art. no. 106243.
[63] I. Gallo, A. Calefati, and S. Nawaz, "Multimodal classification fusion in real-world scenarios," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, vol. 5, Nov. 2017, pp. 36–41.
[64] H. Zhang, L. Cao, L. Feng, and M. Yang, "Multi-modal interactive fusion method for detecting teenagers' psychological stress," *J. Biomed. Informat.*, vol. 106, Jun. 2020, Art. no. 103427.
[65] J. Munro and D. Damen, "Multi-modal domain adaptation for fine-grained action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 119–129.
[66] S. Li, L. Zhang, and X. Diao, "Deep-learning-based human intention prediction using RGB images and optical flow," *J. Intell. Robotic Syst.*, vol. 97, no. 1, pp. 95–107, Jan. 2020.
[67] W. Gao, G. Liao, S. Ma, G. Li, Y. Liang, and W. Lin, "Unified information fusion network for multi-modal RGB-D and RGB-T salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, early access, May 25, 2021, doi: 10.1109/TCSVT.2021.3082939.

[68] K. Gadzicki, R. Khamsehashari, and C. Zetzsche, "Early vs late fusion in multimodal convolutional neural networks," in *Proc. IEEE 23rd Int. Conf. Inf. Fusion (FUSION)*, Jul. 2020, pp. 1–6.
[69] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "Multimodal vehicle detection: Fusing 3D-LIDAR and color camera data," *Pattern Recognit. Lett.*, vol. 115, pp. 20–29, Nov. 2018.
[70] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Image Analysis*, J. Bigun and T. Gustavsson, Eds. Berlin, Germany: Springer, 2003, pp. 363–370.
[71] X. Li, T. Pfister, X. Huang, G. Zhao, and M. Pietikainen, "A spontaneous micro-expression database: Inducement, collection and baseline," in *Proc. 10th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, Apr. 2013, pp. 1–6.
[72] A. Gretton, K. Borgwardt, M. J. Rasch, B. Scholkopf, and A. J. Smola, "A kernel method for the two-sample problem," 2008, *arXiv:0805.2368*. [Online]. Available: https://arxiv.org/abs/0805.2368
[73] Z. Luo, J. Hu, W. Deng, and H. Shen, "Deep unsupervised domain adaptation for face recognition," in *Proc. 13th IEEE Int. Conf. Automat. Face Gesture Recognit. (FG)*, May 2018, pp. 453–457.
[74] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. 32nd Int. Conf. Mach. Learn.*, F. Bach and D. Blei, Eds. Lille, France, Jul. 2015, pp. 97–105. [Online]. Available: https://proceedings.mlr.press/v37/long15.html
[75] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," 2014, *arXiv:1412.3474*. [Online]. Available: https://arxiv.org/abs/1412.3474

YI DAI (Member, IEEE) is currently pursuing the bachelor's degree with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include computational psychology, sentiment analysis, and natural language processing.



LING FENG (Senior Member, IEEE) is currently a Professor of computer science and technology with Tsinghua University, China. Her research interests include computational mental healthcare, context-aware data management and services toward ambient intelligence, data mining and warehousing, and distributed object-oriented database systems.