Neurocomputing 448 (2021) 276–289

Contents lists available at ScienceDirect

Neurocomputing

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

A two-stage 3D CNN based learning method for spontaneous micro-expression recognition

Sirui Zhao [a,b], Hanqing Tao [a], Yangsong Zhang [b], Tong Xu [a], Kun Zhang [c], Zhongkai Hao [a], Enhong Chen [a,*]

[a] School of Computer Science and Technology, University of Science and Technology of China (USTC), Hefei, China
[b] School of Computer Science and Technology, Southwest University of Science and Technology (SWUST), Mianyang, China
[c] School of Computer Science and Information Engineering, Hefei University of Technology (HFUT), Hefei, China

ARTICLE INFO

Article history:
Received 20 June 2020
Revised 19 January 2021
Accepted 22 March 2021
Available online 29 March 2021
Communicated by Zidong Wang

Keywords:
Facial micro-expression
Emotion recognition
Siamese network
3D convolutional neural network
Key-frames

ABSTRACT

Micro-expressions (MEs) are spontaneous and involuntary facial subtle reactions which often reveal the genuine emotions within human beings. Recognizing MEs automatically is becoming increasingly crucial for many areas such as diagnosis and security. However, the short time duration and low spatial intensity of MEs pose great challenges for accurately recognizing them. Additionally, the lack of sufficient and balanced spontaneous MEs data also makes this problem even harder to solve, and some adaptive modeling strategies have been quite urgent recently. To this end, this paper draws inspirations from few-shot learning to propose a novel two-stage learning (i.e., *prior learning* and *target learning*) method based on a siamese 3D convolutional neural network for MEs recognition (MERSiamC3D). Specifically, in the prior learning stage, the proposed MERSiamC3D is used to extract the generic features of MEs. In the target learning stage, the structure and parameters of the MERSiamC3D will be carefully adjusted and the Focal Loss is adopted for high-level features learning. Afterwards, in order to effectively retain the spatiotemporal information of the original MEs video, an adaptive construction method based on adaptive convolutional neural network is proposed to construct the key-frames sequence to summarize the original MEs video, which is able to help drop the redundant frames and relatively highlight the movement of the apex frame. Then, the new key-frames are taken as the input of the two-stage learning method. Finally, through extensive evaluations and experiments on three publically available MEs datasets, the proposed method in this work could outperform traditional methods and other deep learning baselines, which provides a novel insight on how to leverage scarce data for MEs recognition.

1. Introduction

Micro-expressions (MEs) are spontaneous and subtle facial movements, which usually occur in a high-risk environment when people attempt to conceal their true feelings [9,32]. As one of the facial expressions, MEs have some special and unique characteristics. For example, MEs usually occur in a very short time ($0.04s \sim 0.5s$) [9,51] with a quite low intensity in local facial units. Meanwhile, because MEs are repressed and cannot be posed to deceive others, they especially reflect the genuine psychological state of an individual [9]. These characteristics make MEs totally different from general facial expressions and its analysis has many potential applications such as lie detection [8,48], treatment of depression [16], business negotiation [14], homeland security [8,39,48] and so on.

Previous investigations on MEs were mainly carried out in the field of psychology and relied on tedious manual analysis [2,7,10]. Recently, with the rapid development of the video acquisition technologies and intelligent learning algorithms [32], many researchers turn to focus more on automatic MEs analysis in the field of computer vision and affective computing, especially for MEs recognition (MER) tasks. For example, Polikovsky et al. [38] proposed to use 3D histograms of oriented gradients (3DHOG) descriptor to recognize the motion of MEs, and Wu et al. [49] adopted the gabor filters to extract MEs features rather than using the support vector machine (SVM) to recognize them. Afterwards, Pfister et al. [37] proposed to harness the local binary pattern-three orthogonal planes (LBP-TOP) [54] descriptor and SVM classifier to recognize MEs. Recently, to extract the discriminative spatiotemporal MEs features, Kim et al. [20] designed a CNN-LSTM model and the expression-states are taken into account in the objective functions for MER. In addition, Song et al. [42] also proposed a three-stream convolutional neural network (TSCNN) to recognize

* Corresponding author.
E-mail address: cheneh@ustc.edu.cn (E. Chen).

https://doi.org/10.1016/j.neucom.2021.03.058
0925-2312/

MEs by learning discriminative features in three key-frames of ME videos. Furthermore, many other deep learning-based methods [18,19,27,29,33–35,43,47,53,55] have been proposed for better extracting features for MER.

However, there is still a huge gap between the actual demands and high-precision MER methods, with three main technical and domain challenges inherent in MEs feature extraction and classification. Firstly, MEs usually have low intensity and only occur in local facial units, which is indistinct and hard to perceive. Secondly, MEs are usually captured and recorded by the high-speed camera due to their short duration (0.04–0.5 s) [9,51]. While high-speed camera usually captures a lot of redundant frames and produces much noisy data during recording [32], which is also a detrimental factor for the extraction of scarce MEs features. Thirdly, since MEs are self-repressed facial expressions of emotions and hard to induce, it is a gordian knot to collect large scale spontaneous MEs datasets, and the existing datasets usually contain insufficient samples. For instance, merely 255 micro-expression sequences for all emotion categories in the largest CASME II dataset [50]. Moreover, the distribution of samples per class is quite imbalanced, e.g., 99 sequences for category "Happiness" and 15 sequences for category "Sadness" in CASME II dataset with objective class labels [3,50]. The above problems have brought great difficulties for training a deep CNNs model to extract discriminative spatiotemporal features for MEs, let alone recognize them accurately.

To address the above challenges, our work here proposes MERSiamC3D, a novel two-stage learning (i.e., *prior learning* and *target learning*) method based on a siamese 3D convolutional neural network for MEs recognition. To be specific, we first propose an adaptive construction method based on adaptive CNN to construct the key-frames sequence to summarize the original MEs video. This method can help to preserve the spatiotemporal information of the original MEs video without using redundant frames andrelatively highlight the movement of the apex frame. Moreover, to solve the problem of insufficient samples in MER and inspired by the few-shot learning methods [22,24], we subdivide the features learning of MEs into two stages, i.e., *prior learning* and *target learning*. As illustrated in Fig. 1, at the prior learning stage, we divide the original MEs dataset (*with key-frames*) into a collection of same and different sample-pairs according to the original class labels for training our MERSiamC3D to get basic experiences, which are capable of extracting generic features of MEs. Then, we carefully fine-tune the structure and parameters of MERSiamC3D, and continue to train it on the original datasets (*with key-frames*) to get advanced features of MEs for target classification at the target learning stage. Furthermore, we also introduce the Focal Loss [26] into MERSiamC3D to alleviate the inefficient model training caused by the class imbalance in the MEs Datasets. To the best of our knowledge, this paper is the first work to incorporate techniques in few-shot learning to deal with data deficiency in MER task, and extensive and comprehensive experiments have been conducted on the public SMIC–HS [25], CASME II [50] and SAMM [4] datasets to demonstrate the superiority and rationality of our methods.

The rest of this paper is organized as follows: Section 2 briefly reviews the related work. Section 3 introduces the technical details of our proposed methods. Section 4 presents our extensive experiments on three publically available MEs datasets SMIC–HS, CASMEII and SAMM to demonstrate the superiority and rationality of our MERSiamC3D. Afterwards, Section 5 presents the ablation studies to verify the rationality and effectiveness of each part of our method. Finally, we summarize the whole paper in Section 6.

2. Related work

Over the past few years, MER has garnered increasing attention with a special focus on exploring the efficient and discriminative spatiotemporal features for MEs. These works could be roughly divided into two distinct categories: hand-crafted approaches

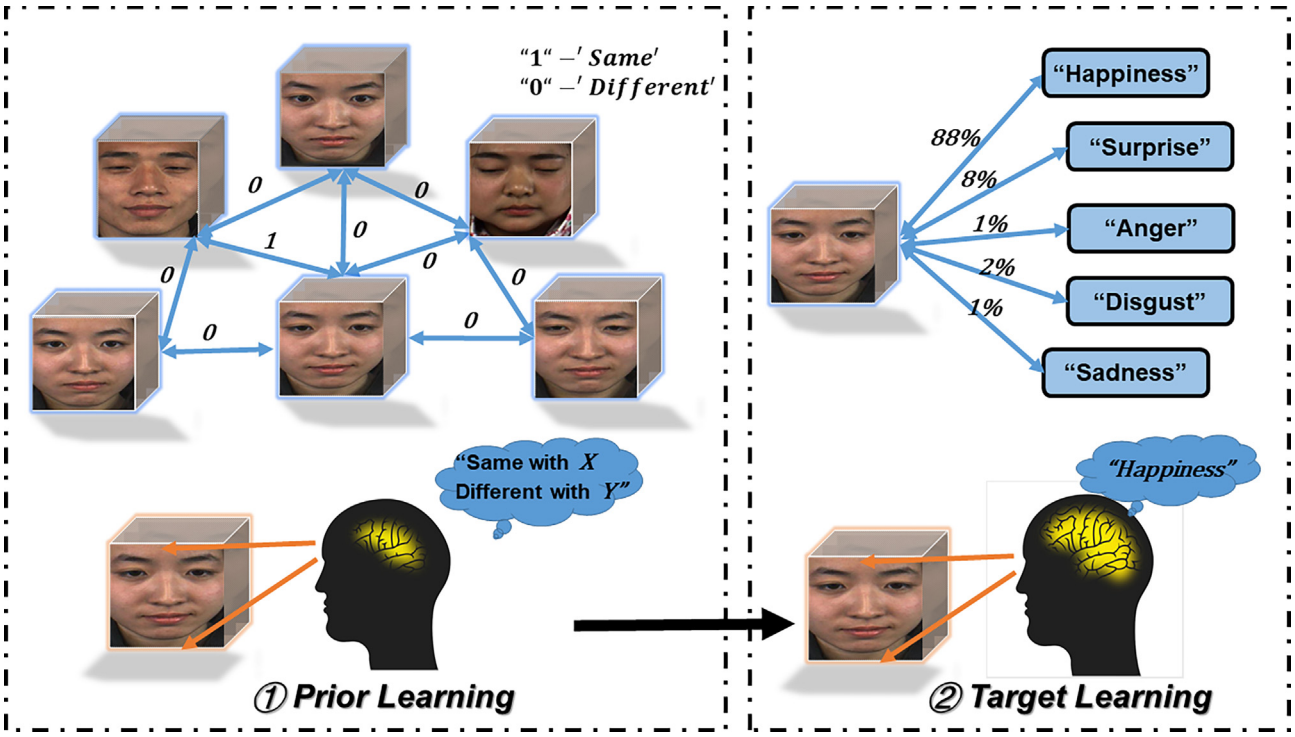


Fig. 1. An illustration of the proposed two-stage features learning strategy for MER. The left part depicts the *prior learning* stage. Here, we divide the original MEs dataset (*with key-frames*) into a collection of same and different samples pairs for training model to get basic experiences, which are capable of extracting generic features of MEs. While the right part shows the idea of target learning, where we continue to train the model on the original datasets (*with key-frames*) to learn high-level MEs features for target classification.

and deep learning approaches. In this section, we will give an overview on features extraction for MEs, together with related studies on few-shot learning.

2.1. Hand-crafted approaches

These approaches for MER could date back to one decade ago. At first, Polikovsky et al. [38] proposed a 3D histograms of oriented gradients (3DHOG) descriptor and used *k*-means algorithm to classify onset, apex and offset frames. Later, recognition research gradually focused on spontaneous micro-expression. Then, Pfister et al. [37] harnessed the temporal interpolation model (TIM) to increase the frames and use local binary pattern-three orthogonal planes (LBP-TOP) [54] as a spatiotemporal local texture descriptor to extract dynamic features. Afterwards, the support vector machine (SVM), multiple kernel learning (MKL) and random forest (RF) have been applied in the classification. Moreover, the basic histogram of oriented optical flow (HOOF) descriptor [1] was exploited by Liu et al. [30] as a comparative feature when spotting MEs and then performing recognition. Recently, many MER approaches have relied on the above three descriptors. For example, Huang et al. [15] proposed spatiotemporal local quantized pattern (STCLQP), which exploits magnitude and orientation as complementary of sign information for improving the performance of MER. Later, Guo et al. [13] proposed the extended local binary patterns on three orthogonal planes (ELBP-TOP) for MER, which is actually a further extension of LBP-TOP. Besides, Davison et al. [3] proposed to utilize action units (AUs) to classify MEs for spontaneous MEs datasets CASME II [50] and SAMM [4], then the proposed classes were tested by using 3DHOG, LBP-TOP and HOOF.

2.2. Deep learning approaches

The first work to explore the possibility of using deep learning for MER is proposed by Patel et al. [33], where the deep CNN models are used for transfer learning from objects for features extraction on small MEs datasets. However, only 47% accuracy could obtained on the CASME II dataset. At the same time, Kim et al. [20] designed a CNN-LSTM model and the expression-states are taken into account in the objective functions to extract the spatiotemporal features for MEs, whose overall accuracy achieved 60.98% on CASME II. Later, Peng et al. [35] proposed a dual-template 3DCNN model to adapt to the different frame rates of MEs video clips, which is a two-stream shallow network with 3D convolution units fed with the optical-flow sequences. To avoid the overfitting problem on small MEs datasets, Wang et al. [47] proposed a Transfer Long-term Convolutional Neural Network (TLCNN) and alleviated this problem to some extent. As an improvement, Khor et al. [19] further proposed an Enriched Long-term Recurrent Convolutional Network (ELRCN). Besides, Sun et al. [43] proposed a novel knowledge transfer technique distills and transfers knowledge from action unit for MER. Additionally, Song et al. [42] proposed a TSCNN model to learn ME-discriminative features by fusing the spatial, temporal and facial local region cues or the MEs video clips. Recently, Liong et al. [28] have demonstrated that it is sufficient to encode facial MEs features by only utilizing the apex frame. Inspired by this, Khor et al. [18] and Liu et al. [29] creatively utilized the single optical flow image estimated by the onset frame and apex frame to represent the entire MEs video, then the CNN-based models could be used directly. Considering the fact that using the apex frame could get rid of redundant video frames but the relevant temporal evidence of MEs would be thereby left out, Peng et al. [34] proposed an Apex-Time Network (ATNet) to recognize MEs. On the whole, the MER has been boosted ever since the introduction of deep learning approaches. However, deep learning-based MER research is still in its infancy, and the lack of large-scale MEs datasets still remains the biggest challenge.

2.3. Few-shot learning

Since deep learning methods are often data-intensive, they cannot provide satisfactory performance for learning features from limited samples. Fortunately, few-shot learning strategies could well deal with such cases. Recent years have witnessed the power of few-shot learning in extracting informative features from small datasets. As a result, many few-shot learning methods have been proposed one after another. Koch et al. [22] first employed a supervised metric-based approach with siamese convolutional neural networks to learn good features, then reused the network's features for one-shot learning without any retraining. For the same task, Vinyals et al. [45] and Snell et al. [41] respectively proposed matching networks and prototypical networks to alleviate this issue. However, to the best of our knowledge, there are no studies currently in the literature adopting few-shot learning ideas for MER. To this end, we propose our MERSiamC3D for spatiotemporal feature extraction and MEs recognition, which could enjoy the advantage of few-shot learning hence provide robust performance when faced with small datasets.

3. Methodology

In this section, we will introduce our methods mainly from two aspects. Firstly, we propose an adaptive construction method to construct the key-frame sequences and establish the corresponding optical flow (OF) sequences for data preparation, which is shown in Fig. 2. Then, we delve into the technical details of our MERSiamC3D as shown in Fig. 3 by elaborating the two-stage MEs feature learning strategy (i.e., *prior learning* and *target learning*).

3.1. Data preprocessing

3.1.1. Adaptive construction of key-frames sequence.

As different MEs samples have different durations, the first step to use them as our model's input is to normalize the length of all MEs samples. Additionally, since the original MEs video is usually captured by high-speed cameras, it is unavoidable to contain a lot of redundant frames and noise. Therefore directly using the original MEs samples as the model input will be unpractical and bring lots of unnecessary noise. On the one hand, training with inadequate samples is very likely to encounter overfitting problems. On the other hand, the noise existing in input is fatal for extracting the weak and scarce micro-expression features, which will seriously hinder our feature learning process. Taking these factors into account, we propose to construct key-frames sequence to summarize the original MEs sequence.

To obtain the key-frames sequences, three important principles need to be satisfied: (i) The key-frames sequence needs to be sufficient and consistent with the temporal dynamics of MEs, which refers to the movements of MEs that involve onset (start), apex (peak), offset (end), and transitions between them [32]. (ii) No or as few noisy frames as possible in the key-frames sequence. (iii) The movement of the apex frame needs to be highlighted since it has been proven to contribute major information for facial-expression recognition [6].

Following the above principles and based on the 3 known key-frames, i.e., the *onset frame, apex frame* and *offset frame*, we propose an adaptive construction method to generate the final RGB key-frames sequence, and then the corresponding OF key-frames sequence is calculated to describe the dynamic spatiotemporal

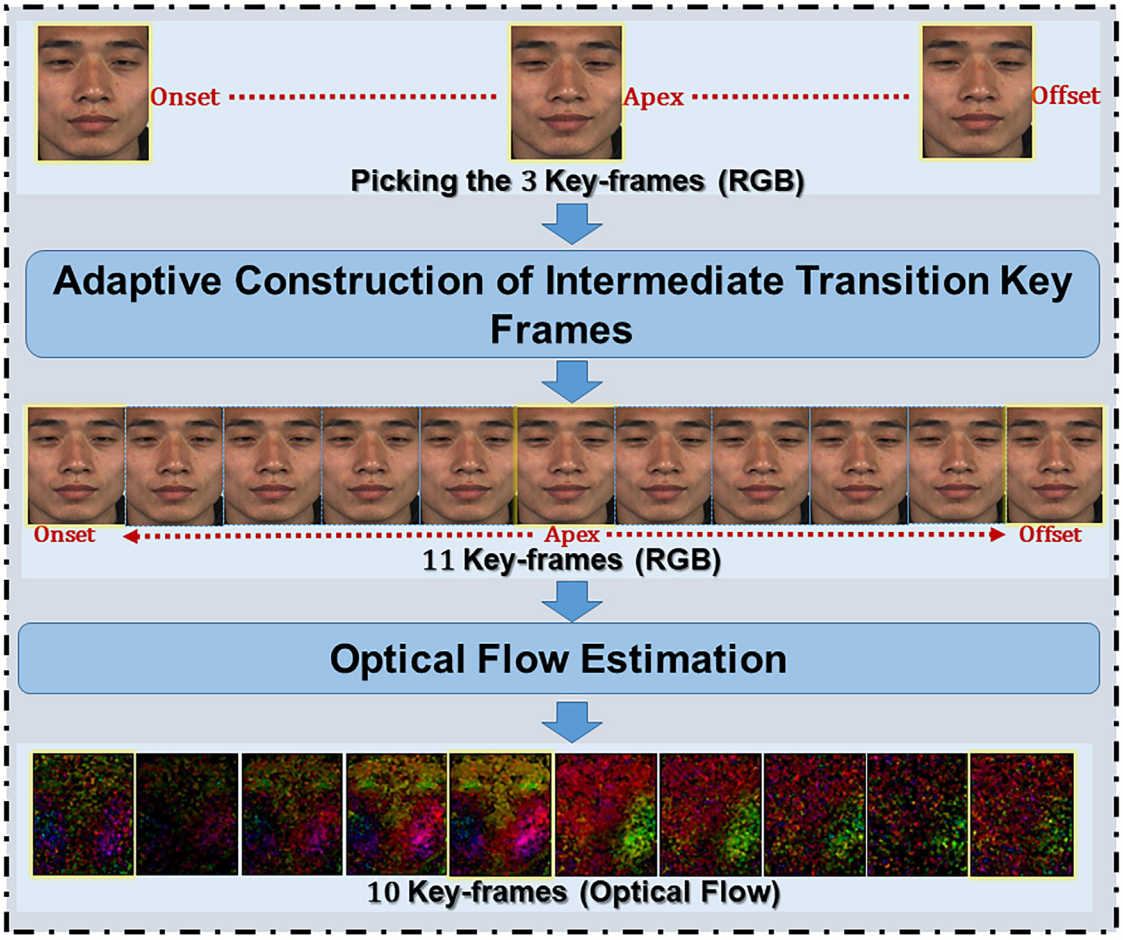


Fig. 2. A schematic illustration of data preparation. Based on the onset frame, apex frame, and offset frame from the original MEs sequence, we leverage the adaptive construction method to generate 8 intermediate transition frames. The 11 RGB key-frames are used to summarize the original MEs sequence. Finally, the optical-flow sequence with 10 frames will be obtained by calculating the adjacent frames of the 11 RGB key-frames sequence.

information of the original MEs video. Besides, the length of the final OF key-frames sequence is empirically set to 10, which has also been proven to be effective by many previous works [13,19,25,36]. That is, based on the 3 known key-frames, the adaptive construction method will generate 8 intermediate transition frames, therefore using an RGB sequence with 11 key-frames to describe a MEs sample. Besides, it should be mentioned that although the onset frame, apex frame and offset frame are annotated in the CASME II [50] and SAMM [4] datasets, the apex frames in the SMIC–HS [25] are omitted. To solve this problem, and inspired by the work of Zhou et al. [55], the mid-position frame between the onset and offset is applied as the approximated apex frame in the SMIC–HS dataset.

To be more specific, the adaptive construction method includes spatially adaptive and temporally adaptive strategies. For the former ones, when given two frames, the adaptive convolutional neural network (AdConv) [31] is used to generate the middle frame, which could well integrate motion estimation and pixel synthesis. For the latter ones, the numbers of interpolated frames for the onset-apex clip and apex-offset clip are not equal, which will be respectively determined by time intervals and the length of the original sequence. For notation convenience, we utilize N_{sp} and N_{pe} to respectively denote the number of interpolated frames for the onset-apex clip and apex-offset clip, and use t_s, t_p and t_e to respectively represent the time index of onset frame (k_s), apex frame (k_p) and offset frame (k_e) in the original MEs video. Then, for a given MEs video with N frames, we could have the equations below:

$$\begin{aligned} N_{sp} &= (t_p - t_s + 1) * 8/N, \\ N_{pe} &= (t_e - t_p) * 8/N, \\ N_{sp} &+ N_{pe} = 8. \end{aligned} \tag{1}$$

Furthermore, as the apex frame contains the strongest intensity expression, the motion of the apex frame should be highlighted when considering the spatiotemporal variations between MEs adjacent frames. Meanwhile, from the perspective of movement, the motion intensity of the apex frame could be regarded as the continuous accumulation of the motion intensities of all frames before it. Therefore, when generating the intermediate transition key-frames in the onset-apex clip and apex-offset clip, increasing the time interval between the apex frame and its adjacent frames can relatively highlight the change in the motion intensity of the apex frame when calculating the optical flow of adjacent frames. To this end, we introduce the sparse generation of intermediate transition frames on both sides of the apex frame and estimate the optical flow of adjacent two frames to describe the MEs motion. The method of adaptively constructing key-frames sequence is shown in Algorithm 1.

Algorithm 1. Adaptive construction of key-frames sequence. N is the length of a MEs video. N_{sp} and N_{pe} respectively denote the number of interpolated frames for the onset-apex clip and apex-offset clip. ∪ dontes ordered merge operation, and / indicates the division to round down here.

Input: the 3 key-frames sequence:$I_i = \{k_s, k_p, k_e\}$, and its time index collection:$T_i = \{t_s, t_p, t_e\}$.
Parameter: N, N_{sp}, N_{pe}, I_i', I_i'', T_i', T_i'', k_m', m.
Output: the final key-frames sequence I_o.
1: Let $N = t_e - t_s + 1$.
2: Let $I_i' = \{k_s, k_p\}$, $I_i'' = \{k_p, k_e\}$.
3: Let $N_{sp} = (t_p - t_s + 1) * 8/N$, $N_{pe} = 8 - N_{sp}$.
4: Let $t_1 = t_s$, $t_2 = t_p$, $k_1 = k_s$, $k_2 = k_p$, $L_1 = 0$, $t = 0$.
5: Clear T_i', T_i''
6: **while** $N_{sp} > L_1$ and $t = 0$ **do**
7: Update $m = (t_1 + t_2)/2$, and it's the time index of frame k_m'.
8: **if** $| m - t_1 | > 1$ **then**
9: Update $k_m' = AdConv(k_1, k_2)$.
10: Update $t_2 = m$, $k_2 = k_m'$, $L_1 = L_1 + 1$.
11: $I_i' = I_i' \cup k_m'$, $T_i' = T_i' \cup m$.
12: **else**
13: Let $t = 1$, $L = L_1$, $I = \{\}$.
14: **while** $t < L$ and $L_1 < N_{sp}$ **do**
15: Update $m = (T_i'[t] + T_i'[t-1])/2$.
16: **if** $| m - T_i'[t-1] | > 1$ **then**
17: Update $k_m' = AdConv(I_i'[t], I_i'[t+1])$.
18: Update $I = I \cup k_m'$, $L_1 = L_1 + 1$.
19: **end if**
20: Update $t = t + 1$
21: **end while**
22: $I_i' = I_i' \cup I$, $I_o = I_o \cup I_i'$
23: **end if**
24: **end while**
25: Let $t_1 = t_e, t_2 = t_p, k_1 = k_e$, $k_2 = k_p, T_i' = T_i'', I_i' = I_i'', L_1 = 0, t = 0$.
26: **while** $N_{pe} > L_1$ and $t = 0$ **do**
27: Repeat step7-24.
28: **end while**
29: Let M denotes the length of the collection I_o.
30: **while** $M < 11$ **do**
31: Using k_e padding I_o.
32: Let $M = M + 1$.
33: **end while**
34: **return** I_o

3.1.2. Optical flow sequence estimation

To describe the dynamic spatiotemporal information of the MEs sequence, and considering that if the RGB key-frames sequence is directly taken as our model input, it will be challenging to train the model to extract the high-level MEs features when the sample size of MEs datasets is small. To this end, based on the obtained RGB key-frames sequence, our work here choose to further estimate the corresponding OF sequence, which can approximately describe the facial motion in MEs video and has also been demonstrated by many current research works [19,27,29,35,55] that it can be used to enrich the input except for RGB channels. Hence, the optical flow between two adjacent RGB key-frames are calculated to obtain a 10-frames OF sequence as our model input.

Specifically, for an RGB key-frames sequence, $I(x, y, t)$ denotes the image intensity at position (x, y) and time t. Based on the assumption that brightness during a short period dt is invariant, we could have the equation below:

$$I(x, y, t) = I(x + \mathrm{d}x, y + \mathrm{d}y, t + \mathrm{d}t). \tag{2}$$

Furthermore, we assume that the image flow field is continuous and differentiable in both spatial and temporal domains. According to the Taylor series expansion, the formula above can be expanded as:

$$I(x + \mathrm{d}x, y + \mathrm{d}y, t + \mathrm{d}t) = I(x, y, t) + \frac{\partial I}{\partial x}\mathrm{d}x + \frac{\partial I}{\partial y}\mathrm{d}y + \frac{\partial I}{\partial t}\mathrm{d}t + \xi, \tag{3}$$

where ξ is the two-order or above estimator of time dt. When dt tends to be infinitesimal, we can get the OF constraint by combining (2) and (3) as follows:

$$\frac{\partial I}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial I}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}t} + \frac{\partial I}{\partial t} = 0. \tag{4}$$

Then, the corresponding OF vector V could be calculated via the following equation:

$$V = \left[u = \frac{\mathrm{d}x}{\mathrm{d}t}, v = \frac{\mathrm{d}y}{\mathrm{d}t}\right]^T, \tag{5}$$

where the amplitude is $m = \sqrt{u^2 + v^2}$. We follow the classic Farneback method [11] to implement the OF estimation, which has been implemented and integrated into the Opencv library and can be used easily.

3.2. Spatiotemporal MEs feature learning

In fact, it is relatively common for us that when a child sees a new object for the first time, he or she could generalize the characteristics of the new object by comparing it with other objects and identifying it as the same or different class, even without knowing what it is. This kind of prior experience may indeed form memories, which contributes to further learning and understanding. Inspired by similar ideas, many few-shot learning methods have been proposed to guide machine to successfully learn features from little data [22,45,41]. Encouraged by these methods, our work propose to divide the features learning of MEs into two stages, i.e., *prior learning* and *target learning*. In fact, the prior learning is analogous to the early exploration for new objects by a child mentioned above, which aims to train our model to acquire the ability and experience of extracting generic features by inputting MEs sample pairs and judging their similarity. Moreover, the target learning refers to the high-level features extraction and classification of the model based on the prior learning. The two-stage learning method is actually forming the subject of our MERSiamC3D model to obtain MEs features. Consequently, as shown in Fig. 3, the structure of MERSiamC3D model will have two different variants which correspond to prior learning and target learning, respectively.

3.2.1. Prior learning

Here, our intention lies in training a neural network first to get basic experience which are capable of extracting generic features of MEs and could be stored for target learning. As shown in Fig. 3 (A), a variance of siamese 3D convolutional neural network (MERSiamC3D) is designed which takes MEs-sample pairs as input.

3.2.1.1. The architecture of MERSiamC3D model. Our MERSiamC3D model is solidly designed based on 3D convolutional neural networks (C3D), which has the capability of extracting features from spatial and temporal dimensions simultaneously by performing 3D convolutional operations in multiple contiguous frames. Meanwhile, the 3D convolution is achieved by convolving a 3D kernel to

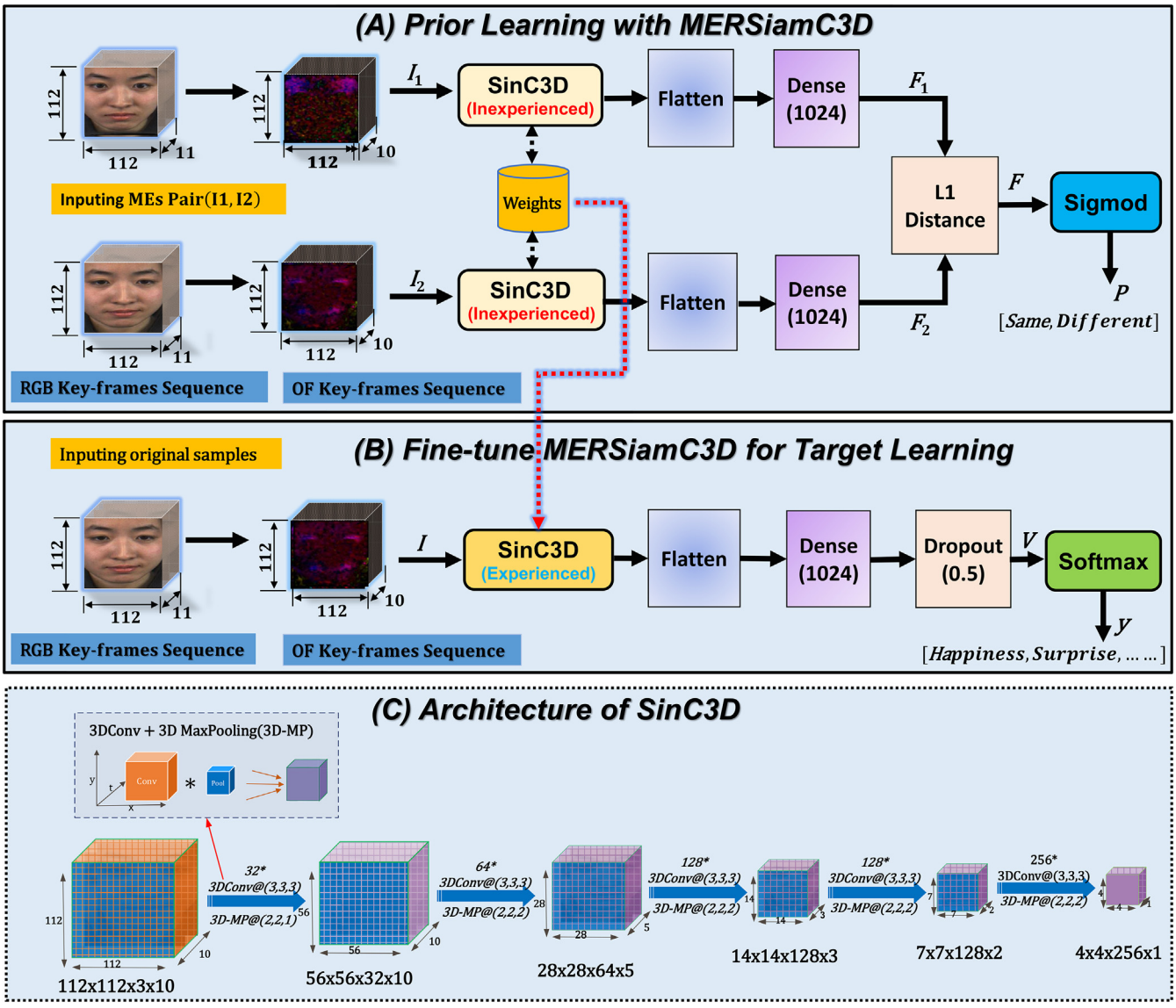


Fig. 3. Illustration of the MERSiamC3D framework for MEs features learning and classification. (A) is the architecture of MERSiamC3D and using MEs pair (I_1, I_2) as input for prior learning. (B) shows the target learning stage and corresponding model structure used, specifically, using the parameters obtained in the prior learning stage as the initial value of SinC3D and fine-tuning the middle layers for target learning and classification. (C) shows the process of feature mapping by SinC3D.

the cube formed by stacking multiple contiguous frames together [17]. As shown in Fig. 3(A), our MERSiamC3D model actually consists of twin networks whose parameters are tied together. For each network, it is mainly composed of five components: 1) a single C3D network (SinC3D); 2) a flatten layer; 3) a fully-connected layer (or dense layer); 4) a similarity measurement layer with L1-distance; and 5) an output layer with the sigmoid function. Among all of them, the SinC3D network is the core component of MERSiamC3D, which is shown in Fig. 3 (C). It consists of five 3D Convolutional (3DConv) layers and five corresponding 3D Max-Pooling (3D-MP) layers. Besides, we apply the ReLU activation function in the convolutional layers to get the output feature maps. Formally, the value at position (x, y, z) on the j-th feature map in the i-th layer could be formulated by:

$$v_{ij}^{xyz} = ReLU\left(b_{ij} + \sum_{m}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1}\sum_{r=0}^{R_i-1} w_{ijm}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)}\right), \tag{6}$$

where b_{ij} is the bias for this feature map, P_i, Q_i and R_i are the size of the 3D kernel along the X-space, Y-space and temporal dimension. While m indexes over the set of feature maps in the $(i-1)$-th layer connected to the current feature map, and w_{ijm}^{pqr} is the value at the position (p, q, r) of the kernel connected to the mth feature map.

3.2.1.2. The training of MERSiamC3D model. Before training MERSiamC3D model, a key step is to build the collection of positive and negative samples from the original datasets. To this end, we make sample pairs according to the original labels of the samples, i.e., if two samples are taken from the same category of MEs, they are treated as the "same" and labeled as "1", otherwise they are treated as "different" and labeled as "0". Supposing the original dataset have a total of K-categories and N samples, and the number of samples in each category is N_i, then the number of sample pairs M could be obtained through:

$$M = \frac{1}{2}\sum_{i=1}^{K}\left(C_{N_i}^1 C_{N_i-1}^1 + C_{N_i}^1 C_{N-N_i}^1\right). \tag{7}$$

Afterwards, during the training process, for an input sample pair $[I_1, I_2]$, the model will output two feature vectors F_1 and F_2 after the processing of SinC3D modules, flatten layers and dense layers like that of Fig. 3 (A). To determine whether F_1 and F_2 are

the "same" or "different", the L1 distance between F_1 and F_2 together with a sigmoid function are applied, which maps onto the interval $[0, 1]$. Then, the prediction vector could be computed through:

$$P(I_1, I_2) = sigmod\left(\sum_{j=1}^{D} | f_{1_j} - f_{2_j} |\right), \tag{8}$$

where D is the dimension of F_1 and F_2. Naturally, the binary cross entropy objective is a suitable choice for training the neural network.

3.2.2. Target learning and classification

After the procedure of prior learning, SinC3D, the core of MERSiamC3D, has mastered the basic experience to extract the general features for MEs. However, it is not enough for identifying and learning weak movements of MEs. To handle this issue, our work here proposes to continue training MERSiamC3D to obtain the high-level features for target classification.

3.2.2.1. Fine-tuning the MERSiamC3D model. For solving the multi-classification tasks, we first adjust the structure of MERSiamC3D as shown in Fig. 3(A) to that in Fig. 3(B). From Fig. 3(B), we could observe that the new architecture is identical to one of the MERSiamC3D except for the last two layers. Specifically, the SinC3D module is followed by a flatten layer, a fully-connected layer, a dropout layer, and an output layer sequentially. In the output layer, the softmax regression is choosen as the activation function. Formally, given a sample x and the output vector is $V = \{v_1, v_2, \ldots, v_n\}$, the probability of classifying x into class k is calculated as:

$$\tilde{y} = P(y = k/v_i) = \frac{e^{v_i}}{\sum_{j=1}^{K} e^{v_j}}, k \in [1, K], \tag{9}$$

where y is the ground-truth value of input v_i, and K is the number of total categories.

3.2.2.2. Training with the focal loss. For multiple classification tasks, the cross-entropy (CE) loss is usually used for back propagation to update model parameters. However, there is an imbalanced distribution of samples in the spontaneous MEs datasets. This could be biased toward particular emotions that constitute a larger portion of the training set. Therefore, applying a fairer loss function is critical.

Fortunately, according to [26], the focal loss is designed to address the one-stage object detection where there is an extreme imbalance between foreground and background classes during training, which has been widely used in object-detection tasks with excellent performance. Therefore, the aim of our work here is trying to introduce the focal loss to solve the unfair training problem caused by unbalanced MEs samples. The original definition of focal loss is shown as follows:

$$FL(p_t) = -\alpha_t(1 - p_t)^{\gamma} log(p_t), \tag{10}$$

where α_t is the weight balance factor for samples, γ is the balance factor for loss, and p_t is the binary classification probability distribution of the sample. To adapt for our multi-classification task, we modify the focal loss in this paper by combining Eq. (9) and (10), which is shown as:

$$FL(y, \tilde{y}) = -\sum_{i=1}^{K} \alpha_i (1 - y_i * \tilde{y}_i)^{\gamma} \log \tilde{y}_i, \tag{11}$$

where γ is set as 2 in practice, and α is treated as a hyper-parameter to set by cross validation.

4. Experiments

In this section, the validation experiments including experimental settings and results will be detailed, where the experimental settings consist of datasets description, implementation details, and the introduction of evaluation metrics.

4.1. Experimental settings

4.1.1. Datasets

To fit the problem we are dealing with, three popular spontaneous MEs datasets are used to evaluate the recognition performance of the proposed method, including SMIC–HS [25], CASME II [50] and SAMM [4]. The different characteristics of these datasets are summarized in Table 1. It is worth mentioning that these datasets are collected by high-speed cameras: the SMIC–HS is recorded at 100fps, the CASME II and SMIC are both recorded at 200fps. Besides, the SMIC–HS dataset has only one type of class labels, including positive (51), negative (70) and surprise (43), however, the CASME II and SMIC datasets have two kinds of class labels, i.e., *objective classes* and *original emotion classes*. For the former one, the two datasets have uniform classes I-VII, which are based on facial Action Units (AUs) with the bias of human reporting removed by Davison et al. [3]. Moreover, the classes I-VI are respectively linked with happiness, surprise, anger, disgust, sadness, and fear. While class VII relates to contempt and other AUs without emotional link in EMFACS [10]. The number of samples for each objective class in CASME II and SAMM datasets is shown in Table 2, from which we could easily see that the total size of two datasets is both small, with a distinct imbalance between the sample sizes of different classes. For the original emotion classes, the CASME II and SAMM datasets are usually categorized into five different classes: the samples in CASME II are divided into happiness (32), surprise (25), disgust (63), repression (27), and others (99), but for the SAMM, the samples are divided into happiness (26), anger (57), contempt (12), surprise (15), and others (26).

4.1.2. Implementation details

To comprehensively evaluate the performance of our method and compare it as much as possible with current other methods, two kinds of experiments were conducted: (1) *Five-classification experiments* on the CASME II and SAMM datasets with their objective classes and emotion classes. For the objective classes classification, the distribution of samples is shown in Table 2. Considering the extremely small sample size of class VI and the fact that class VII is ambiguous in terms of emotion, only classes I-V are selected for utilization. (2) *Three-classification experiment* fully refers to MEGC 2019 [40] on the SMIC–HS, CASME II, SAMM and their combination. At first, the original emotion classes of each dataset are grouped into three main categories: negative (including "repression", "anger", "contempt", "disgust", "fear" and "sadness"), positive ("happiness") and surprise. Then, the three datasets cap-

Table 1
Summary of characteristics for SMIC–HS, CASME II and SAMM datasets.

Characteristics	SMIC	CASME II	SAMM
Samples	164	255	159
Subjects	16	26	29
Ethnicities	3	1	13
FPS	100	200	200
Resolution	640 × 480	640 × 480	2040 × 1088
Facial Area	160 × 130	340 × 280	400 × 400
Emotion Classes	3	7	8
Objective Classes	–	7	7
FACS Coded	NO	YES	YES

Table 2
Distribution of samples based **objective classes** for both CASME II and SAMM.

Objective classes based on AUs	CASME II	SAMM
I (Happiness)	25	24
II (Surprise)	15	13
III (Anger)	99	20
IV (Disgust)	26	8
V (Sadness)	20	3
VI (Fear)	1	7
VII (Contempt and other AUs)	69	84
Total Size	255	159
Subjects	26	21

tured from different stimuli and environment are merged to form a single composite dataset. The summary of samples distribution for these three datasets and their combination are shown in Table 3.

In the preprocessing procedure, to avoid the interference of the non-facial area and the influence of the head posture, the basic processing over each frame is first conducted in MEs video to detect, align and crop face region by using the DLib toolkit [21] and the face alignment method described in [42], then resize the face region to 112×112 pixel resolution to match the input spatial dimension of our model.

In the model training procedure, our models are implemented on Keras with Tensorflow as the backend. For the parameter setting of MERSiamC3D, all of the 3D convolution filters are set as $3 \times 3 \times 3$ with stride $1 \times 1 \times 1$, and all of the 3D max-pooling layers are set as $2 \times 2 \times 2$ with stride $2 \times 2 \times 2$ except for the first layer which has a kernel size of $2 \times 2 \times 1$ and a stride of $2 \times 2 \times 1$ with intention of preserving the temporal information in the early phase. Additionally, the number of convolutional kernels corresponding to each layer is set as 32, 64, 128, 128, 256. For both prior learning and target learning stages, the stochastic gradient descent (SGD) is used as the optimizer with the momentum set to 0.9, and the learning rate is initialized with 0.004 which will decrease 10 times smaller after every 10 epochs in SGD. Meanwhile, the total epoch is set as 100. To be more specific, at the prior learning stage, 90% of the samples pairs are used as training set and the rest 10% are applied as testing set. Afterwards, the final convolutional layers parameters are kept for target learning. Then, at the target learning stage, the original labeled datasets are exploited as input, and the convolutional layers parameters obtained by the prior learning stage are used to initialize SinC3D module. In our experiments, the best recognition performance could be obtained by fixing the first two convolutional layers of SinC3D for retraining.

Additionally, data augmentation is also applied to alleviate over-fitting, and each frame of samples is first cut randomly with 2–5 pixels at different places, i.e., *up*, *bottom*, *left*, *right*, *center*, *upper left*, *upper right*, *lower left* and *lower right* part of the frame, then resize to its previous size by using bilinear interpolation. Besides, for each MEs sample, three new samples are constructed by discarding one or all of the first two frames and copying the last frame to fill them. As a result, 36 times more data combined with original samples could be obtained, which substantially alleviates the adverse effects brought by data sparsity.

4.1.3. Evaluation metrics

For the purpose of ensuring subject-independent evaluation, most of the existing methods adopt leave-one-subject-out (LOSO) strategy for evaluation [18,40,42,47]. In view of their practice, we also apply LOSO for evaluation in our experiments. That is to say, for each fold, all samples from one subject are used as a testing set and the rest for training. Therefore, there are 16 training and testing procedures for SMIC–HS dataset, 26 training and testing procedures for CASME II dataset, while the SAMM dataset has 21 (27) procedures for five-classification with objective (original emotion) five-classes and 28 procedures for three-classification, the composite 3-dataset has 68 procedures for three-classification. Following the previous studies, Accuracy (Acc) and F1-score are used here to measure the performance of the five-classification experiments. Differently, when experimenting on three-classification, we refer to MEGC 2019 using the unweighted F1-score (UF1) and unweighted average Recall (UAR) to evaluate the model performance. Actually, UF1 is also commonly known as the macro-averaged F1-score and UAR is the "balanced accuracy" [40]. Given the true positives (TP_c), false positives (FP_c) and false negatives (FN_c) for each class k (K classes in total) over N folds, UF1 and UAR could be calculated as:

$$UF1 = \sum_{i=1}^{K} UF1_i / K, \tag{12}$$

$$UAR = \sum_{i=1}^{K} ACC_i / K, \tag{13}$$

where we have:

$$UF1_i = \frac{2 * TP_i}{2 * TP_i + FP_i + FN_i}, \tag{14}$$

$$ACC_i = TP_i / N_i. \tag{15}$$

Table 3
Distribution of samples after being classified into 3-categories.

Classes	SMIC	CASME II	SAMM	Combined
Negative	70	88	92	250
Positive	51	32	26	109
Surprise	43	25	15	83
Total Size	164	145	133	442

4.2. Experimental results

4.2.1. Five-classification experiments

4.2.1.1. Objective classes classification. In this subsection, the effectiveness of our proposed method is demonstrated by comparing its recognition performance on the five objective classes classification with several recent state-of-the-art (SOTA) methods, which consist of hand-crafted methods and deep learning methods. The hand-crafted methods include LBP-TOP [54], 3DHOG [38], HOOF [1] and ELBP-TOP [13], in which LBP-TOP, 3DHOG and HOOF were also reproduced by Davison et al. [3] to recognize the objective class labels, while ELBP-TOP is an improvement based on LBP-TOP and recently proposed by Guo et al. [13]. The deep learning methods mainly include ResNet with attention by Wang et al. [46], and DSCNN model proposed by Khor et al. [18], which is also reproduced by this paper to recognize the five objective class labels. Accordingly, the comparison results are shown in Table 4.

Actually, when evaluating the recognition performance, F1-score is more objective and convincing because there are serious class imbalances in CASME II and SAMM [52]. From Table 4, it could be observed that the MERSiamC3D consistently achieves the highest F1-score of 0.81 on CASME II and 0.60 on SAMM, which significantly outperforms the comparison methods. Particularly, although the class imbalances on SAMM dataset are much more severe than that of CASME II, while the MERSiamC3D could gain a higher performance than any other baselines. The main cause for this phenomenon might be that the introduction of focal loss effectively solves the class imbalance problem, which will be discussed in detail in the following sections. Besides, the proposed method also gains the highest accuracy of 80.05% on CASME II and 64.03% on SAMM. Compared with other deep learning methods (i.e., DSCNN [18] and ResNet model with Attention [46]) in Table 4, despite the same challenges caused by the lack of large-

Table 4
Performance of comparison methods on five-classification experiment with objective classes.

MER methods	CASME II[a]		SAMM[a]	
	Acc (%)	F1-score	Acc (%)	F1-score
LBP-TOP [3] (repetition 2018)[b]	67.80	0.51	44.70	0.35
3DHOG [3] (repetition 2018)[b]	69.64	0.56	42.17	0.33
HOOF [3] (repetition 2018)[b]	69.53	0.51	34.16	0.22
ResNet + Attention [46] (2018)[c]	65.90	0.54	48.50	0.40
ELBP-TOP [13] (2019)[c]	79.55	0.66	63.44	0.48
DSCNN [18] (repetition 2019)c	72.68	0.74	59.91	0.49
MERSiamC3D (Ours)[c]	**80.05**	**0.81**	**64.03**	**0.60**

[a] CASME II and SAMM with the uniform objective classes I-V.
[b] The hand-crafted method.
[c] The deep learning-based method.

scale MEs data, the MERSiamC3D still achieves an improvement of 7.37%, 14.15% on CASME II, 4.12%, 15.53% on SAMM in accuracy, which also reflects the superiority of our proposed two-stage learning strategy.

4.2.1.2. Original emotion classes classification. Meanwhile, there are also some significant works [3,18,42,43] carrying out the five-classification experiment to recognize the original five emotion class labels of CASME II and SAMM datasets. Table 5 shows the recognition performance of our proposed method and others for this five-classification task.

As shown in Table 5, our MERSiamC3D also yields the highest recognition accuracy (81.89%) and F1-score (0.83) on the CASME II dataset among other SOTA methods, which mainly includes the AlexNet [23] model with apex frame as input, the SSCNN and DSCNN models proposed by Khor et al. [18], the knowledge distillation based method by Sun et al. [43], the TSCNN-I and TSCNN-II models by Song et al. [42]. However, for the SAMM dataset, the F1-score we obtained is 5% lower than the TSCNN-II model. By inferring the following two main possible reasons: (1) Subjectively, compared with TSCNN-II, although our MERSiamC3D considers more the temporal dynamic information of MEs, it does not deal with local spatial information of MEs in a more focused manner like TSCNN-II. (2) Objectively, the SAMM dataset has a smaller sample size than the CASME II, in which case our model is more prone to be overfitting. Nevertheless, compared with other methods in Table 5, our method is relatively competitive, which clearly proves the rationality and superiority of our method for MER.

4.2.2. Three-classification experiment

In this subsection, we fully follow the MEGC 2019 [40] and conduct a series of three-classification experiments on the SMIC–HS, CASME II, SAMM and their composite dataset. Especially for the last one, the samples in it come from different datasets collected from a diverse range of subjects under different experimental scenarios, which requires higher robustness of the model. Similar to the five-classification experiments, the best results achieved by our method are also compared with the SOTA methods in the MEGC 2019, and the comparison methods also include hand-crafted methods (i.e., LBP-TOP by Zhao et al. [54], Bi-WOOF by Liong et al. [28]) and deep learning-based methods (i.e., CapsuleNet by Quange et al. [44], OFF-ApexNet by Gan et al. [12], Dual-Inception Network by Zhou et al. [55], Shallow Triple Stream Three-dimensional CNN (STSTNet) by Liong et al. [27], Expression Magnification and Reduction (EMR) with adversarial training by Liu et al. [29]).

As illustrated in Table 6, our proposed MERSiamC3D yields competitive results on all datasets, particularly on the composite dataset and the CASME II dataset, successfully surpassing the top 4 [12,55,27,29] in MEGC 2019. Indeed, the MERSiamC3D achieves the highest UF1 of 0.8068 and the highest UAR of 0.7986 on the composite dataset, also gets the best performance (UF1-0.8818, UAR-0.8763) on the CASME II, which clearly proves the effectiveness and robustness of our method. In a more in-depth analysis, the most significant difference between our method and the top 4 methods is that they only use the apex frame to describe the MEs sequence, which indicates that when facing the same MEs datasets, they could utilize fewer parameters but deeper CNN-based models to extract the high-level spatial MEs feature. Moreover, many large-scale image datasets [5] can also be used to knowledge transfer, so as to alleviate the overfitting problem caused by the insufficient MEs samples. In contrast, our approach considers more temporal dynamic information of MEs and uses the key-frames sequence with more frames to describe the MEs sequence. Although MERSiamC3D faces a higher risk of overfitting, it has achieved corresponding improvements in the composite dataset and the CASME II dataset, which is mainly due to the effectiveness and rationality of our two-stage learning strategy and key-frame sequences representation method. However, our MERSiamC3D are slightly behind the Top 1 [29] on the SMIC and SAMM datasets, which is partly due to the finer but more complex preprocessing of that method. Additionally, compared with the CASME II dataset, the SMIC and SAMM datasets are more challenging, which mainly comes from two objective factors: (1) The samples of SMIC are captured by a slower frame rate and lower resolution, which means that some fine-grained MEs information is missing. (2) The SAMM dataset is more diverse because the samples are collected from more ethnicities and ages, which puts forward a higher requirement for models' generalization.

5. Ablation studies

To further validate the rationality and effectiveness of the design of our method, we perform a series of ablation studies, which could answer the following questions:

- Does the prior learning strategy work? How much does it contribute to the performance improvement?
- Does the focal loss work well in MER?
- Is the key-frames construction method reasonable? Does it work?

5.1. The effect of prior learning

In this study, extensive five-classification experiments with the objective class labels are conducted on the CASME II dataset to verify the proposed prior learning strategy effectiveness. To this end, we firstly compare the recognition performance between the MERSiamC3D model without a prior learning (PL) stage and transfer knowledge with prior learning stage. Then, the results are detail-

Table 5
Performance of comparison methods on **Five-classification experiment with original emotion classes**.

MER Methods	CASME II[a]		SAMM[a]	
	Acc (%)	F1-score	Acc (%)	F1-score
LBP-TOP [3] (repetition 2018)[b]	68.24	0.48	N/A[d]	N/A
AlexNet [23] (repetition 2019)[c]	62.96	0.67	52.94	0.43
SSCNN [18] (2019)[c]	71.19	0.72	56.62	0.45
DSCNN [18] (2019)[c]	70.78	0.73	57.35	0.46
Knowledge Distillation [43] (2020)[c]	72.61	0.67	N/A	N/A
TSCNN-I [42] (2019)[c]	74.05	0.73	63.53	0.61
TSCNN-II [42] (2019)[c]	80.97	0.81	**71.16**	**0.69**
MERSiamC3D (Ours)[c]	**81.89**	**0.83**	68.75	0.64

[a] CASME II with the emotion classes: happiness, surprise, disgust, repression and others. SAMM with the emotion classes: happiness, anger, contempt, surprise and others.
[b] The hand-crafted method.
[c] The deep learning-based method.
[d] N/A – no results reported.

Table 6
Performance of comparison methods on **Three-classification experiment.**

MER Methods	SMIC		CASME II		SAMM		3DB-combined	
	UF1	UAR	UF1	UAR	UF1	UAR	UF1	UAR
LBP-TOP [54] (2007)[a]	0.2000	0.5280	0.7026	0.7429	0.3954	0.4102	0.5882	0.5785
Bi-WOOF [28] (2018)[a]	0.5727	0.5829	0.7805	0.8026	0.5211	0.5139	0.6296	0.6227
CapsuleNet [44] (2019)[b]	0.5820	0.5877	0.7068	0.7018	0.6209	0.5989	0.6520	0.6506
OFF-ApexNet [12] (2019)[b]	0.6817	0.6695	0.8764	0.8681	0.5409	0.5392	0.7196	0.7090
Dual-Inception [55] (2019)[b]	0.6645	0.6726	0.8621	0.8560	0.5868	0.5663	0.7322	0.7278
STSTNet [27] (2019)[b]	0.6801	0.7013	0.8382	0.8686	0.6588	0.6810	0.7353	0.7605
EMR with adversarial training [29] (2019)[b]	**0.7461**	0.7530	0.8293	0.8209	**0.7754**	0.7152	0.7885	0.7824
MERSiamC3D (Ours)[b]	0.7356	**0.7598**	**0.8818**	**0.8763**	0.7475	**0.7280**	**0.8068**	**0.7986**

[a] The hand-crafted method.
[b] The deep learning method.

edly analyzed after fixing different convolutional layer parameters of the SinC3D network in the target learning stage.

As shown in Table 7, after applying the prior training strategy, the performance of various variants of our model is improved significantly except for the last one. Especially, after fixing the first two convolutional layers of the SinC3D network, our model could achieve the best performance, which is 6.78% higher in accuracy and 5.53% higher in F1-score than those models without prior learning. These results are quite in accordance with our assumptions, because fixing all convolution layers of our model means that all parameters of convolutional layers obtained at the prior learning stage will be thoroughly duplicated to the network of target learning stage, and will no longer be updated in model training at the latter stage. Consequently, the model's ability of extracting high-level MEs features will not be improved through the training on target datasets. On the contrary, the fixation of parameters in the first two convolutional layers and the initiation of parameters in other layers based on the corresponding parameters obtained at the prior learning stage can not only decrease the update of low-level parameters in target learning, but also enhance the model's ability to extract high-level MEs features, which will facilitate the target classification tasks.

As a summary, the significant improvement over accuracy and F1-score clearly proves the effectiveness of our proposed two-stage learning strategy (i.e., *prior learning* and *target learning*) for MER.

Table 7
Comparative results of different feature learning strategies.

Method	Fixed Convolution Layers of SinC3D	CASME II	
MERSiamC3D without PL[a]	—	73.27	0.7578
MERSiamC3D[b]	—	75.74 (↑ 2.47)	0.7728 (↑ 0.0150)
MERSiamC3D[b]	Conv1	77.93 (↑ 4.66)	0.7937 (↑ 0.0359)
MERSiamC3D[b]	Conv1,Conv2	**80.05 (↑ 6.78)**	**0.8131 (↑ 0.0553)**
MERSiamC3D[b]	Conv1,Conv2,Conv3	78.18 (↑ 4.91)	0.7969 (↑ 0.0391)
MERSiamC3D[b]	Conv1,Conv2,Conv3, Conv4	75.61 (↑ 2.34)	0.7704 (↑ 0.0126)
MERSiamC3D[b]	Conv1,Conv2,Conv3, Conv4,Conv5	58.79 (↓ 14.48)	0.5878 (↓ 0.1700)

[a] Without prior learning.
[b] After the prior learning, and the convolution layers' parameters are transferred from prior learning.

5.2. The effect of focal loss

To verify the effectiveness of focal loss in MER tasks, we utilize the standard cross-entropy (CE) loss and focal loss (FL) with different balance factor α to guide the training of MERSiamC3D respectively. Then, through the comparison of five-classification performance, the effect of the focal loss for MER could be verified. Intuitively, the results of our proposed MERSiamC3D with different loss functions are shown in Fig. 4.

As depicted in Fig. 4, it is obvious that the recognition performance is consistently and significantly improved on CASME II and SAMM datasets with objective classes when the focal loss is used and matched with the optimal balance factor α. Concretely, our model could indeed obtain the best results on CASME II when α is set as 7. However, setting the value of α to 8 to get the best

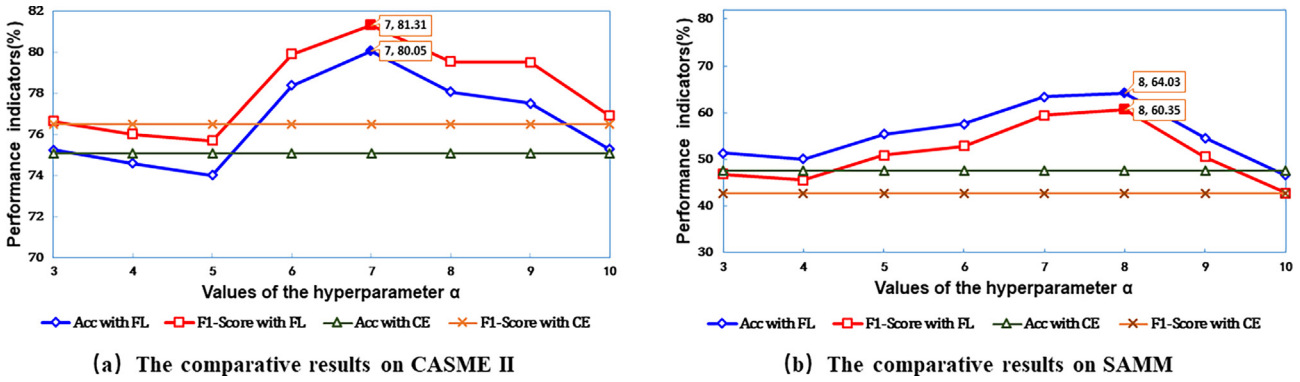


Fig. 4. Comparative results of different loss functions with various hyperparameters.

results on SAMM is what we need. Furthermore, by comparing the statistics in Table 2 and Fig. 4, we could find an interesting thing that the classification performance of our model is better when α is set close to the maximum class ratio, which is equal to the ratio of the largest single class samples size to the smallest single class samples size in the datasets. For example, the maximum class ratio of SAMM is 8, which is equal to the sample size (24) of category I divided by the sample size (3) of category V, and our model appears to get the best classification performance exactly when the balance factor α is set to 8. By combining the above analysis together, we can conclude that FL can effectively solve the problem of low recognition performance caused by imbalance samples in MER.

5.3. Analysis of adaptive construction of key-frames

5.3.1. The rationality of adaptive construction of key-frames

In this subsection, we conduct a rationality analysis on the key-frames sequence obtained by our method from the perspective of vision and movement. In fact, MEs are a facial reaction that could reflect the change of facial action caused by emotions. Therefore, the constructed key-frames sequence requires not only to summarize visually of the original MEs video but also need to be highly consistent with the movement of the original MEs sequence.

To specify our analysis, we take the "*Happiness-EP05-02*" sample in the CASME II dataset as an example. Note that, this sample is one specific MEs clip that contains 97 frames, of which the onset frame (25), apex frame (91), and offset frame (121) are three key-frames that have been located. Based on the three key-frames and our proposed adaptive construction method, the final 11 RGB key-frames sequence and its corresponding 10 optical-flow key-frames sequence are obtained and shown in Fig. 5. As expected, we could observe that the key-frames sequence obtained by the proposed adaptive construction method can well visually summarize the MEs of "Happiness". Besides, it could also be concluded that the motion of the apex frame (91) is the tensest one from the optical-flow sequence.

Meanwhile, as for the original MEs sequence and the key-frames sequence obtained by our method, to intuitively show the movements of their MEs, we use the first frame as the reference frame, and then sequentially calculate the OF amplitudes of every subsequent frames and the reference frame according to the method of Section 3.1.2. Last but not least, the motion curve is also shown in Fig. 6. Intuitively, the motion changing curves showed in Fig. 6(a) and (b) are highly consistent with each other, which reflects that the key-frames sequence obtained through the adaptive construction method in this paper is highly consistent with the original MEs sequence in terms of movement changes.

5.3.2. The effect of adaptive construction of key-frames

For further analysis, another experiment is carried out in this paper to explore how the adaptive construction method of key-

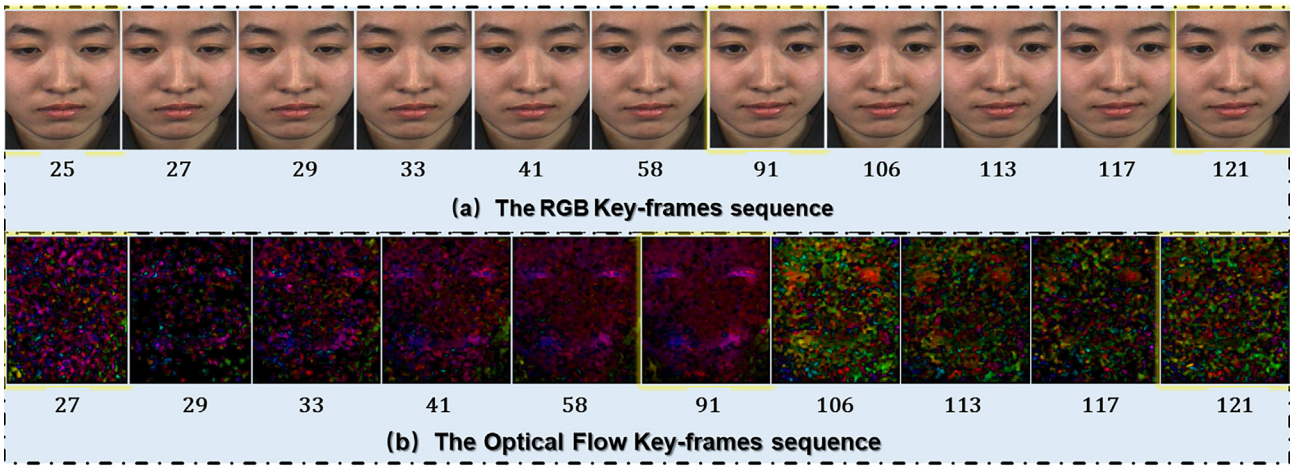


Fig. 5. The key-frames result by our adaptive construction method (the sample "Happiness-EP05-02" as input).

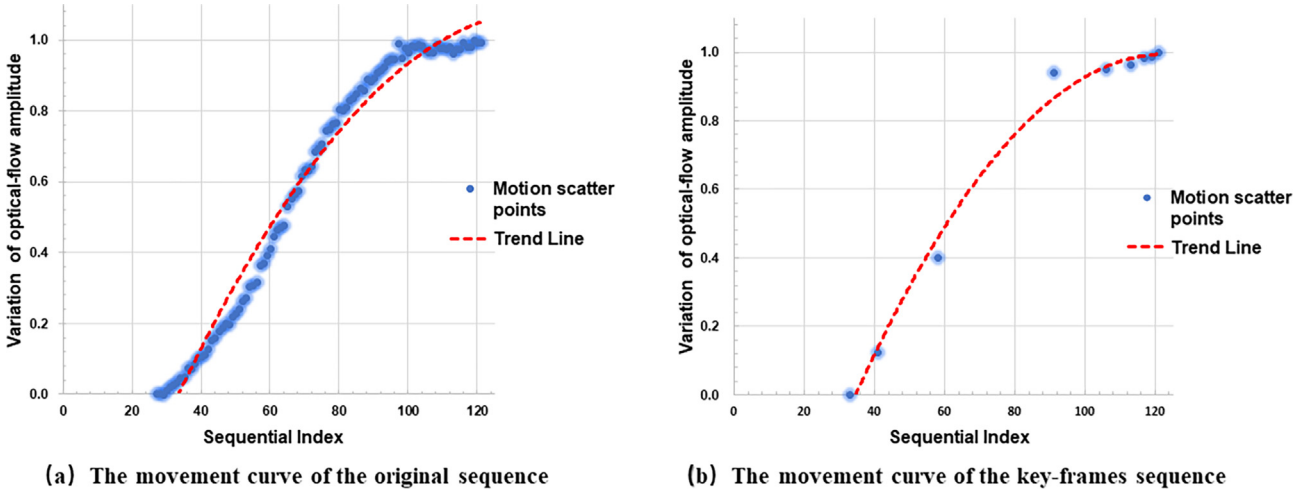


Fig. 6. Comparison of motion trends between the original micro-expression sequence and the key-frame sequence.

frames affects the final recognition performance. Specifically, three different methods are first applied to extract 11 RGB key-frames sequences based on the annotated three key-frames (i.e., onset-apex-offset frames). Then we conduct a series of five-classification experiments on the CASME II dataset. The three different key-frames extraction methods include time interpolation model (TIM) [56] adopted by most jobs [19,13,33], time adaptive sampling (TAS) and ours. It should be noted that TAS is quite similar to our method. Since both of the two methods first determine the time index of the key-frame according to the method described in Section 3.1.1, and then obtain the key-frame at the corresponding position. The main difference lies in that TAS directly takes the frame at the corresponding time index from the original ME sequence as the key-frame, while our method is to generate the intermediate transition frame at the corresponding position by using the adaptive convolutional neural network [31]. The comparison results are shown in Fig. 7, from which it could be observed that our adaptive construction method has achieved the best recognition performance. And it also proves the adaptive construction method of key-frames sequence can help boost the final recognition performance.

Nevertheless, since the proposed adaptive construction method for key-frames relies on manually annotated apex frame, which is not always accurate. Therefore, in the future, it is necessary to design an intelligent and more accurate method for apex frame locating and then construct key-frames.

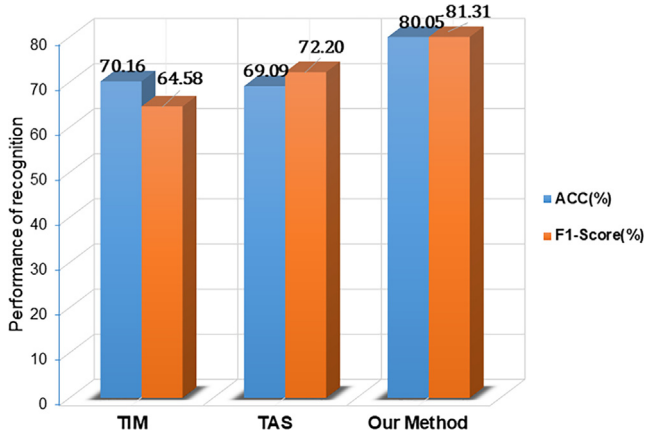


Fig. 7. Comparison of performance results obtained by different key-frames construction methods.

6. Conclusion

In this work, a novel two-stage learning method with MERSiamC3D model are proposed, which has significant advantages in tackling insufficient and imbalance samples problem in MER. Specifically, in order to effectively retain the spatiotemporal information of the original MEs video, an adaptive construction method was firstly proposed to construct the key-frames sequence to summarize the original MEs video, which has the ability of helping drop the redundant frames and highlight the movement of the apex frame. Afterwards, considering the shortages of directly utilizing current MEs samples, we choose to decompose the ordinary feature learning procedure into two-stage, i.e., *prior learning* and *target learning*, which means that we first divide the original dataset (*with key-frames*) into a collection of same and different sample pairs for training our MERSiamC3D to extract the generic features of MEs at the prior learning stage, and then fine-tune the MERSiamC3D's structure and parameters so as to introduce the focal loss for target learning and classification. Finally, through the evaluations on three publically available MEs datasets, we could find that the proposed method outperformed other deep learning and traditional methods. The extensive experimental results demonstrated the effectiveness and superiority of the proposed method.

In the future, we will continue our exploration to make efforts to design more granular and differentiated MEs features, which is the premise for achieving real-time, more accurate and efficient MER applications.

CRediT authorship contribution statement

Sirui Zhao: Conceptualization, Methodology, Software, Validation, Writing - original draft, Writing - review & editing, Visualization. **Hanqing Tao:** Software, Writing - review & editing. **Yangsong Zhang:** Methodology, Writing - review & editing. **Tong Xu:** Validation, Writing - review & editing. **Kun Zhang:** Writing - review & editing. **Zhongkai Hao:** Writing - review & editing. **Enhong Chen:** Supervision, Writing - review & editing.

Declaration of Competing Interest

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

Acknowledgments

The authors thank the reviewers for their insightful comments. This work was supported by the National Natural Science Found of China [Grant Nos.61727809, U1605251, 62072423 and 62076209]; The National Key Research and Development Project of China [Grant No.2017 YFB1002501].

References

[1] R. Chaudhry, A. Ravichandran, G. Hager, R. Vidal, Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 1932–1939.
[2] J.F. Cohn, Z. Ambadar, P. Ekman, Observer-based measurement of facial expression with the facial action coding system, The Handbook of Emotion Elicitation and Assessment 1 (2007) 203–221.
[3] A. Davison, W. Merghani, M. Yap, Objective classes for micro-facial expression recognition, J. Imag. 4 (2018) 119.
[4] A.K. Davison, C. Lansley, N. Costen, K. Tan, M.H. Yap, Samm: a spontaneous micro-facial movement dataset, IEEE Trans. Affect. Comput. 9 (2016) 116–129.
[5] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.
[6] P. Ekman, Facial expression and emotion, Am. Pychol. 48 (1993) 384.
[7] P. Ekman, Micro Expression Training Tool (mett) and Subtle Expression Training Tool (sett), Paul Ekman Company, San Francisco, CA, 2003.
[8] P. Ekman, Telling Lies: Clues to Deceit in the Marketplace, Politics, and Marriage, revised ed., 2009, WW Norton & Company..
[9] P. Ekman, W.V. Friesen, Nonverbal leakage and clues to deception, Psychiatry 32 (1969) 88–106.
[10] R. Ekman, What the Face Reveals: Basic and Applied Studies of Spontaneous Expression using the Facial Action Coding System (FACS), Oxford University Press, USA, 1997.
[11] G. Farnebäck, Two-frame motion estimation based on polynomial expansion, in: Scandinavian Conference on Image Analysis, Springer, 2003, pp. 363–370..
[12] Y. Gan, S.T. Liong, W.C. Yau, Y.C. Huang, L.K. Tan, Off-apexnet on micro-expression recognition system, Signal Process.: Image Commun. 74 (2019) 129–139.
[13] C. Guo, J. Liang, G. Zhan, Z. Liu, M. Pietikäinen, L. Liu, Extended local binary patterns for efficient and robust spontaneous facial micro-expression recognition, IEEE Access (2019).
[14] M.P. Haselhuhn, E.M. Wong, M.E. Ormiston, M.E. Inesi, A.D. Galinsky, Negotiating face-to-face: men's facial structure predicts negotiation performance, Leadership Quart. 25 (2014) 835–845.
[15] X. Huang, G. Zhao, X. Hong, W. Zheng, M. Pietikäinen, Spontaneous facial micro-expression analysis using spatiotemporal completed local quantized patterns, Neurocomputing 175 (2016) 564–578.
[16] L. Hunter, L. Roland, A. Ferozpuri, Emotional expression processing and depressive symptomatology: eye-tracking reveals differential importance of lower and middle facial areas of interest, Depress. Res. Treatment (2020)..
[17] S. Ji, W. Xu, M. Yang, K. Yu, 3d convolutional neural networks for human action recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2012) 221–231.
[18] H.Q. Khor, J. See, S.T. Liong, R.C. Phan, W. Lin, Dual-stream shallow networks for facial micro-expression recognition, in: 2019 IEEE International Conference on Image Processing (ICIP), IEEE, 2019, pp. 36–40.
[19] H.Q. Khor, J. See, R.C.W. Phan, W. Lin, Enriched long-term recurrent convolutional network for facial micro-expression recognition, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), IEEE, 2018, pp. 667–674..
[20] D.H. Kim, W.J. Baddar, Y.M. Ro, Micro-expression recognition with expression-state constrained spatio-temporal feature representations, in: Proceedings of the 24th ACM international conference on Multimedia, ACM, 2016, pp. 382–386..
[21] D.E. King, Dlib-ml: a machine learning toolkit, J. Mach. Learn. Res. 10 (2009) 1755–1758.
[22] G. Koch, R. Zemel, R. Salakhutdinov, Siamese neural networks for one-shot image recognition, in: ICML Deep Learning Workshop, 2015..
[23] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. (2012) 1097–1105.
[24] B. Lake, R. Salakhutdinov, J. Gross, J. Tenenbaum, One shot learning of simple visual concepts, in: Proceedings of the Annual Meeting of the Cognitive Science Society, 2011.
[25] X. Li, T. Pfister, X. Huang, G. Zhao, M. Pietikäinen, A spontaneous micro-expression database: inducement, collection and baseline, in: 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (fg), IEEE, 2013, pp. 1–6.
[26] T.Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2980–2988.
[27] S.T. Liong, Y. Gan, J. See, H.Q. Khor, Y.C. Huang, Shallow triple stream three-dimensional cnn (ststnet) for micro-expression recognition, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), IEEE, 2019, pp. 1–5..
[28] S.T. Liong, J. See, K. Wong, R.C.W. Phan, Less is more: micro-expression recognition from video using apex frame, Signal Process.: Image Commun. 62 (2018) 82–92.
[29] Y. Liu, H. Du, L. Zheng, T. Gedeon, A neural micro-expression recognizer, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), IEEE, 2019, pp. 1–4..
[30] Y.J. Liu, J.K. Zhang, W.J. Yan, S.J. Wang, G. Zhao, X. Fu, A main directional mean optical flow feature for spontaneous micro-expression recognition, IEEE Trans. Affect. Comput. 7 (2015) 299–310.
[31] S. Niklaus, L. Mai, F. Liu, Video frame interpolation via adaptive convolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 670–679.
[32] Y.H. Oh, J. See, A.C. Le Ngo, R.C.W. Phan, V.M. Baskaran, A survey of automatic facial micro-expression analysis: databases, methods, and challenges, Front. Psychol. 9 (2018) 1128.
[33] D. Patel, X. Hong, G. Zhao, Selective deep features for micro-expression recognition, in: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 2258–2263..
[34] M. Peng, C. Wang, T. Bi, Y. Shi, X. Zhou, T. Chen, A novel apex-time network for cross-dataset micro-expression recognition, in: 2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII), IEEE, 2019, pp. 1–6..
[35] M. Peng, C. Wang, T. Chen, G. Liu, X. Fu, Dual temporal scale convolutional neural network for micro-expression recognition, Front. Psychol. 8 (2017) 1745.
[36] W. Peng, X. Hong, Y. Xu, G. Zhao, A boost in revealing subtle facial expressions: a consolidated eulerian framework, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), IEEE, 2019, pp. 1–5..
[37] T. Pfister, X. Li, G. Zhao, M. Pietikäinen, Recognising spontaneous facial micro-expressions, in: 2011 International Conference on Computer Vision, IEEE, 2011, pp. 1449–1456.
[38] S. Polikovsky, Y. Kameda, Y. Ohta, Facial micro-expressions recognition using high speed camera and 3d-gradient descriptor, 2009.
[39] S. Porter, L. Ten Brinke, Reading between the lies: identifying concealed and falsified emotions in universal facial expressions, Psychol. Sci. 19 (2008) 508–514.
[40] J. See, M.H. Yap, J. Li, X. Hong, S.J. Wang, Megc 2019–the second facial micro-expressions grand challenge, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), IEEE, 2019, pp. 1–5..
[41] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, Adv. Neural Inf. Process. Syst. (2017) 4077–4087.
[42] B. Song, K. Li, Y. Zong, J. Zhu, W. Zheng, J. Shi, L. Zhao, Recognizing spontaneous micro-expression using a three-stream convolutional neural network, IEEE Access 7 (2019) 184537–184551.
[43] B. Sun, S. Cao, D. Li, J. He, L. Yu, Dynamic micro-expression recognition using knowledge distillation, IEEE Trans. Affect. Comput. (2020).
[44] N. Van Quang, J. Chun, T. Tokuyama, Capsulenet for micro-expression recognition, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), IEEE, 2019, pp. 1–7..
[45] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al., Matching networks for one shot learning, Adv. Neural Inf. Process. Syst. (2016) 3630–3638.
[46] C. Wang, M. Peng, T. Bi, T. Chen, Micro-attention for micro-expression recognition, 2018. arXiv, arXiv-1811..
[47] S.J. Wang, B.J. Li, Y.J. Liu, W.J. Yan, X. Ou, X. Huang, F. Xu, X. Fu, Micro-expression recognition with small sample size by transferring long-term convolutional neural network, Neurocomputing 312 (2018) 251–262.
[48] S. Weinberger, Intent to deceive? Can the science of deception detection help to catch terrorists? sharon weinberger takes a close look at the evidence for it, Nature 465 (2010) 412–416.
[49] Q. Wu, X. Shen, X. Fu, The machine knows what you are hiding: an automatic micro-expression recognition system, in: International Conference on Affective Computing and Intelligent Interaction, Springer, 2011, pp. 152–162.
[50] W.J. Yan, X. Li, S.J. Wang, G. Zhao, Y.J. Liu, Y.H. Chen, X. Fu, Casme ii: an improved spontaneous micro-expression database and the baseline evaluation, PloS One 9 (2014) e86041.
[51] W.J. Yan, Q. Wu, J. Liang, Y.H. Chen, X. Fu, How fast are the leaked facial expressions: the duration of micro-expressions, J. Nonverbal Behav. 37 (2013) 217–230.
[52] M.H. Yap, J. See, X. Hong, S.J. Wang, Facial micro-expressions grand challenge 2018 summary, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), IEEE, 2018, pp. 675–678..
[53] T. Zhang, Y. Zong, W. Zheng, C.P. Chen, X. Hong, C. Tang, Z. Cui, G. Zhao, Cross-database micro-expression recognition: a benchmark, in: IEEE Trans. Knowl. Data Eng., 2020.

[54] G. Zhao, M. Pietikainen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Trans. Pattern Anal. Mach. Intell. (2007) 915–928.
[55] L. Zhou, Q. Mao, L. Xue, Dual-inception network for cross-database micro-expression recognition, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), IEEE, 2019, pp. 1–5..
[56] Z. Zhou, G. Zhao, M. Pietikäinen, Towards a practical lipreading system, in: CVPR 2011, IEEE, 2011, pp. 137–144..

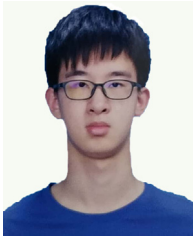
Sirui Zhao received the B.S. and M.S. degree in computer science and technology from the Southwest University of Science and Technology (SWUST) in 2014 and 2017. He is currently working toward the Ph.D. degree in the Department of Computer Science and Technology from University of Science and Technology of China (USTC). He is also a faculty member with SWUST. His research interests include automatic micro-expressions analysis, affect computing, human–computer interaction (HCI), meta learning, etc.

Hanqing Tao received the B.S. degree in electrical engineering and automation from China University of Mining and Technology, Xuzhou, China, in 2017. He is currently working toward the Ph.D. degree in the Department of Computer Science and Technology from University of Science and Technology of China (USTC). His research interests include data mining, deep learning, natural language processing, Chinese language analysis and interpretable artificial intelligence. He has published several papers in referred conference proceedings, such as AAAI, ICME, etc.

Yangsong Zhang received the Ph.D. degree in Signal and Information Processing from the School of Life Science and Technology, University of Electronic Science and Technology of China in 2013. He is currently an Associate Professor at the School of Computer Science and Technology, Southwest University of Science and Technology, China. His research interests include Brain-Computer Interface (BCI), biomedical signal processing, machine learning, etc.

Tong Xu received the Ph.D. degree in University of Science and Technology of China (USTC), Hefei, China, in 2016. He is currently working as an Associate Professor of the Anhui Province Key Laboratory of Big Data Analysis and Application, USTC. He has authored 50+ journal and conference papers in the fields of social network and social media analysis, including IEEE TKDE, IEEE TMC, IEEE TMM, KDD, AAAI, ICDM, etc.

Kun Zhang received the Ph.D. degree in computer science and technology from University of Science and Technology of China, Hefei, China, in 2019. He is currently a faculty member with the Hefei University of Technology (HFUT), China. His research interests include natural language processing, and text mining. He has published several papers in refereed conference proceedings such AAAI, KDD, ICDM. He received the KDD 2018 Best Student Paper Award.



Zhongkai Hao is currently pursuing the B.S. degree in the School of Gifted Young from University of Science and Technology of China, Hefei, China. His research interests mainly focus on data mining for scientific and social data, deep learning and statistical machine learning. He has published several papers in referred conference proceedings, such as NIPS, SIGKDD Conference on knowlegde discovery and data mining.

Enhong Chen received Ph.D. degree in computer science from University of Science and Technology of China (USTC) in 1996. He is currently a professor and the dean of the School of Data Science and vice dean of the School of Computer Science and Technology. His general area of research includes data mining and machine learning, social network analysis and recommender systems. He has published more than 100 papers in refereed conferences and journals, including IEEE Trans. KDE, IEEE Trans. MC, KDD, ICDM, NIPS, and CIKM. He was on program committees of numerous conferences including KDD, ICDM, SDM. He received the Best Application Paper Award on KDD-2008, the Best Student Paper Award on KDD-2018 (Research), the Best Research Paper Award on ICDM2011 and Best of SDM-2015. His research is supported by the National Science Foundation for Distinguished Young Scholars of China. He is a senior member of the IEEE.