A Novel Graph-TCN with a Graph Structured Representation for Micro-expression Recognition

Ling Lei
Chongqing Key Laboratory of Nonlinear Circuits and Intelligent Information Processing
School of Electronic and Information Engineering, Southwest University
Chongqing, China
leiling_swu@163.com

Jianfeng Li*
Chongqing Key Laboratory of Nonlinear Circuits and Intelligent Information Processing
School of Electronic and Information Engineering, Southwest University
Chongqing, China
popqlee@swu.edu.cn

Tong Chen
Chongqing Key Laboratory of Nonlinear Circuits and Intelligent Information Processing
School of Electronic and Information Engineering, Southwest University
Chongqing, China
c_tong@swu.edu.cn

Shigang Li
Graduate School of Information sciences
Hiroshima City University
Hiroshima, Japan
shigangli@hiroshima-cu.ac.jp

ABSTRACT

Facial micro-expressions (MEs) recognition has attracted much attention recently. However, because MEs are spontaneous, subtle and transient, recognizing MEs is a challenge task. In this paper, first, we use transfer learning to apply learning-based video motion magnification to magnify MEs and extract the shape information, aiming to solve the problem of the low muscle movement intensity of MEs. Then, we design a novel graph-temporal convolutional network (Graph-TCN) to extract the features of the local muscle movements of MEs. First, we define a graph structure based on the facial landmarks. Second, the Graph-TCN deals with the graph structure in dual channels with a TCN block. One channel is for node feature extraction, and the other one is for edge feature extraction. Last, the edges and nodes are fused for classification. The Graph-TCN can automatically train the graph representation to distinguish MEs while not using a hand-crafted graph representation. To the best of our knowledge, we are the first to use the learning-based video motion magnification method to extract the features of shape representations from the intermediate layer while magnifying MEs. Furthermore, we are also the first to use deep learning to automatically train the graph representation for MEs.

*Corresponding Author.

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413714

CCS CONCEPTS

•Computing methodologies → Computer vision; Image representations

KEYWORDS

Micro-expression recognition, Transfer learning, Graph-TCN, Graph representation

ACM Reference format:

Ling Lei, Jianfeng Li, Tong Chen, Shigang Li. 2020. A Novel Graph-TCN with a Graph Structured Representation for Micro-expression Recognition. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413714

1 INTRODUCTION

Micro-expressions (MEs) are brief, spontaneous and subtle movements of facial muscles. MEs are reflected in the local movements of the face, and the intensity of these movements is so low that it is difficult to distinguish their emotional types. Therefore, Micro-expressions recognition (MER) can be improved by magnifying these subtle facial movements. To solve this problem, a technique to magnify these delicate facial movements can be adopted. A method based on Eulerian motion magnification (EMM) [1] can be used to amplify MEs. [2], [3], [4] have demonstrated that ME recognition performance was improved through EMM amplification. Then, global Lagrangian motion magnification (GLMM) [5] was proposed and proved to be better than EMM at MER. However, these amplifying methods are based on

manual designs, leading to a lack of adaptability. In addition, they are prone to producing noises and excessive blurs. There are limitations to the handcrafted magnification methods for processing MEs. Therefore, we use transfer learning to apply a learning-based video motion magnification network [6] to ME amplification.

In the ME feature extraction process, the face is usually first divided into subregions according to the areas of the facial organs or the locations of the facial landmarks [7], [8]. Such processing can remove the redundant information with irrelevant noises and improve the efficiency for subsequent processing. A model or structure is usually designed to represent the face, and then the features are extracted based on this structure to obtain a feature representation. Common methods used in this step are the LBP-TOP [9], LBP-SIP [10], STLBP-IP [11], Hierarchical STLBP-IP [7], and DiSTLBP-RIP [12]. Methods based on optical flow are MDMO [13], sparse MDMO [14], Bi-WOOF [15], [16], and FHOFO [8]. Methods based on gradient are HOG and HIGO [2], [17].

The recent development of deep learning has also contributed to MER. Kim *et al*. [18] used deep learning to train the spatiotemporal feature representations of MEs. The spatial features are encoded by a CNN, while the temporal features are encoded by LSTM. Li *et al*. [4] used the Eulerian motion magnification method to magnify the apex frames and used the fine-adjusted VGG-FACE model for further recognition. Khor *et al*. [19] designed a DUAL-STREAM SHALLOW NETWORK (DSSN) based on the optical flow. They calculated three optical flow features. Through experiments on different MEs datasets, the optimal combination method was found, which can select two of the three optical flow features as the input to the dual network. Furthermore, they also find the best method to merge the two-channel outputs. Peng *et al*. [20] proposed a two-stream apex-time network to extract the spatiotemporal information of MEs. Here, the spatial stream uses the Res-10 network to extract the spatial information of the apex frame, while the temporal stream uses the LSTM network to extract the temporal information of the frames near the top frame. Peng *et al*. [21] use macro-expression datasets to pretrain a network and then use transfer learning to apply this network to MER. Xia *et al*. [22] proposed an STRCN, which can simultaneously pay attention to the spatial information and temporal changes in MEs. Verma *et al*. [23] proposed a dynamic image-based network named LEARNet to recognize MEs. First, they use a dynamic image process to transfer the video sample into a frame that contains the ME information from the video. Then, they feed the dynamic image into the Lateral Accretive Hybrid Network (LEARNet) for the training and classification tasks.

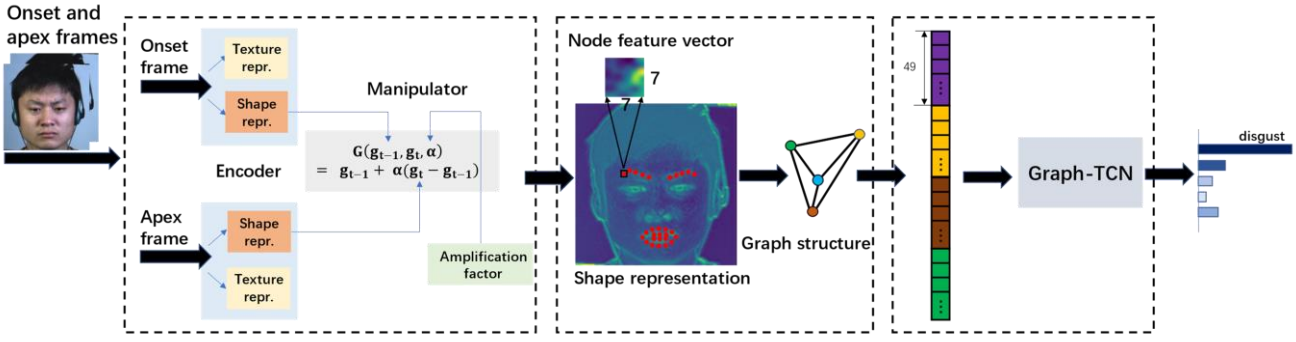
Recently, Zhong *et al*. [24] proposed the use of a graph structure to represent facial expressions in the field of static facial expression recognition. A graph structure refers to connecting the facial landmarks to each other to form a structure containing node values and edge weights. The graph representation can represent the texture feature information around the nodes and the geometric change information between these nodes. This feature representation can reduce the irrelevant noises to some extent and can be more discriminative. However, the node values and edge weights of the graph representation are calculated manually. This will lead to a lack of adaptability. Therefore, on this basis, we propose a network to obtain the node values and edge weights through training to obtain a graph representation.

Our contributions are mainly in the preprocessing and feature extraction processes. The details are as follows.

(1) We use learning-based video motion magnification based on transfer learning to enhance the low intensity of facial movements, while we extract the magnified shape features, which represent the shape variation of MEs, from the intermediate layer.

(2) Based on statistical analysis, we propose a facial graph structure for MEs by fusing shape information. This graph structure can better analyze the local muscle motion information of MEs and can be more discriminative.

(3) We design a novel Graph-TCN to train the graph representation, including one channel for node training and one channel



(a) (b) (c)

Figure 1: The framework. (a) is the magnification network without a decoder, which has the shape representation extracted from the middle layer (the manipulator) as the output. (b) is the process of converting a shape representation to a graph structure. Each facial landmark corresponds to a node in the graph, and every two points form an edge on the graph structure. For example, we use four nodes to illustrate the graph structure. (c) is the Graph-TCN used for training the graph representation and classification.

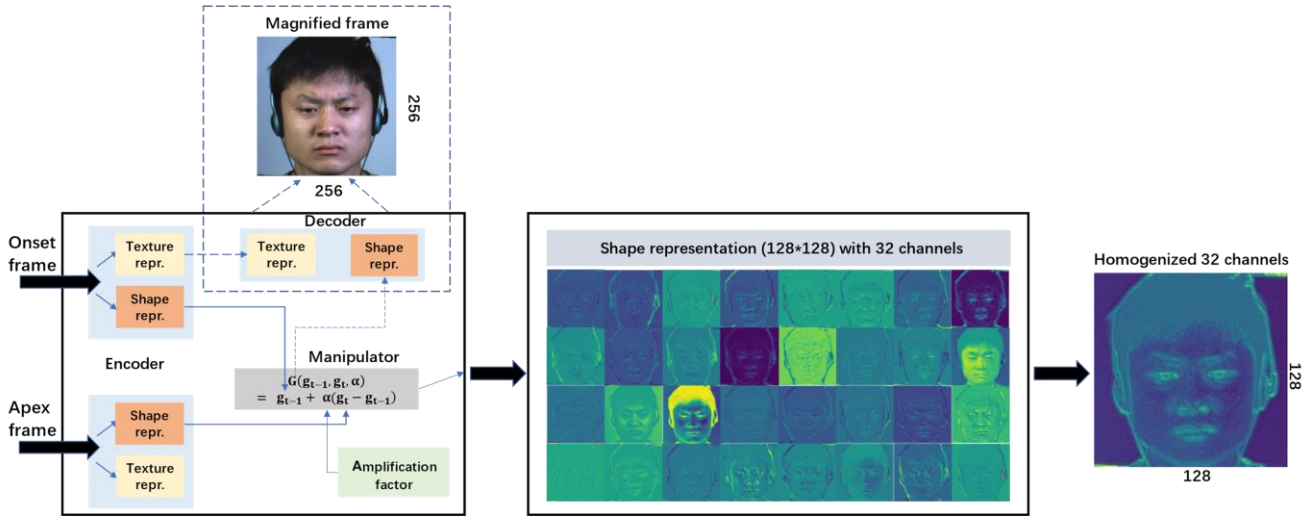


Figure 2: Extracting the shape representation through learning-based video motion magnification. (a) is the structure of the network, in which the part in the dashed line is the decoder that is not used in our task. (b) is the visualization of the 32 channels in the shape representation. (c) is the output of this process.

for edge training, which has better adaptability.

2 PROPOSED METHOD

The framework of our method is shown in figure 1. First, we input the onset frame and the apex frame of the ME sequence into the network (1a) for learning-based video motion magnification and extract the magnified shape representation (1b). Specific process is described in detail in section 2.1. Next, according to the location information of the facial landmark points, the eyebrows and mouth areas of the shape representation are made into a graph structure (1b). The graph structure is introduced in detail in section 2.2. Finally, the graph structure is input into the Graph-TCN network (1c) for feature extraction and recognition. The Graph-TCN network is given a detailed introduction in section 2.3.

2.1 Learning-based Video Motion Magnification Based on Transfer Learning

In the field of MEs, the onset frame is the beginning of an ME, and the apex frame is the stage where the ME is most pronounced. Furthermore, some recent studies such as [4], [15], [16], [20] have shown that the apex frames are sufficient to replace the ME sequences for feature extraction and classification. Therefore, we use the onset and apex frames as inputs. To capture the subtle movements in the amplification process, Oh *et al*. [6] proposed the Learning-based Video Motion Magnification. The network divides the input frames into shape and texture representations, and magnifies the shape representation. We apply this network to MEs through transfer learning. We use the pretrained network model for our amplification process. However, we do not use the amplified frame reconstructed in the decoder as the output. As shown in figure 2(a), we extract the magnified shape representation from the intermediate layer of the manipulator. We visualize the 32 channels of the magnified shape representation, which are shown in figure 2(b). After homogenizing 32 channels, magnified shape representation is used as original feature input (figure 2(c)).

The reasons we use the magnified shape representation of the intermediate layer are as follows. (1) The manipulator of the learning-based video motion magnification method only magnifies the shape representation. (2) The relevant literature [8], [25] has proved that the shape representation has been widely used in the field of MEs. Moreover, in the supplementary material of literature [6], there is a comparative analysis of the texture representation and shape representation. The shape representation tends to denote edges and boundaries that are of a geometric nature, while the texture representation tends to denote color properties. The shape representation contributes even more to the muscle movements' relationships, which we focus on. Texture contributes very little useful information to MEs and probably less than the effects of noise. (3) Directly extracting shape representations as features from the intermediate layer can reduce the probable errors caused by reconstruction.

In this section, we obtain a magnified facial shape representation, whose subtle muscle motions have been enhanced. We will introduce further feature extraction methods in the following sections.

2.2 Facial graph structure

In MER, we usually need to find more discriminative features. According to the studies of [7], [8], MER performs better when we extract the features from specific facial regions rather than the whole face. According to our statistical experiments, the eyebrow and mouth regions contribute significantly to MEs. This finding is also consistent with the heatmaps in literature [19], and more details will be shown in experiments part (3.3). Therefore, we pay more attention to the eyebrow and mouth areas that have more

significant contributions to MEs. However, this division of regions is still very rough for MEs. For refinement, we used a small window based on landmarks to locate the feature since the muscle movements of facial expressions are more concentrated around landmarks [24], let alone in MEs. As shown in figure 3(c), we locate 66 facial landmarks by using the DRMF [26]. We selected 28 landmarks of the eyebrows and mouth. Then, we take the 7x7 domain based on these landmarks. We select a window size of 7x7 through a comparison experiment (including 3x3, 5x5 and 7x7). See the experiment part (3.4) for details. In addition, we construct a facial graph structure model to represent MEs based on these 28 domains.

Graph structure. The graph structure consists of a node set and edge set, which can be shown as follows:

$$G = (N, E) \tag{2}$$

$$N = (n_1, n_2, n_3, \dots, n_i) \tag{3}$$

$$E = (e_{12}, e_{13}, e_{14}, \dots, e_{ij}) \tag{4}$$

Nodes (N) are selected according to some meaningful quantity, and edges (E) are formed between these nodes. Edge (e_{ij}) connects the node (n_i) and node (n_j). As shown in figure 3(d), the example graph structure has four nodes and six edges. The use of a graph structure can avoid redundant information but can also represent some complex data relationships.

Facial graph structure (FGS). Inspired by the work in [24], we use the FGS to represent MEs. As shown in figure 3, we illustrate the FGS using four landmarks of the magnified shape representation in figure 3(c). Figure 3(a) is a 7x7 domain of landmark which is represented as a node feature matrix. Figure 3(b) is a node feature vector (length is 49) compressed from a node feature matrix. The reason for compressing into vectors is designed to extract edge features. See the following section for specific reasons. As shown in figure 3(d), every two node feature vectors are joined to form an edge. For example, edge (e_{12}) connects the node (n_1) and node (n_2). In the FGS, each node represents the internal motion relationship of a muscle group, and an edge represents the motion relationships between these muscle groups. In addition, the occurrence of MEs will cause facial muscle groups to move. For different emotional types of MEs, the facial muscle groups have different patterns of motion. The node values and edge weights in the FGS will be different as well. In theory, the graph structure can be used for MER.

In this section, we create an FGS to represent the features of MEs, which can be shown as follows:

$$F_G = \{h(N),\ p(E)\} \tag{5}$$

F_G represents the FGS. $h(N)$ represents the extraction of the node features, and $p(E)$ represents the extraction of the edge features. We introduce how to get these features of the FGS in the next section.

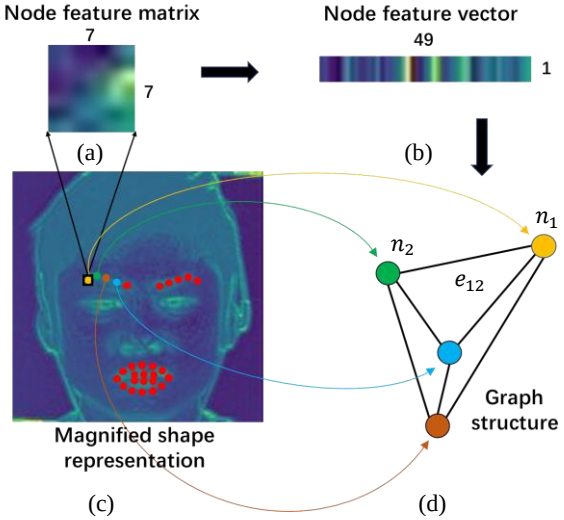


Figure 3: (c), (a), (b), and (d) show the process of building a graph structure. For the convenience of the introduction, we only use four nodes to describe the graph structure.

2.3 Graph-TCN

To adaptively extract the features of the nodes and edges in the graph structure, we design a novel Graph-TCN. By training the network, we obtain a more robust graph representation.

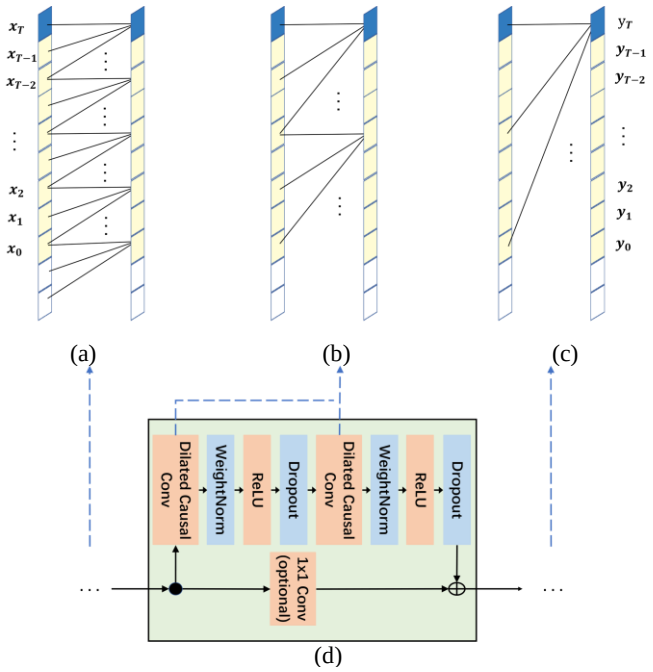


Figure 4: (a), (b), and (c) are the dilated casual convolutions [27]. Their kernel size k is 3, and their dilation factors d are 1, 2, and 4, respectively. For the sake of clarity, we left out some of the convolution kernels in the middle. (d) is a TCN residual block.

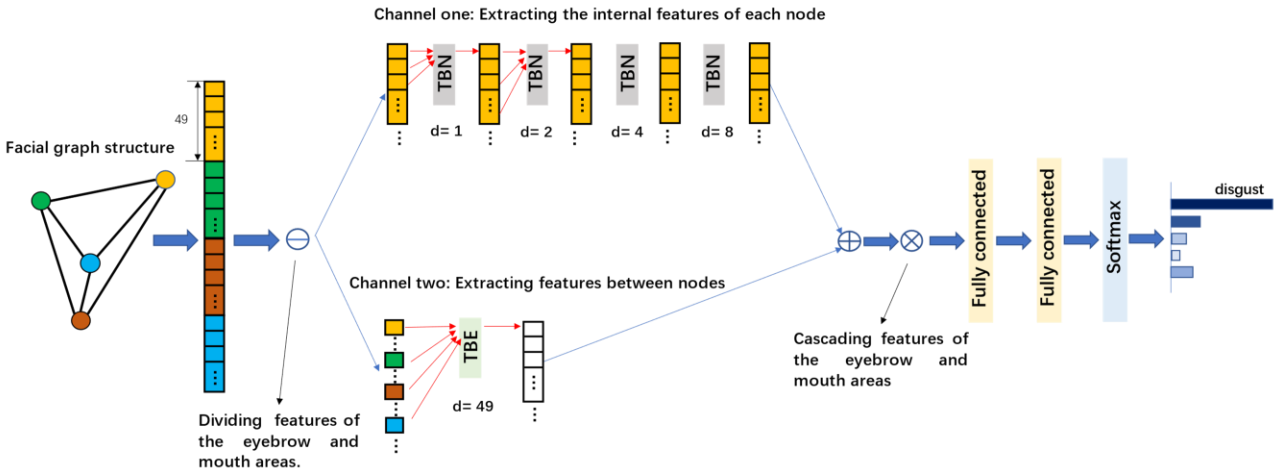


Figure 5: Architecture of the Graph-TCN. Channel one: the kernel size (red arrows) is 3. Channel two: the kernel size is 4.

TCN residual block. In the prediction task of exploring the sequence model, Bai *et al*. [27] proposed a TCN residual block to enable the task to be completed on a simple convolutional network framework. The basic components of the TCN residual block are the 1D fully convolutional network (FCN) and the causal convolutions. The FCN with zero padding can ensure that its input and output are the same length. Causal convolutions can make the output at time t merely the result of the convolution of the element at time t with the previous elements. Then, on the basis of the two methods, a dilated causal convolution is used to increase the scope of involvement of the previous elements. The dilated causal convolution has a receptive field that grows wider with the number of layers. The dilation factor d represents a fixed step between the elements involved in the convolution, which can determine the size of the receptive field. As seen in figure 4(a), when d=1, the dilated convolution represents the regular convolution. Moreover, the receptive field is increased in the second layer. In some of the longer sequence models, we need a wider receptive field, which means deeper network layers. Therefore, the residual connections that are often used in deeper networks (ResNet) [28] are used in this network block. In this TCN residual block, as shown in figure 4(d), two layers of dilated convolutions have been used. As with other convolution networks, ReLU, weight normalization and dropout are also used after each dilated convolution to make the network easier to train.

TBN and TBE. The above introduction to the TCN residual block is the case that the size of the convolution kernel remains unchanged. In other words, we can obtain a relatively flexible receptive field when the kernel size k and dilation factor d are changed. This structure makes it possible to extract the features of points and edges. The specific operation is as follows. In our sample sequence model, the length of a node sequence is 49. As shown in figure 5, the TCN residual block of channel one can be used to convolve the elements that are inside one node sequence, which can extract the node features. We call this block the TBN (TCN residual block of a node). In addition, we set d to a constant value of 49, which is same as the length of a node sequence. This setting allows the TCN residual block of channel two to convolve the elements that are from more than one node sequence at the same time, which can extract the edge features. We call this block the TBE (TCN residual block of an edge). The TBE and TBN can respectively train the node feature and edge feature to obtain the graph representation instead using the previous artificial methods. In addition, our network block, which is based on the TCN residual block, has the following advantages [27]. (1) Our network block has a flexible receptive field, which was previously mentioned and is the most critical aspect of our method. (2) Because the TCN residual block shares the convolution filter across the layers, less memory is required for training than RNN-based networks. (3) Because the back-propagation path of the TCN residual block is different from the time direction of the sequence, the exploding gradient problem in RNNs will not occur.

Graph-TCN. The framework of the Graph-TCN is shown in figure 5, and we take the graph structure with four nodes as an example to introduce the Graph-TCN. First, we squash the 7x7 region of each node into a one-dimensional vector, and then we cascade all the node vectors. Therefore, the graph structure is transformed into a one-dimensional vector, which is also a sequence model. Next, because the eyebrow areas and the mouth areas are relatively independent muscle groups and we are more concerned with the local motion relationships, we divide the graph structure of the face into two parts. Each part will use a dual-channel network to train the respective features of the graph structure. Channel one uses four TBN layers to extract the node features, and channel two uses a TBE layer to extract the edge features. Then, we add the features of the two channels together and cascade the graph representation of the two parts. We put the results into two fully connected layers, BatchNorm, Dropout, and ReLU for further feature training. At last, we use a softmax layer for classification.

3 EXPERIMENTS

3.1 The Datasets and Preprocessing

To evaluate our proposed approach, we conduct experiments using two databases: CASME II [29] (the most commonly used) and SAMM [17] (a relatively new database). They both have a high frame rate (200 fps) and the markers of the apex frame. The subjects in the CASME II were all Chinese and thus are ethnically homogeneous, while the subjects of the SAMM are ethnically diverse. The CASME II database contains 255 ME samples from 26 subjects. We select five categories from the database. They are happiness (32 samples), disgust (63 samples), repression (27 samples), surprise (25 samples) and other (99 samples), which results in a total of 246 samples. The SAMM database contains 159 ME samples from 32 subjects. We also select five categories from the database. They are anger (57 samples), happiness (26 samples), contempt (12 samples), surprise (15 samples) and other (26 samples), which results in a total of 136 samples. After reorganizing the database, the number of subjects in the SAMM is 27.

To better recognize MEs, we adopted alignment and resizing processes, and finally made the image size 256*256. The size meets the input requirement of the learning-based video motion magnification.

Because the samples are unbalanced and the number is too small, we applied data augmentation to both databases. We first crop the image and enlarge it to 256*256, where the four corners and the center of the image are used as the basis for cropping.

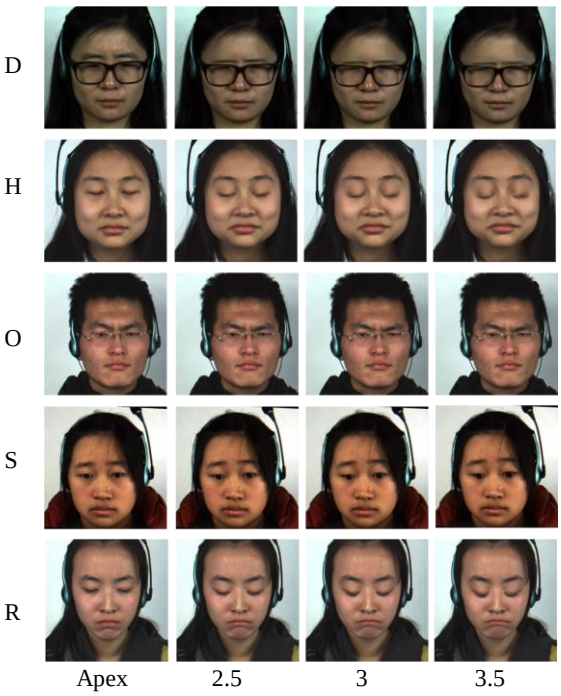


Figure 6: Magnified results. D, H, O, S and R refer to disgust, happiness, other, surprise and repression, respectively. The horizontal axis is the apex frames and the output images with magnification factors of 2.5, 3, 3.5.

This step adjusts the sample size of each category to be the same. Then, we use the magnification network to further expand the database, where the amplification factor is set as 1.2, 1.4, 1.6...3.0, respectively. The samples with an amplification factor of 3 are used as the benchmarks, and the other magnified samples are used as the expansion samples. The reason why we choose the amplification factor for this range will be explained in the next section.

3.2 Experiment on the Amplification Factor

To evaluate the amplification effect of magnification networks based on transfer learning on MEs, we conducted a visual comparison experiment on the CASME II. To get a better view, we used a reconstructed magnified frame instead of a shape representation. We transfer using the pretrained model, and we set the size of the amplification factor as 2.5, 3, 3.5 for the comparison experiment. The results for each category of MEs are shown in figure 6. The apex frames are the benchmarks. With the increase of the amplification factor, the muscle movement intensity of MEs gradually increases. The magnification effect of the eyebrow and mouth areas is more obvious. However, when the amplification factor is greater than 3, the facial muscle movements begin to excessively deform. Therefore, we choose an amplification factor of less than 3 (including 3) to magnify the expression and expand the samples.

3.3 Experiment on the RoMC

To make the feature representation more discriminative and less noisy, we conduct a statistical analysis to find the regions of most contribution (RoMC). The experiment is conducted on the CASME II. For the facial landmarks, we have removed the landmarks along the contour of the face and in the eye region. The reason is that Liong *et al*. [30] think that blinking is a natural action of the eyelids and should not be considered as an ME. If the eye region is included in the statistics, it can cause interference. Plus, the visualization of the activation maps of the final conv blocks from the DSSN [19] shows that the activations of the eye areas are also very small. Therefore, we remove the eye areas to eliminate the effect of blinking. According to the residual facial

Figure 7: Top 13 regions of the five categories of MEs

landmark points, we divide the face into suspicious regions for the onset frame and apex frame. Each region is 64*64 pixels in size. Then, we cascade each region into a feature vector and calculate the Pearson correlation coefficient of each corresponding region on the onset frame and the apex frame. The criterion for regional ranking is that the smaller the Pearson correlation coefficient is, the higher the ranking. The smaller the Pearson correlation coefficient between the two regions is, the greater the difference between them, which represents greater muscle movement intensity in the region. As shown in figure 7, we obtain the top 13 regions with the highest intensity of muscle movements. These regions are called the RoMC, which mainly contain the eyebrow and the mouth areas.

3.4 Experiment on the selection of the landmark window size

In order to select the appropriate window size of landmark, we conduct a group of comparative experiments on the CASME II. The test size is 3x3, 5x5, and 7x7. According to the window size, we extract 28 landmark domain matrices of eyebrows and mouth. 28 domain matrices are compressed into vectors respectively and cascade into a feature representation vector. Due to graph structure analysis is not involved here, we adopted the ten-fold cross validation and SVM classifier. The results are shown in table 1. The best window size is 7x7.

Table 1: Experiments on Graph-TCN parameter settings.

Window size	Accuracy
3x3	51.48%
5x5	54.48%
7x7	55.23%

3.5 Experiment on the Graph-TCN

We evaluated our proposed approach on two databases, the CASME II and SAMM, separately. We adopt the leave-one-subject-out (LOSO) protocol, which is commonly used in evaluating MER. The metric for calculating the accuracy rate is as follows:

$$acc = \frac{T}{N} \times 100\% \tag{6}$$

In equation 6, T is the total number of correct predictions and N is the total number of testing samples. Since the datasets are unbalanced, we use the F1-score to evaluate the recognition performance, which is calculated as follows:

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \tag{7}$$

$$P = \frac{1}{c} \sum_{i=1}^{c} \frac{tp_i}{tp_i + fp_i} \tag{8}$$

$$R = \frac{1}{c} \sum_{i=1}^{c} \frac{tp_i}{tp_i + fn_i} \tag{9}$$

In equations (7), (8), and (9), P is the precision, R is the recall, and c is the number of classes. Our experimental environment is a desktop with an NVIDIA GeForce GTX1080 Ti graphics card and 16 GB of RAM. The parameters are listed in Table 2, which are almost the same with TCN [27]. For obtaining the most appropriate layer numbers of our proposed Graph-TCN, a comparative experiment on layer numbers is conducted (Table 3). The best setting is that layer numbers of TBN is 4 while layer numbers of TBE is 1.

Table 2: Graph-TCN parameter settings.

Key parameters	Values
kernel size	7
dilation factor (TBN)	2^i
dilation factor (TBE)	49
hidden units per layer	25
optimizer	Adam
learning rate	0.0002

*2^i means that the value changes exponentially.

Table 3: Experiments on layer numbers of Graph-TCN.

Layers settings	Accuracy
TBN: 2, TBE: 1	73.58%
TBN: 2, TBE: 2	72.76%
TBN: 4, TBE: 1	73.98%
TBN: 4, TBE: 2	72.36%

Table 4: Comparison to recent state-of-the-art approaches on the CASME II. (5 classes)

Methods	Descriptors	Accuracy	F1-score
Khor *et al*. [31] (2011)	LBP-TOP	39.68%	35.89%
Khor *et al*. [32] (2012)	AlexNet	62.96%	66.75%
Kim *et al*. [18] (2016)	CNN+LSTM	60.98%	N\A
Liong *et al*. [16] (2017)	Bi-WOOF+Phase	62.55%	65.00%
Li *et al*. [4] (2018)	MagGA	63.30%	N\A
Zong *et al*. [7] (2018)	Hier. STLBP-IP	63.97%	61.25%
Liu *et al*. [14] (2018)	Sparse MDMO	66.95%	69.11%
Li *et al*. [2] (2018)	HIGO+Mag	67.21%	N\A
Huang *et al*. [12] (2019)	DiSTLBP-RIP	64.78%	N\A
Peng *et al*. [3] (2019)	ME-Booster	70.85%	N\A
Khor *et al*. [19] (2019)	DSSN	70.78%	**72.97%**
Khor *et al*. [19] (2019)	SSSN	71.19%	71.51%
ours	**Graph-TCN**	**73.98%**	**72.46%**

*N\A means there is no data reported in the original literature.

In the comparison experiments with other methods, we address some important studies from recent years that conducted five-class experiments on the CASME II and SAMM. [14], [16], [19] are modified methods based on the optical flow. [7], [12] are LBP-based approaches. [2], [3], [4] use the Eulerian motion magnification method to magnify the low intensity of subtle facial movements. [4], [19], [18] use a convolutional neural network to recognize MEs. The feature extraction in our method includes motion magnification, extracting a magnified shape representation, and training the graph representation, which all use the deep learning method. In addition, we are the first to extract the magnified shape representation from the intermediate layer and propose a novel Graph-TCN to train the graph representation.

Table 5: Comparison to recent state-of-the-art approaches on the SAMM. (5 classes)

Methods	Accuracy	F1-score
LBP-TOP [31] (2011)	39.68%	35.89%
AlexNet [32] (2012)	52.94%	42.60%
SSSN [19] (2019)	56.62%	45.13%
DSSN [19] (2019)	57.35%	46.44%
Graph-TCN	**75.00%**	**69.85%**

Table 6: Comparison to recent state-of-the-art approaches on the SAMM. (4 classes)

Methods	Accuracy	F1-score
LBP-TOP [31] (2011)	41.50%	40.60%
LBP-SIP [10] (2014)	41.70%	40.20%
CNN-GRU [33] (2014)	45.20%	44.10%
LBP-TICS [34] (2015)	39.50%	37.40%
STLBP-IP [11] (2015)	56.80%	52.70%
CNN-LSTM [35] (2015)	44.80%	43.70%
Image-based CNN [36] (2017)	43.60%	42.90%
Bi-WOOOF [15] (2018)	59.80%	59.10%
STRCN-A [22] (2020)	54.50%	49.20%
STRCN-G [22] (2020)	78.60%	74.10%
Graph-TCN	**80.50%**	**76.57%**

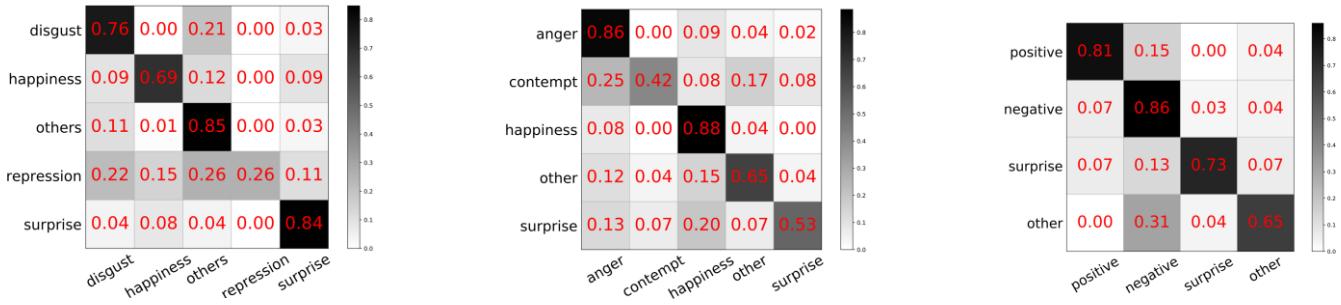
The comparison results (5 classes) are shown in table 4 and table 5. The confusion matrices (5 classes) are shown in figure 8(a) and 8(b). The latest five-class experiment based on the CASME II and SAMM is [19]. The DSSN and SSSN are their proposed methods. Khor *et al*. applied the LBP-TOP and AlexNet [32] to these two databases, and the results are reported in TABLE 4 and TABLE 5, respectively. Compared to the SSSN, our proposed method improves the accuracy by 3.92% (CASME II) and 30.78% (SAMM), respectively. Compared to the DSSN, our F1-score is not the highest on the CASME II. By comparing the confusion matrices, we found that the recognition for the repression class is less than that of the DSSN. However, compared to the other four classes, our method performs better than the DSSN. In addition, on the SAMM, our recognition performance is obviously higher than that of the DSSN. Since the comparison data of the five classes on SAMM are less, we add the comparison experiment of four classes. We classify the dataset into four classes according to [22] and still use LOSO protocol for the experiment. The comparison results (SAMM 4 classes) are shown in table 6 and the confusion matrices (SAMM 4 classes) are shown in figure 8(c). The data in table III are from [22]. Above all, our proposed method currently has the highest accuracy rate on both databases.

4 CONCLUSION

In this paper, we use learning-based video motion magnification to amplify MEs and extract the shape representations for subsequent processing. This strategy uses transfer learning rather than a hand-based approach to increase the intensity of MEs. To make the features of MEs more discriminative and the amount of data in the calculation smaller, we use statistical analysis to obtain the RoMC. According to the facial landmarks and RoMC, we build a graph structure based on the shape representation. Then, we design a novel Graph-TCN to extract the node and edge features, which can form the graph representation, and conduct classification. This is the first time we have used deep learning instead of manual calculations to obtain the node and edge features of a graph structure. We conduct experimental evaluations using the CASME II and SAMM datasets, and the results for both show that we obtain the best accuracy to date compared to those of other methods.

ACKNOWLEDGMENTS

This work was supported in part by the Fundamental Research Funds for the Central Universities (XDJK2020C016).



(a) CASME II (5 classes) (b) SAMM (5 classes) (c) SAMM (4 classes)

Figure 8: Confusion matrices

REFERENCES

[1] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William Freeman. 2012. Eulerian video magnification for revealing subtle changes in the world. *ACM Transactions on Graphics*. 31, 4 (July 2012), 1–8.

[2] Xiaobai Li, Xiaopeng Hong, Antti Moilanen, Xiaohua Huang, Tomas Pfister, Guoying Zhao, and Matti Pietikäinen. 2018. Towards Reading Hidden Emotions: A Comparative Study of Spontaneous Micro-Expression Spotting and Recognition Methods. *IEEE Transactions on Affective Computing*. 9, 4 (2018), 563-577.

[3] Wei Peng, Xiaopeng Hong, Yingyue Xu, and Guoying Zhao. 2019. A Boost in Revealing Subtle Facial Expressions: A Consolidated Eulerian Framework. In *Proceedings of IEEE International Conference on Automatic Face & Gesture Recognition*. 1-5.

[4] Yante Li, Xiaohua Huang, Guoying Zhao. 2018. Can Micro-Expression be Recognized Based on Single Apex Frame? In *Proceedings of IEEE International Conference on Image Processing*. 3094-3098.

[5] Anh Cat Le Ngo, Alan Johnston, Raphael C.-W. Phan, John See. 2018. Micro-Expression Motion Magnification: Global Lagrangian vs. Local Eulerian Approaches. In *Proceedings of IEEE International Conference on Automatic Face & Gesture Recognition*. 650-656.

[6] Tae-Hyun Oh, Ronnachai Jaroensri, Changil Kim, Mohamed Elgharib, Fr'edo Durand, William T. Freeman, and Wojciech Matusik. 2018. Learning-Based Video Motion Magnification. In *ECCV*. 663–679.

[7] Yuan Zong,Xiaohua Huang, Wenming Zheng, Zhen Cui, and Guoying Zhao. 2018. Learning From Hierarchical Spatiotemporal Descriptors for Micro-Expression Recognition. *IEEE Transactions on Multimedia*. 20, 11 (2018), 3160-3172.

[8] S L Happy and Aurobinda Routray. 2019. Fuzzy Histogram of Optical Flow Orientations for Micro-Expression Recognition. *IEEE Transactions on Affective Computing*. 10, 3 (2019), 394-406.

[9] Guoying Zhao and Matti Pietikäinen. 2007. Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 29, 6 (2007), 915-928.

[10] Yandan Wang, John See, Raphael C.-W. Phan, and Yee-Hui Oh. 2014. LBP with Six Intersection Points: Reducing Redundant Information in LBP-TOP for Micro-expression Recognition. In *ACCV*. 525–537.

[11] Xiaohua Huang, Su-Jing Wang, Guoying Zhao, and Matti Piteikäinen. 2015. Facial Micro-Expression Recognition Using Spatiotemporal Local Binary Pattern with Integral Projection. In *Proceedings of IEEE International Conference on Computer Vision Workshop*. 1-9.

[12] Xiaohua Huang, Su-Jing Wang, Xin Liu, Guoying Zhao, Xiaoyi Feng, and Matti Pietikäinen. 2019. Discriminative Spatiotemporal Local Binary Pattern with Revisited Integral Projection for Spontaneous Facial Micro-Expression Recognition. *IEEE Transactions on Affective Computing*. 10, 1 (2019), 32-47.

[13] Yong-Jin Liu, Jin-Kai Zhang, Wen-Jing Yan, Su-Jing Wang, Guoying Zhao, and Xiaolan Fu. 2016. A Main Directional Mean Optical Flow Feature for Spontaneous Micro-Expression Recognition. *IEEE Transactions on Affective Computing*. 7, 4 (2016), 299-310.

[14] Yong-Jin Liu, Bing-Jun Li, and Yu-Kun Lai. 2018. Sparse MDMO: Learning a Discriminative Feature for Spontaneous Micro-Expression Recognition. *IEEE Transactions on Affective Computing*. doi: 10.1109/TAFFC.2018. 2854166.

[15] Sze-Teng Liong, John See, KokSheik Wong, and Raphael C.-W. Phan. 2018. Less is more: Micro-expression recognition from video using apex frame. *Signal Processing: Image Communication*. 62 (2018) 82–92.

[16] Sze-Teng Liong and KokSheik Wong. 2017. Micro-expression recognition using apex frame with phase information. In *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*. 534-537.

[17] Adrian K. Davison, Cliff Lansley, Nicholas Costen, Kevin Tan, and Moi Hoon Yap. 2018. SAMM: A Spontaneous Micro-Facial Movement Dataset. *IEEE Transactions on Affective Computing*. 9, 1 (2018), 116-129.

[18] Dae Hoe Kim, Wissam J Baddar, and Yong Man Ro. 2016. Micro-Expression Recognition with Expression-State Constrained Spatio-Temporal Feature Representations. In *Proceedings of ACM international conference on Multimedia*. 382–386.

[19] Huai-Qian Khor, John See, Sze-Teng Liong, Raphael C. W. Phan, and Weiyao Lin. 2019. Dual-stream Shallow Networks for Facial Micro-expression Recognition. In *Proceedings of IEEE International Conference on Image Processing*. 36-40.

[20] Min Peng, Chongyang Wang, Tao Bi, Yu Shi, Xiangdong Zhou, and Tong Chen. 2019. A Novel Apex-Time Network for Cross-Dataset Micro-Expression Recognition. In *Proceedings of International Conference on Affective Computing and Intelligent Interaction*. 1-6.

[21] Min Peng, Zhan Wu, Zhihao Zhang, and Tong Chen. From Macro to Micro Expression Recognition: Deep Learning on Small Datasets Using Transfer Learning. In *Proceedings of IEEE International Conference on Automatic Face & Gesture Recognition*. 657-661.

[22] Zhaoqiang Xia, Xiaopeng Hong, Xingyu Gao, Xiaoyi Feng, and Guoying Zhao. 2020. Spatiotemporal Recurrent Convolutional Networks for Recognizing Spontaneous Micro-Expressions. *IEEE Transactions on Multimedia*. 22, 3 (2020), 626-640.

[23] Monu Verma, Santosh Kumar Vipparthi, Girdhari Singh, and Subrahmanyam Murala. 2020. LEARNet: Dynamic Imaging Network for Micro Expression Recognition. *IEEE Transactions on Image Processing*. 29 (2020), 1618-1627.

[24] Lei Zhong, Changmin Bai, Jianfeng Li, Tong Chen, Shigang Li, and Yiguang Liu. 2019. A Graph-Structured Representation with BRNN for Static-based Facial Expression Recognition. In *Proceedings of IEEE International Conference on Automatic Face & Gesture Recognition*. 1-5.

[25] Zhaoqiang Xia, Xiaoyi Feng, Jinye Peng, Xianlin Peng, and Guoying Zhao. 2016. Spontaneous micro-expression spotting via geometric deformation modeling. *Computer Vision and Image Understanding*. 147 (2016), 87–94.

[26] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. 2013. Robust Discriminative Response Map Fitting with Constrained Local Models. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 3444-3451.

[27] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. 2018. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv preprint arXiv:1803.01271*.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 770-778.

[29] Wen-Jing Yan, Xiaobai Li, Su-Jing Wang, Guoying Zhao, Yong-Jin Liu, Yu-Hsin Chen, Xiaolan Fu. 2014. CASME II: An Improved Spontaneous Micro-Expression Database and the Baseline Evaluation. *PLoS ONE*. 9, 1 (2014), e86041.

[30] Sze-Teng Liong, John See, KokSheik Wong, and Raphael Chung-Wei Phan. 2016. Automatic Micro-expression Recognition from Long Video Using a Single Spotted Apex. In *ACCV Workshop*. 345-360.

[31] Tomas Pfister, Xiaobai Li, Guoying Zhao, and Matti Pietikäinen. 2011. Recognising spontaneous facial micro-expressions. In *Proceedings of International Conference on Computer Vision*. 1449-1456.

[32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*. 1097–1105.

[33] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS Workshop*. 1–9.

[34] Su-Jing Wang, Wen-Jing Yan, Xiaobai Li, Guoying Zhao, Chun-Guang Zhou, and Xiaolan Fu. 2015. Micro-Expression Recognition Using Color Spaces. *IEEE Transactions on Image Processing*. 24, 12 (2015), 6034-6047.

[35] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. 2015. Beyond short snippets: Deep networks for video classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 4694-4702.

[36] Madhumita A. Takalkar and Min Xu. 2017. Image Based Facial Micro-Expression Recognition Using Deep Learning on Small Datasets. In *Proceedings of International Conference on Digital Image Computing: Techniques and Applications*. 1-7.