Micro-Expression Classification based on Landmark Relations with Graph Attention Convolutional Network

Ankith Jain Rakesh Kumar and Bir Bhanu
Department of Electrical and Computer Engineering
University of California, Riverside
arake001@ucr.edu, bhanu@ece.ucr.edu

Abstract

Facial micro-expressions are brief, rapid, spontaneous gestures of the facial muscles that express an individual's genuine emotions. Because of their short duration and subtlety, detecting and classifying these micro-expressions by humans and machines is difficult. In this paper, a novel approach is proposed that exploits relationships between landmark points and the optical flow patch for the given landmark points. It consists of a two-stream graph attention convolutional network that extracts the relationships between the landmark points and local texture using an optical flow patch. A graph structure is built to draw-out temporal information using the triplet of frames. One stream is for node feature location, and the other one is for a patch of optical-flow information. These two streams (node location stream and optical flow stream) are fused for classification. The results are shown on, CASME II and SAMM, publicly available datasets, for three classes and five classes of micro-expressions. The proposed approach outperforms the state-of-the-art methods for 3 and 5 categories of expressions.

1. Introduction

Facial expressions play a vital role in social interactions. The facial expressions are categorized into two groups: facial macro-expressions and facial micro-expressions. Facial macro-expressions are prolonged, have large intensity, and are easily recognizable by humans and machines. The research in spotting and classification of facial macro-expressions has been one of the key research areas of computer vision. Compared to the research in the field of macro-expressions recognition, micro-expressions is relatively new. Facial micro-expressions (MEs) are brief, subtle, rapid, and involuntary facial muscle movements beneath the skin, and the time-frame of these expressions is less than a fraction of a second. These facial micro-expressions show the person's genuine emotions [1]. These micro-expressions cannot be faked, concealed, or used to deceive an individual's true feelings or state-of-the-mind. Micro-expressions cannot be identified or recognized easily by a human without any training. Micro-expressions have a wide range of applications in the fields of lie detection, online learning, security, health care (depression recovery, therapies, and more), and online gaming. Therefore, it is necessary to develop a micro-expression recognition system.

In the last decade, micro-expression recognition was based on the traditional hand-crafted approaches such as Local Binary Pattern with Three Orthogonal Planes (LBP-TOP) [2], Bi-Weighted Oriented Optical Flow (Bi-WOOF) [3], and 3D Histogram of Oriented Gradient (3DHOG) [4] to extract the spatio-temporal information. However, these techniques need improvements to recognize subtle changes in the facial muscle movements. With the recent advancement in the deep learning field, researchers have used convolutional neural networks (CNN) to extract features for the classification of micro-expressions. For instance, Liong *et al.* [5] used a divide and conquer approach to identify the apex frame. They used an onset frame and an apex frame to extract the optical flow features and further classified them using CNN. Peng *et al.* [6] used a two-stream 3D-CNN model to accommodate different frame-rates of facial micro-expression videos and extract the spatio-temporal features. Another approach [7] determines the occurrence of facial muscle movements represented by Action Units (AUs) and classifies them using CNN.

The classification of facial micro-expressions is a challenging task due to three important characteristics: (i) subtle behavior (low intensity of facial expressions), (ii) brief and rapid change, and (iii) short time duration (less than a second). Another significant problem with facial micro-expression classification is the lack of adequate and balanced training data. The limited and unbalanced data make the training of an end-to-end neural network model challenging, it results in higher accuracy for the majority class. Thus, resulting in biased prediction results.

To overcome the above significant issues, we propose a approach for *end-to-end training of a novel graph structure that is used to extract the temporal information using the triplet of frames and a two-stream Graph Attention Convolutional Neural Network (GACNN) model using the relationship between the landmark points location information and the optical flow patch information*. To address the unbalanced data samples issue, we use videos from the other datasets of the same class to increase the number of samples of data. In addition to the above data augmentation approach, we use various amplification factors of EMM [8] technique to maximize the number of data samples for the class with lower data samples, thus, balancing the dataset.

The rest of this paper is organized as follows. In section 2, we introduce the related works and our contributions. In section 3, we explain the technical approach for the classification of facial micro-expression videos. In Section 4, we present the qualitative and quantitative experimental results, including ablation study results. Finally, in section 5, we present conclusions and future work.

2. Related Work and Contributions

Micro-expression recognition (MER) has received a lot of interest in the last decade, but limited work has been done until now. The methods used in the classification of micro-expressions are based on various feature extraction, namely: i) handcrafted feature extraction, ii) convolutional neural network, and iii) graph networks.

2.1. MER using Handcrafted Features

Zhao *et al.* [2] used a handcrafted approach such as Local Binary Patterns with Three Orthogonal Planes (LBP-TOP) to extract the facial features robust to illumination changes for classifying micro-expressions. These LBP-TOP features help in discriminating the local texture feature information by translating the vector code into histograms on three planes (XY, XT, YT). Finally, the histograms of these planes are concatenated into a single histogram feature. Davison *et al.* [3] proposed a temporal feature extractor known as 3D Histogram of Oriented Gradient (3DHOG). The 3DHOG approach extracted texture features from all three directions of motion for the classification. Liong *et al.* [4] proposed to use only the apex frame (high-intensity expression frame) of the video. The feature extractor Bi-Weighted Oriented Optical Flow (Bi-WOOF) is used to enhance the apex frame feature for classifying the micro-expressions. Liong *et al.* [9] used optical flow and optical strain magnitudes to classify the micro-expressions on the two datasets CASME II and SMIC. Liu *et al.* [10] used optical flow features and processed the textual features using affine transformation to remove any sensitivity of head movements and lighting conditions. Furthermore, the facial areas are divided into regions-of-interest (ROIs). They used support vector machine (SVM) to classify expressions.

2.2. MER using Convolutional Neural Networks

Khor *et al.* [11] proposed a CNN-LSTM method called ELRCN, which used both optical flow and optical strain features as inputs to the CNN-LSTM network that extracted spatio-temporal features. Further, Support Vector Machines (SVM) is used to classify the videos. Peng *et al.* [6] proposed a two-stream 3D CNN model called Dual Temporal Scale Convolutional Neural Network (DTSCNN). Different frame rates of facial micro-expression videos are accommodated by the two-stream 3D CNN models. Liong *et al.* [5] used a divide-and-conquer approach to determine the apex frame. They used the onset frame and apex frame to extract the optical flow features and further classified them using CNN. Kumar *et al.* [12] eliminated the low-intensity expression frames of the video in the frequency domain. The rest of the remaining high-intensity expression frames are converted into a single-motion magnified avatar image. Then the avatar image [13] is used as an input to the CNN model to classify the expressions. Khor *et al.* [14] presented a robust approach that learned micro-expression features by exploiting two-stream CNNs with heterogeneous motion-based inputs called Dual-Stream Shallow Network (DSSN). Xia *et al.* [15] proposed a framework that leverages macro-expression datasets as a guidance system to assist the micro-expression network. They used two disentangle networks, MicroNet, and MacroNet to extract the features. The MacroNet is fixed and used to guide the fine-tuning of MicroNet from both facial features and label space.

2.3. MER using Graph Networks

Lo *et al.* [16] proposed an AU-oriented graph convolutional neural network, namely MER-GCN. They used 3D CNN to extract the AU features and then appllied the GCN network to determine the dependency among AU nodes for ME recognition. This was the first work using AU-based GCN to classify facial micro-expressions. Lei et al. [17] used transfer learning to magnify the MEs using the learning-based video motion magnification and extracted shape information. A novel graph temporal convolutional network is proposed to extract the local muscle movement features. They used two channels, one for node features and the other for the edge feature extraction. Xie et al. [18] proposed a recognition approach by combining emotion category labels and AUs. They modeled AUs based on relational information and integrated it with the AUs recognition task. They used generative adversarial networks (GANs) for data augmentation in imbalanced datasets.

2.4. Contributions

The contributions of this paper are given below:

- We propose an end-to-end landmark-assisted two-stream Graph Attention Convolutional Network, which integrates landmark points location with optical flow information to classify facial micro-expressions.
- We design a graph to extract the temporal information using the triplet of frames structure. We use a two-streams graph attention network, one for node locations and the other for optical flow patch information, and later fuse them. We describe an approach to automatically select the high intensity expression frames from the video based on the optical flow magnitude.
- We provide a comprehensive evaluation of the proposed approach on two publicly available datasets for 3 and 5 classes of facial micro-expressions.

3. Proposed Approach

The overall framework of our method for the classification of facial micro-expression is shown in Fig. 1. First, we use Eulerian Motion Magnification (EMM) to amplify the signals and extract the magnified input video. Second, we define an approach to automatically select the high-intensity expression frames and remove the rest of low-intensity frames using a optical flow magnitude threshold value. We use dlib software [19] to detect the landmark points on the face. Based on these detected landmark points, we construct our graph and calculate the optical flow patch feature information at the selected landmark points. Finally, we classify the micro-expressions using a two-stream Graph Attention Convolutional Neural Network using the landmark points location information and the optical flow patch information.

3.1. Eulerian Motion Magnification

Eulerian Motion Magnification (EMM) [8] amplifies the small motions in videos by integrating spatial and temporal processing to pay attention to the subtle facial features in a video. The advantages of using EMM are as follows: firstly, magnifying the videos helps in exaggerating the small signals and makes it easier for the human eye to recognize these micro-expressions. Secondly, to balance the database, we can use different amplification factors α and augment the datasets using these samples in the training sets.

The selection of the right amplification factor to magnify the videos is crucial. Higher the value of α, more artifacts are added in the video due to the noise amplification. Therefore, we select lower values of α to magnify our videos. We chose a preset value of α to be from 2 to 5. We ran an experiment to determine the best α value using the PSNR ratio and visually noticed the deformation of videos at α values five and above. Lei et al. [17] also show that the higher the α value, the video quality gets worse. Therefore, we finally chose α to be 4 for our testing samples during the experiments. For training process and to balance the datasets, we use α equal to 1, 2, 3, 4 and 5. The EMM with different α values ranging from 1 to 4 is shown in Fig. 2.

3.2. Frame Selection Approach

Since micro-expressions are subtle and last less than a fraction of a second, we are only concerned with frames that have high-intensity expression information for the classification task. Therefore, it is crucial to remove the low-intensity expression frames from the video.

We calculate the optical flow [20] for the video frames, and obtain its magnitude for each frame. A graph using the flow magnitude (vs) the frame number is shown in Fig. 3. Next, we calculate the threshold value by taking the average flow magnitude values of the first five video frames. Finally, we remove the low-intensity frames using the threshold value equal to 1.25 times the average flow magnitude of the first five video frames. Any frames having above the threshold values are selected and, the rest of the frames are discarded. We require a minimum of 3 frames (the graphs are connected from current frame to the previous frame and the future frame) for our graph construction to extract the temporal information. In any scenario, if any of the videos do not have the required 3 video frames then, we select all the video frames for the classification task. The reason for the selection of higher and strict threshold values is to reduce the number of frames for the training and testing process because the graph networks are slow to converge.

3.3. Landmark Points Detection and Node Feature Extraction

We use dlib software [19] to obtain 68 landmark points. But, all these landmark points are not predominant for the classification of facial micro-expressions. As a result, we eliminate the landmark points along the face's contour area, as well as a few points on the nose and the inner points of the mouth. Finally, 37 landmark points are remaining out of 68. We add a few extra reference points on the forehead region (10) (above the eyebrow region) and near the mouth region (4) of the face as shown in Fig. 4. These landmark points are obtained and added using the onset frame of the video. These 14 landmark points are used as reference for all video frames. The significance of these points is that we capture the subtle changes on the forehead region and near the cheek regions. The other important information that we obtain is how other landmark points move with respect to these 14 reference landmark points. Now, we have a total of 51 landmark points on the face as shown in Fig. 4

After extracting the 51 landmark points on the face re-

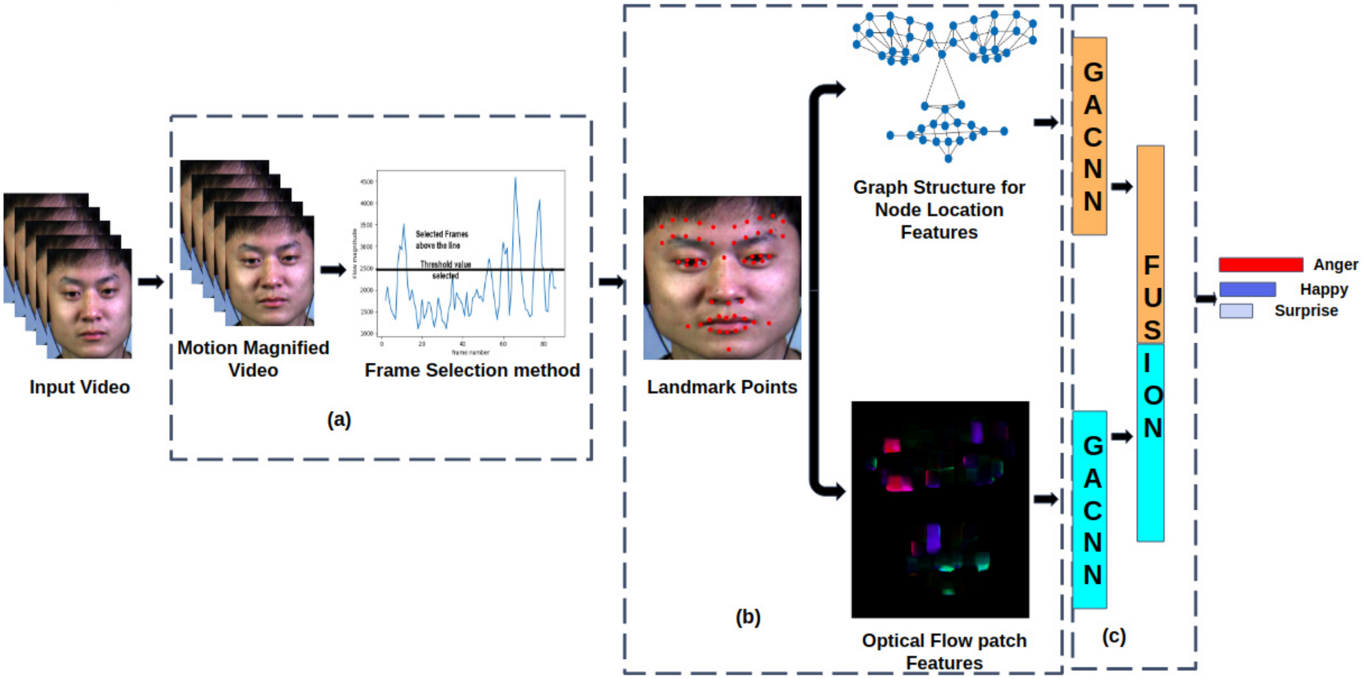

Figure 1: The overall architecture of our approach. (a) pre-processing step magnifies the input video. Further, we remove the low-intensity expression frames. (b) Creates a graph structure using landmark points and optical flow features, and (c) graph attention convolutional network for training the graph representation and, classification.

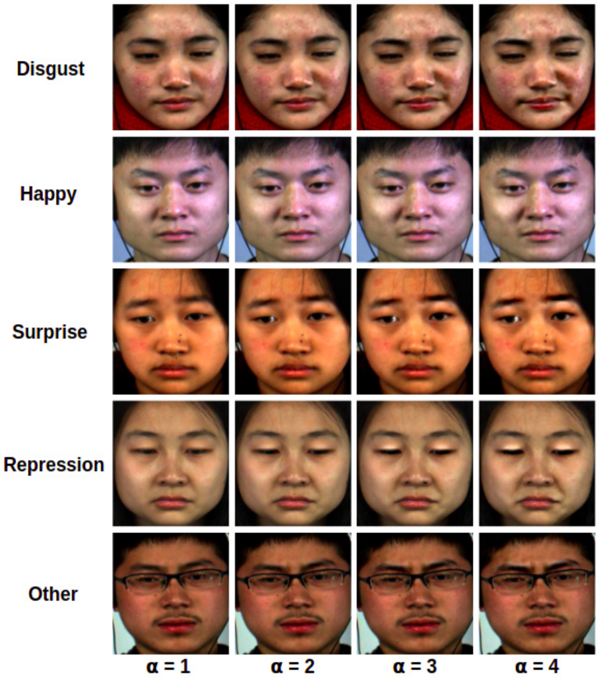

Figure 2: EMM for different α values ranging from 1 to 4

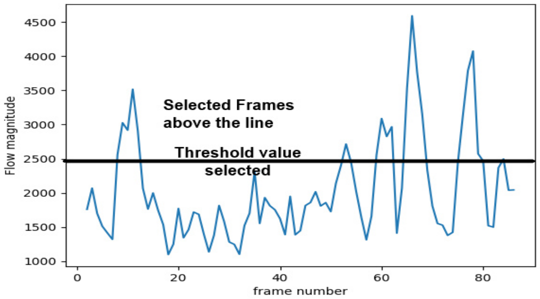

Figure 3: Elimination of the low intensity expression frames from the video using a threshold value for the optical flow magnitude.

gion, we connect the landmark points based on the human facial structure. Later we connect the 10 reference landmark points located on the forehead with the eye and eyebrow regions. The 4 landmark points near the mouth regions are connected. We use node locations as the feature vector for the first stream of graph network.

For the image features, we calculate the optical flow for a patch size of *10*×*10* domain at the respective landmark location as shown in Fig. 5. The reason for selecting a *10*×*10* patch size is that we do not want to miss any changes in the facial muscle movement near the landmark points. The other reason is that we have motion magnified the video

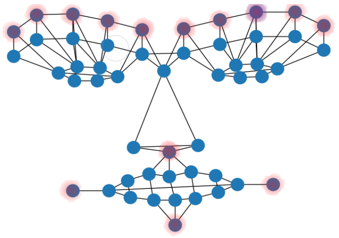

Figure 4: Graph structure of an input frame with 51 landmark points. The red color points (14) are reference points added to capture extra information on the forehead and near the cheek and mouth region.

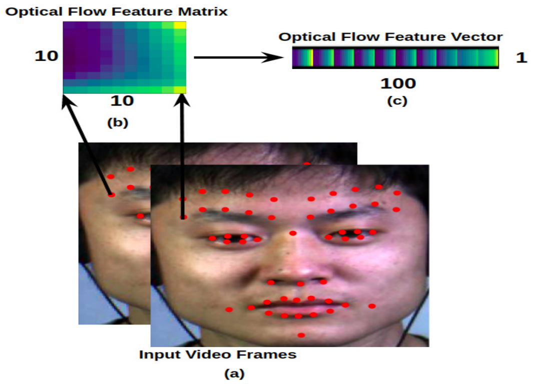

Figure 5: The process of obtaining the optical flow patch information. (a) are the input frames of video, (b) 10×10 optical flow feature matrix is the patch around each landmark point, and (c) optical flow feature vector as an input for the graph node features.

samples. Therefore, the new magnified videos are not subtle anymore. The *10×10* optical flow feature matrix is flattened to a *1D* of size *100×1* optical flow feature vector. The *reason to flatten the matrix to vector is to extract the edge features and reduce the amount of computation*. The optical flow feature vector is an input to the second stream of the graph network for 51 features.

3.4. Facial Graph Structure

The basic building block for the graph structure is the node data and edge data which can be shown as follows:

$$G = (N, E) \tag{1}$$

where N = Nodes and E = Edges. N is the number of landmark points selected based on the facial regions. In our case, N = n_1, n_2, n_3, …., n_{51}. The E is the number of edges obtained by the connection between two nodes, where E = (e_{12}, e_{13}, e_{24}, .., e_{ij}), and i and j are the respective nodes. The graph structure is shown in Fig. 4. The nodes and edges will move based on the facial muscle movements. Each micro-expression class will have different patterns of muscle movement on the face. Therefore, the muscle movements of nodes and edges will vary, and thus, graph structure can be used for the classification of facial micro-expressions.

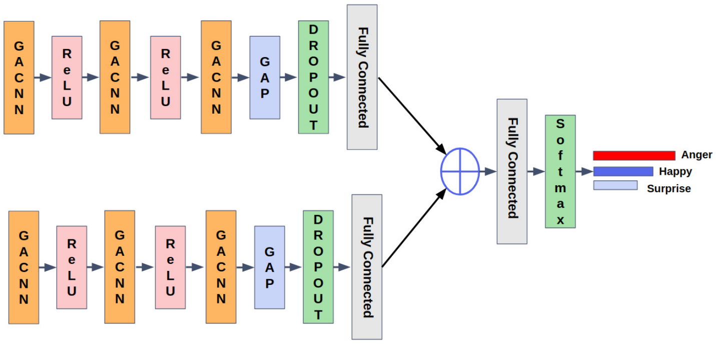

Figure 6: Architecture of Two-Stream Graph Attention Convolutional Neural network. GAP stands for global average pooling.

We designed a graph to extract the temporal information using the triplet of frames structure (three video frames). Here, the current frame is connected to both the previous frame and the future frame. The entire video is converted into a single graph with connections to the frames using triplet of frames.

3.5. Two-Stream Graph Attention Convolutional Network

To extract the temporal features from the video, we designed a novel Two-stream Graph Attention Network for training the graph structure shown in Fig. 6. We design a triplet of frames structure. The entire video is converted into a single graph using triplet of frames. We extract the node features and optical flow features for the classification.

In the attention task for the graph structure, Velickovic *et al.* [21] proposed Graph Attention Network (GAT) that employs self-attention of node features. The approach assumes that the contributions of neighboring nodes to the central nodes are neither identical nor predetermined like the Graph Convolutional Network (GCN) model. GAT adopts to learn the weights between two connected nodes using an attention mechanism.

We use GAT [21] and (GCN) [22] to design our graph network as shown in Fig. 6. We use three GACNN layers with the ReLu activation function after each graph layer. We use 64 hidden channels and, the concatenation operation is off, and the number of heads = 1 for the GAT layer. We use the dropout function after the global average pooling operation. For the first stream, the node feature vector size is equal to x and y coordinates and, for the second stream, the length of the node vector is of size 100.

At the end of the fully connected layer of the two-stream networks, the results are concatenated for the graph representation of the two streams. Finally, the output is passed through the final fully connected layer and softmax layer for classification. We use Adam optimizer with the learning rate equal to 0.001. The learning rate decreases by half every 100 epochs.

4. Experimental Results

In this section of the paper, the experimental details will be described, including the datasets used, experimental setup, results, and ablation study.

4.1. Experimental Setup

We conduct experiments on two publicly available datasets CASME II [23] and SAMM [24] datasets for the evaluation on 3 and 5 classes of expressions. We evaluate our results using l*eave-one-subject-out* cross validation approach. The experiments were conducted on a workstation running Ubuntu 16.04 with 64GB RAM and two NVIDIA GeForce GTX 1080Ti GPUs. We use PyTorch for network implementation.

4.2. Datasets and Preprocessing

The *two* publicly available datasets are: CASME II [23] and SAMM [24]. Both the datasets use high-speed cameras with 200fps and, the apex frame of each video is marked. CASME II dataset has 255 video samples from 26 subjects of 7 classes whereas, the SAMM database has 159 micro-expression videos from 29 participants of 8 categories. We are interested in classifying the expressions into 3 and 5 categories. The selection of CASME II and SAMM data for 5 classes is based on the graph-based approach papers [17] and [18] to compare our approach with their methods. *We use Leave-one-subject-out cross validation approach for evaluation of our approach.* Table. 1 and 2 shows the dataset distributions for each expression class for CASME II and SAMM 3 and 5 class categories.

For better extraction of facial features, we aligned and resized the image frames to 256x256. To solve the issue of data imbalance, we used the data (Happy and Surprise) from the other dataset while training for the class having lower number of samples of videos to improve the training accuracy. Also, we used different magnitudes of motion magnification (1, 2, 3, 4, and 5) to increase the data samples to overcome the class imbalance of the datasets during training. The amplification factor of 4 is used during the evaluation process and the other magnified samples are used to increase the number of samples for the classes having the least number of samples (happy, surprise, repression, and contempt)

Table 1: Summary of the data distributions for CASME II and SAMM for 3 classes

Expression Class	CASME II	SAMM
Negative	88	92
Positive	32	26
Surprise	25	15

Table 2: Summary of the data distributions for CASME II and SAMM for 5 classes

Expressions	CASME II	Expressions	SAMM
Disgust	63	Anger	57
Happy	32	Happy	26
Surprise	25	Surprise	15
Repression	27	Contempt	12
Other	99	Other	26

4.3. Evaluation Metrics

We use the Unweighted F1 score to evaluate the recognition performance and also use accuracy as a metric.

4.3.1 Unweighted F1 score (UF1)

F1 score provides equal emphasis on each class of the datasets. From the confusion matrix, we compute the True Positives (TP), False Positives (FP), and False Negatives (FP) for each class c. The final balanced F1 score is computed by taking the average for each class F1 scores shown in equation (3).

$$F1_c = \frac{2 \times TP_c}{2 \times TP_c + FP_c + FN_c} \tag{2}$$

$$UF1 = \frac{F1_c}{C}, \tag{3}$$

where, F1$_c$ is F1-score for each individual class, C is the number of classes.

4.3.2 Accuracy

The accuracy is calculated using the equation (4).

$$Acc = \frac{P}{N} \times 100\% \tag{4}$$

where P, is the total number of correct predictions and N is the number of video samples.

4.4. Experimental Results

Table 3 shows the comparison of results between the state-of-the-art methods and our approach for CASME II and SAMM datasets for three categories of expressions: Negative, Happy, and Surprise using the Leave-One-Subject-Out Cross-Validation (LOSO-CV). LOSO-CV is a

Table 3: Comparison with the state-of-the-art approaches for CASME II and SAMM datasets for 3 categories of expressions

Method	Feature Extraction Approach	CASME II		SAMM	
		Accuracy	F1 Score	Accuracy	F1 Score
Ngo *et al.* [25]	Handcrafted	0.4900	0.5100	0.5900	0.364
Wang *et al.* [26]	Handcrafted	0.4650	0.4480	0.4150	0.4060
Liong *et al.* [4]	Handcrafted	0.5880	0.6100	0.5830	0.3970
Huang *et al.* [27]	CNN	0.6400	0.6380	0.6380	0.6110
Khor *et al.* [14]	CNN	0.7080	0.7300	0.5740	0.4640
Gan *et al.* [28]	CNN	0.8828	**0.8697**	0.6818	0.5423
Kumar *et al.* [12]	CNN	0.8621	0.8280	0.8195	0.7056
Lo *et al.* [16]	Graph based	0.5440	0.3030	0.5340	0.2830
Xie *et al.* [18]	Graph based	0.7120	0.3550	0.5230	0.3570
Ours	Graph based	**0.8966**	0.8695	**0.8872**	**0.8118**

Table 4: Comparison with the state-of-the-art approaches for CASME II datasets for 5 categories of expressions

Methods	Descriptors	Accuracy	F1-Score
Khor et al. [29]	LBP-TOP	0.3968	0.3589
Khor et al. [29]	Alexnet	0.6296	0.6675
Kim et al. [30]	CNN-LSTM	0.6098	N/A
Liong et al. [31]	Bi-WOOF	0.6255	0.6500
Zong et al. [32]	Hier. STLBP-IP	0.6397	0.6125
Liu et al. [33]	Sparse MDMO	0.6695	0.6911
Li et al. [34]	HIGO-Mag	0.6721	N/A
Huang et al. [35]	DiSTLBP-RIP	0.6478	N/A
Peng et al. [36]	ME-Booster	0.7085	N/A
Khor et al. [29]	DSSN	0.7078	**0.7297**
Khor et al. [29]	SSSN	0.7119	0.7151
Lei et al. [17]	Graph TCN	0.7398	0.7246
Ours	GACNN	**0.8130**	0.7090

Table 5: Comparison with the state-of-the-art approaches for SAMM datasets for 5 categories of expressions

Method	Descriptors	Accuracy	F1-Score
Khor et al.[29]	LBP-TOP	0.3968	0.3589
Khor et al. [29]	CNN	0.5294	0.4260
Khor et al. [29]	SSSN	0.5662	0.4513
Khor et al. [29]	DSSN	0.5735	0.4644
Lei. et al. [17]	Graph TCN	0.7500	0.6985
Ours	GACNN	**0.8824**	**0.8279**

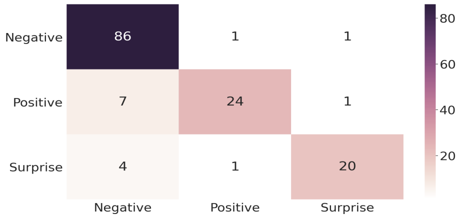

Figure 7: Confusion matrix for CASME2 datasets (3 classes)

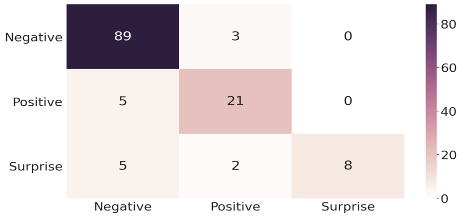

Figure 8: Confusion matrix for SAMM datasets (3 classes)

K-fold cross-validation technique, with K equal to N subjects, which means we repeat the experiment N times with N-1 subjects for the training process and the remaining 1 subject for the testing. As a measure of the robustness of our approach, we quantify our results using balanced metrics: Unweighted F1 score and Accuracy. Our two-stream graph attention convolutional neural network outperforms all the state-of-the-art methods as shown in Table 3. For CASME II datasets, our approach gets *1.38*% higher accuracy results and F1-Score is lower by *0.02*% as compared to Gan et al [28]. The confusion matrix for the CASME II dataset (3 classes) is shown in Fig. 7. Similarly, for the SAMM dataset our approach improves accuracy by *6.77*% and F1-Score is higher by *10.62*% as compared to other methods. Our approach gets *18.46*% higher accuracy and *52.06*% better F1-Score when compared to the current graph-based approaches for the CASME II dataset. Similarly, for the SAMM dataset, our method gets *36.42*% higher accuracy and *45.48*% better F1-Score when compared to current graph-based approaches for the classification of 3 classes. The confusion matrix for the SAMM dataset (3 classes) is shown in Fig. 8.

The comparison results (5 classes) for the CASME II dataset using the state-of-the-art approaches and our ap-

Table 6: Ablation study for CASME II and SAMM dataset for 3 classes of expressions.

Method	CASME II (3 classes)		SAMM (3 classes)	
	Accuracy.	F1-Score	Accuracy.	F1-Score
GCN (without Frame Selection. and Attention.)	0.7586	0.6648	0.8271	0.6746
GCN (with Frame Selection.)	0.8000	0.7235	0.8496	0.7381
GCN (with Frame Selection. and Attention.)	**0.8966**	**0.8695**	**0.8872**	**0.8118**

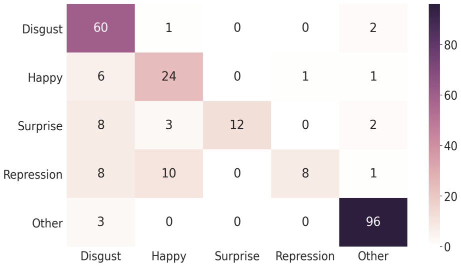

Figure 9: Confusion matrix for CASME2 datasets (5 classes)

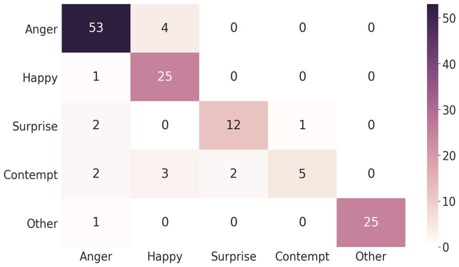

Figure 10: Confusion matrix for SAMM datasets (5 classes)

proach is shown in Table 4. The recent paper from Khor et al. [29] proposed Dual Stream Shallow Network (DSSN) and Single Stream Shallow Network (SSSN) as their proposed methods. These methods had the best result for CASME II datasets for three classes until recent times using CNN architectures. When compared to SSSN, our proposed approach improves the accuracy by *10.11*%, but the F1 score is less than their method by *2.07*%. Similarly, our approach increases precision by *7.32*%, but its F1-score is lower by *1.56*% as compared to the graph-based approach (GRAPH TCN) [17]. The confusion matrix for the CASME II dataset (5 classes) is shown in Fig. 9.

Table 5 shows the comparative results (5 classes) using the state-of-the-art approaches and our approach for the SAMM dataset. There are very few researchers who have worked on SAMM datasets for the classification of micro-expressions into five categories. Compared to the best state-of-the-art method using the CNN for the classification of micro-expressions into five categories, our proposed method improves the accuracy by *30.89*% and F1-Score by *36.35*%. Compared to the Graph-based approach (Graph TCN), our method achieves *13.24*% higher accuracy and *12.94*% better F1-Score. The confusion matrix for the SAMM dataset (5 classes) is shown in Fig. 10.

Table 6 shows the ablation study results (3 classes) for the CASME II and SAMM datasets, respectively. We observe that there is an improvement in accuracy by *4.14*% and *5.87*% in F1-score for CASME II dataset and *2.25*% in accuracy and *6.35*% in F1-score for the SAMM dataset, respectively, when using graph convolutional network (GCN) [22] with our proposed frame selection process without attention network. When we use frame selection process along with the attention network, our results improves in accuracy by *9.66*% and *15.21*% in F1-score for CASME II dataset and *3.76*% and *7.37*% in F1-score for the SAMM datasets, respectively.

5. Conclusions and Future Work

In this paper, we proposed a Two-stream Graph Attention Convolutional Neural Network for the node location features and the optical flow feature vector with the help of a triplet of frames to extract the temporal information. We define a frame selection process to discard the low-intensity expression frames. The results from the two-stream network are fused for the classification of micro-expression (MEs). We conduct a comprehensive evaluation of the CASME II and SAMM datasets for 3 and 5 categories of expressions. Our proposed approach outperforms the state-of-the-art methods by *1.38*% and *7.32*% accuracy for the CASME II dataset for three and five categories, respectively. For the SAMM dataset, our method improves the accuracy results from the current approaches by *6.77*% and *13.24*% for 3 and 5 categories of expressions, respectively. Our frame-selection method improves the overall performance for the classification of MEs. We observe that the frame selection process helps in improving the accuracy by *4.14*% for CASME II and *2.25*% for SAMM datasets for 3 classes, respectively without using the attention network. In the future, we will work on automatically learning the edge connections and the relationship between the edges and node features.

6. Acknowledgements

This material is based upon work supported by the National Science Foundation under grant number 1911197.

References

[1] "What are facial micro-expressions?," https://www.paulekman.com/resources/micro-expressions/.

[2] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 915–928, June 2007.

[3] A. K. Davison, W. Merghani, and M. H. Yap, "Objective classes for micro-facial expression recognition," *Journal of Imaging*, vol. 4, no. 10, 2018.

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

[5] S. T. Liong, Y. S. Gan, W. C. Yau, Y. C. Huang, and T. L. Ken, "Off-apexnet on micro-expression recognition system," *CoRR*, vol. abs/1805.08699, 2018.

[6] M. Peng, C. Wang, T. Chen, G. Liu, and X. Fu, "Dual temporal scale convolutional neural network for micro-expression recognition," *Frontiers in Psychology*, vol. 8, p. 1745, 2017.

[7] Y. Li, X. Huang, and G. Zhao, "Micro-expression action unit detection with spatial and channel attention," *Neurocomputing*, vol. 436, pp. 221–231, 2021.

[8] H. Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. T. Freeman, "Eulerian video magnification for revealing subtle changes in the world," *ACM Transactions on Graphics (Proc. SIGGRAPH 2012)*, vol. 31, no. 4, 2012.

[9] S. Liong, J. See, R. C. Phan, Y. Oh, A. C. L. Ngo, K. Wong, and S. Tan, "Spontaneous subtle expression detection and recognition based on facial strain," *CoRR*, vol. abs/1606.02792, 2016.

[10] Y. Liu, J. Zhang, W. Yan, S. Wang, G. Zhao, and X. Fu, "A main directional mean optical flow feature for spontaneous micro-expression recognition," *IEEE Transactions on Affective Computing*, vol. 7, no. 4, pp. 299–310, 2016.

[11] H. Khor, J. See, R. C. W. Phan, and 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)*, pp. 667–674, May 2018.

[12] A. J. R. Kumar, R. Theagarajan, O. Peraza, and B. Bhanu, "Classification of facial micro-expressions using motion magnified emotion avatar images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[13] S. Yang and B. Bhanu, "Understanding discrete facial expressions in video using an emotion avatar image," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 980–992, 2012.

[14] H. Khor, J. See, S. Liong, R. C. W. Phan, and W. Lin, "Dual-stream shallow networks for facial micro-expression recognition," in *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 36–40, 2019.

[15] B. Xia, W. Wang, S. Wang, and E. Chen, "Learning from macro-expression: A micro-expression recognition framework," in *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, (New York, NY, USA), p. 2936–2944, Association for Computing Machinery, 2020.

[16] L. Lo, H. Xie, H. Shuai, and W. Cheng, "Mer-gcn: Micro-expression recognition based on relation modeling with graph convolutional networks," in *2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, (Los Alamitos, CA, USA), pp. 79–84, IEEE Computer Society, aug 2020.

[17] L. Lei, J. Li, T. Chen, and S. Li, "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, (New York, NY, USA), p. 2237–2245, Association for Computing Machinery, 2020.

[18] H.-X. Xie, L. Lo, H.-H. Shuai, and W.-H. Cheng, "Au-assisted graph attention convolutional network for micro-expression recognition," in *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, (New York, NY, USA), p. 2871–2880, Association for Computing Machinery, 2020.

[19] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, no. 60, pp. 1755–1758, 2009.

[20] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Image Analysis* (J. Bigun and T. Gustavsson, eds.), (Berlin, Heidelberg), pp. 363–370, Springer Berlin Heidelberg, 2003.

[21] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2018.

[22] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[23] W. J. Yan, X. Li, S. J. Wang, G. Zhao, Y. J. Liu, Y. H. Chen, and X. Fu, "Casme ii: An improved spontaneous micro-expression database and the baseline evaluation," *PLOS ONE*, vol. 9, pp. 1–8, 01 2014.

[24] A. K. Davison, C. Lansley, N. Costen, K. Tan, and M. H. Yap, "Samm: A spontaneous micro-facial movement dataset," *IEEE Transactions on Affective Computing*, vol. 9, pp. 116–129, Jan 2018.

[25] A. C. Le Ngo, J. See, and R. C. . Phan, "Sparsity in dynamics of spontaneous subtle emotions: Analysis and application," *IEEE Transactions on Affective Computing*, vol. 8, no. 3, pp. 396–411, 2017.

[26] Y. Wang, J. See, R. C.-W. Phan, and Y.-H. Oh, "Lbp with six intersection points: Reducing redundant information in lbp-top for micro-expression recognition," in *Computer Vision – ACCV 2014* (D. Cremers, I. Reid, H. Saito, and M.-H. Yang, eds.), (Cham), pp. 525–537, Springer International Publishing, 2015.

[27] X. Huang, G. Zhao, X. Hong, W. Zheng, and M. Pietikäinen, "Spontaneous facial micro-expression analysis using spatiotemporal completed local quantized patterns," *Neurocomputing*, vol. 175, pp. 564–578, 2016.

[28] Y. Gan, S. T. Liong, W. C. Yau, Y. C. Huang, and L. K. Tan, "OFF-ApexNet on micro-expression recognition system," *Signal Processing: Image Communication*, vol. 74, pp. 129 – 139, 2019.

[29] H. Khor, J. See, S. Liong, R. C. W. Phan, and W. Lin, "Dual-stream shallow networks for facial micro-expression recognition," in *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 36–40, 2019.

[30] D. H. Kim, W. J. Baddar, and Y. M. Ro, "Micro-expression recognition with expression-state constrained spatio-temporal feature representations," in *Proceedings of the 24th ACM International Conference on Multimedia*, MM '16, (New York, NY, USA), p. 382–386, Association for Computing Machinery, 2016.

[31] S. Liong and K. Wong, "Micro-expression recognition using apex frame with phase information," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 534–537, 2017.

[32] Y. Zong, X. Huang, W. Zheng, Z. Cui, and G. Zhao, "Learning from hierarchical spatiotemporal descriptors for micro-expression recognition," *IEEE Transactions on Multimedia*, vol. 20, no. 11, pp. 3160–3172, 2018.

[33] Y. J. Liu, B. J. Li, and Y. K. Lai, "Sparse mdmo: Learning a discriminative feature for micro-expression recognition," *IEEE Transactions on Affective Computing*, vol. 12, no. 1, pp. 254–261, 2021.

[34] X. Li, X. Hong, A. Moilanen, X. Huang, T. Pfister, G. Zhao, and M. Pietikäinen, "Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods," *IEEE Transactions on Affective Computing*, vol. 9, no. 4, pp. 563–577, 2018.

[35] X. Huang, S. Wang, X. Liu, G. Zhao, X. Feng, and M. Pietikäinen, "Discriminative spatiotemporal local binary pattern with revisited integral projection for spontaneous facial micro-expression recognition," *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 32–47, 2019.

[36] W. Peng, X. Hong, Y. Xu, and 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)*, pp. 1–5, 2019.