Dynamic Emotion Modeling with Learnable Graphs and Graph Inception Network

Amir Shirian, *Member, IEEE,* Subarna Tripathi, and Tanaya Guha, *Member, IEEE*

***Abstract*—Human emotion is expressed, perceived and captured using a variety of dynamic data modalities, such as speech (verbal), videos (facial expressions) and motion sensors (body gestures). We propose a generalized approach to emotion recognition that can adapt across modalities by modeling dynamic data as structured graphs. The motivation behind the graph approach is to build compact models without compromising on performance. To alleviate the problem of optimal graph construction, we cast this as a joint graph learning and classification task. To this end, we present the Learnable Graph Inception Network (L-GrIN) that jointly learns to recognize emotion and to identify the underlying graph structure in the dynamic data. Our architecture comprises multiple novel components: a new graph convolution operation, a graph inception layer, learnable adjacency, and a learnable pooling function that yields a graph-level embedding. We evaluate the proposed architecture on five benchmark emotion recognition databases spanning three different modalities (video, audio, motion capture), where each database captures one of the following emotional cues: facial expressions, speech and body gestures. We achieve state-of-the-art performance on all five databases outperforming several competitive baselines and relevant existing methods. Our graph architecture shows superior performance with significantly fewer parameters (compared to convolutional or recurrent neural networks) promising its applicability to resource-constrained devices. Our code is available at `/github.com/AmirSh15/graph_emotion_recognition`.**

***Index Terms*—Graph learning, graph neural network, inception network, emotion recognition.**

I. Introduction

Human emotion is expressed, perceived and captured using a variety of dynamic data modalities, such as speech (verbal), videos (facial expressions) and motion capture (body gestures). Modeling and analysis of these cues are critical for many human-centric systems with applications ranging from driver's safety to mental healthcare to human-robot conversational systems. In recent years, significant progress has been made towards the recognition and analysis of emotion using dynamic facial expressions [1], [2], speech [3], [4] and body gestures [5]. Since human emotion is inherently multimodal, research efforts that combine information from multiple modalities are also on the rise [6]. Besides expressed emotion, work has also been done to analyze emotion evoked by natural images [7], videos [8] and music [9].

A. Shirian and T. Guha are with the Department of Computer Science, University of Warwick, Coventry, UK.
S. Tripathi is with Intel Labs, San Diego, US.

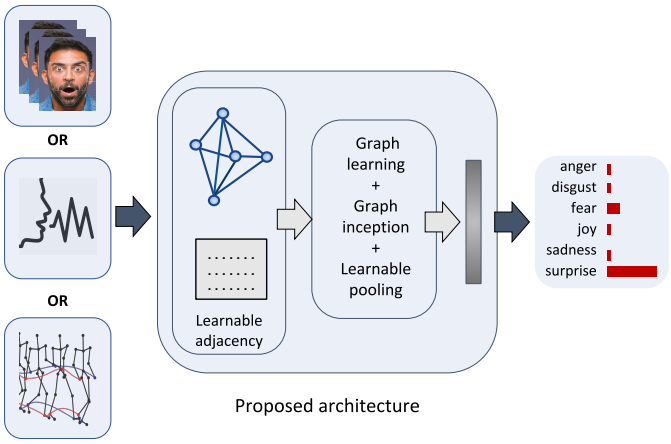


Fig. 1. A generalized graph approach to modeling emotion dynamics. Data samples are transformed to a *learnable* graph structure, where each node corresponds to a short temporal segment or frame. A novel graph architecture (L-GRIN) produces an embedding for the entire graph facilitating emotion recognition.

In the literature of dynamic emotion recognition, recurrent models, such as Long Short Term Memory networks (LSTM) are common [4], [10]. These networks often have complex architecture with millions of trainable parameters requiring large amounts of training data. This makes many emotion recognition models incompatible for use in resource-constrained devices. A compact, efficient and scalable way to represent data is in the form of graphs. We thus adopt a graph approach to building a compact model for dynamic emotion recognition. Furthermore, existing emotion recognition models assume a prior knowledge of the input modality. Since emotion can be sensed through a variety of modalities, a generalized model that can handle disparate modalities efficiently is important. We show that our *modality-agnostic* graph approach is able to achieve state-of-the-art accuracy across various modalities with significantly fewer trainable parameters.

Traditionally, sequences are modeled using Recurrent Neural Networks (RNNs). However, recent literature has successfully used the idea of defining a sequence over a graph [11], [12], [13]. Considering a video frame sequence as a 'structured' graph, Mao et al. showed that graph models can outperform RNNs [11]. Motivated by these recent successes and in the pursuit of a compact model, we propose to adopt a graph approach to model emotion dynamics. Subsequently, we cast emotion recognition as a joint graph learning and classification problem (see Fig. 1 for an overview). In our

approach, each dynamic data sample is represented as a graph, where each node corresponds to a short temporal segment in the data. Each node is associated with the features extracted from the short temporal segment (frame) as its node attributes. This frame-to-node graph construction approach focuses on modeling the temporal dynamics in data; note that spatio-temporal structure (e.g., facial keypoints structure) within the graph resists the idea of a generic, modality-agnostic model and also increases model size significantly. Our graph structure (and hence the model) does not change with the choice of modality or node attributes. Modeling as a graph offers compactness and convenience to handle missing data (particularly common in mocap).

The graph structure i.e., the edge weights connecting the nodes is not naturally defined here. When a graph structure is not known apriori, a common practice is to manually construct the graph. This, however, results into sub-optimal graphs. We thus propose to learn the graph structure itself during the training stage. This is a generalized formulation, where the temporal dependencies between the nodes are automatically discovered. The only assumption we make is that the graph structure remains the same for all samples in a given database. To this end, we propose a novel Graph Convolution Network (GCN) architecture, the *Learnable Graph Inception Network* (L-GrIN), with several novel components: a new definition of graph convolution that uses a non-linear layer-wise projection technique, introduction of an inception module in graph domain, learnable graph structure and a learnable graph-to-vector pooling function. Our architecture produces superior results on five benchmark emotion recognition databases spanning three different modalities (video, audio, mocap). Each database captures one of the following emotional cues: facial expressions, speech and body gestures. In summary, the main contributions of this paper are as follows:

- A generalized, modality-agnostic graph approach to classify dynamic signals that combines graph learning with graph classification.
- A novel graph architecture, termed **L-GrIN**, with a new graph convolution layer, a graph inception module, learnable graph structure and learnable graph-to-vector pooling.
- State-of-the-art performance on dynamic emotion recognition tasks spanning three sensory modalities (video, audio, motion sensors) on five benchmark databases.

II. Related Work

In this section, we review the related work in the areas of GCNs and emotion recognition using various modalities.

A. Graph neural network.

Deep learning on graph data has emerged as a major topic in the past few years. This is because graphs provide a natural and convenient way to deal with large data. Among the different graph neural networks, GCNs have received the most attention [14], [15], [12]. GCNs have been successfully applied to various image and video-based tasks, such as face clustering [16], object detection [17], and video representation learning [11]. GCNs have been used to address skeleton-based action recognition recorded using motion capture [13]. The application of graph networks has also started emerging in automatic speech recognition [18].

GCNs can be broadly classified into two categories: *spatial* and *spectral*. Spatial GCNs imitate the convolution operation of the Convolutional Neural Networks (CNN) by aggregating the information from neighboring nodes [14], [19]. The problem of different graph nodes having different number of neighbours is usually addressed by using a fixed size neighborhood [19] or by converting graph structures to a regular grid and subsequently applying traditional CNNs [20]. A recent work proposed to develop the graph structure considering the Weisfeiler-Lehman graph isomorphism test [21], and achieved state-of-the-art performance in node classification task in citation networks. On the other hand, spectral GCNs formulate the convolution operation as a frequency domain filtering operation following the theory of signal processing [22], where convolution filters are seen as a set of learnable parameters. The ChebNet [23] is proposed to reduce the computational cost of spectral GCNs that redefined the convolution filter in terms of Chebyshev polynomials bypassing the need for eigen decomposition of the graph Laplacian. In a follow-up work [15], a first order approximation of the Chebyshev polynomials was introduced. This further simplified the spectral GCN computation as the convolution operation reduces to a linear projection.

B. Emotion recognition.

Facial emotion recognition.: Recognizing facial expressions is the most common way of analyzing emotion. The majority of work rely on identifying an individual's facial expression from images or videos (fewer work on videos), and associating them to one of the basic emotion classes. Recent efforts in image-based recognition are focused on using CNNs and its variants [24], [25], and on using adversarial learning [26]. A few works have proposed to use attention networks to account for the context [27], [28], [29]. RNNs and 3D CNNs have been used for video-based emotion recognition due to their ability to capture the temporal information [30], [31]. Another line of work focuses on the dynamics of landmark points in faces extracted from videos. In this context, a deep temporal appearance geometry network has been proposed [32] that uses the landmark point geometry and a CNN for emotion recognition. Another recent work constructed a trajectory matrix from the landmark points and used them as inputs to a CNN [33].

Speech emotion recognition: Speech emotion recognition, especially using categorical labels, has been studied widely in the past years. Many speech emotion recognition systems still rely on low-level acoustic, prosodic and lexical features, that are then fed to deep models for classification. Other approaches use spectrograms (usually used as inputs to CNN models) [34] and even raw speech [35]. Recurrent models are prevalent due to their ability to capture the temporal dynamics of emotion [36], [35]. A 3D RNN model has been recently proposed for end-to-end modeling [37]. Attention-based techniques have been widely explored [36], [38], [39],

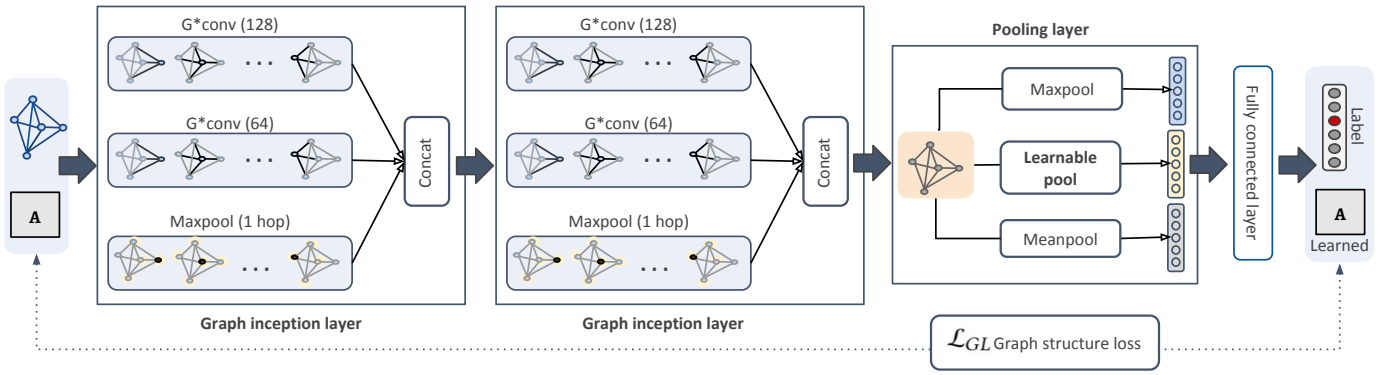


Fig. 2. Our proposed architecture, L-GrIN, consists of two graph inception layers (with a new spectral graph convolution layer) and a pooling layer (two fixed pooling layers and a learnable pooling layer). The inception layers produce node-level representations that are pooled to obtain a graph-level representation by the pooling layer. L-GRIN also learns the underlying graph structure (adjacency matrix) by jointly optimizing a classification loss and a graph structure loss.

while transformer-based architectures are gaining momentum in this field [40].

Body emotion recognition.: Body expressions are relatively less studied in emotion recognition. The existing literature is focused on using motion information in terms of low-level descriptors, such as joint angles, 3D positions, distance between joints, velocity and acceleration [5], [41], [42]. A trajectory learning approach [5] proposed to identify 'neutral' motion from input data, and used the deviation of a given input from the neutral motion as a feature for classifying emotions. Another recent work combined deep features with psychological attributes to detect emotion from 3D body pose using a random Forest classifier [41]. Gait information has also been considered for recognizing emotion, where a spatial GCN is used to detect the emotional state [42].

III. Proposed Approach

In this section, we describe our deep graph approach to emotion recognition. First, we construct a graph from dynamic input data following a generalized frame-to-node approach. Next, we propose a novel architecture that jointly performs graph learning and graph classification. This is achieved by optimizing over a new loss function that combines classification loss and a graph structure loss. The proposed architecture, L-GRIN, is illustrated in Fig. 2. Below, we describe each component of this network in detail.

A. Graph construction

Given a dynamic input sequence, our first task is to construct an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that can efficiently capture the emotion-related dynamics in the data, where $\mathcal{V}$ is the set of nodes with cardinality $|\mathcal{V}| = M$ and $\mathcal{E}$ is the set of all edges between the connected nodes. A representative description of $\mathcal{G}$ is typically given by an adjacency matrix $\mathbf{A} \in \mathbb{R}^{M \times M}$ which is symmetric for an undirected graph.

Our graph construction approach follows a *frame-to-node* transformation, where M frames in the data form the M graph nodes $\{v_i\}_{i=1}^{M} \in \mathcal{V}$ (see Fig. 3). A frame refers to a small temporal segment of the data, e.g., an audio segment of length 40ms. In order to encode the temporal information, a frame (node) should be connected with weights to a series of past and future nodes. An element $(\mathbf{A})_{ij} \in \mathbf{A}$ contains the weight corresponding to the edge $e_{ij} \in \mathcal{E}$ connecting v_i and v_j. Note that the graph structure is not naturally defined here, i.e., the elements in $\mathbf{A}$ are unknown. A common way to define the elements in $\mathbf{A}$ is through constructing a distance function manually [13]. However, this may result into a sub-optimal graph representation. Hence, we propose to learn the elements in $\mathbf{A}$ by jointly optimizing a structural loss combined with a classification loss. This loss function will be discussed in Section III-B.

In order to capture the emotion content at node level, we rely on modality-specific features or even, raw data. Each node v_i is thus associated with a *node feature* vector $\mathbf{n}_i \in \mathbb{R}^P$. A feature matrix $\mathbf{N} \in \mathbb{R}^{M \times P}$ consisting all the node feature vectors is defined as $\mathbf{N} = [\mathbf{n}_1, \mathbf{n}_2, \cdots, \mathbf{n}_M]^T$. Features for individual modalities is discussed in Section IV.

B. Learnable graph inception network

Given a set of (dynamic inputs transformed to) graphs $\{G_1, ..., G_N\}$ and their true labels $\{\mathbf{y}_1, ..., \mathbf{y}_N\}$, our task is to develop a deep graph architecture that is able to recognize the emotional content in the data. Since the graph structure is not naturally defined here, we also learn an optimal $\mathbf{A}$ from the training data with the underlying assumption that each graph has different node features but the same edge weights. We formulate this as a joint graph learning and graph classification problem.

A common GCN architecture takes the node feature matrix $\mathbf{N} \in \mathbb{R}^{M \times P}$ and the graph adjacency matrix $\mathbf{A}$ as inputs and produces a *node-level* representation matrix $\mathbf{Z} \in \mathbb{R}^{M \times Q}$, where Q is the dimension of the output feature vector at each node. A GCN layer $\mathbf{H}^{(k+1)}$ can be defined as a non-linear function of its previous layer as follows

$$\mathbf{H}^{(k+1)} = \sigma(\mathbf{A}\mathbf{H}^{(k)}\mathbf{W}^{(k)}) \tag{1}$$

where $\mathbf{W}^{(k)}$ is the weight matrix for the k^{th} layer of the neural network, σ is a non-linear activation function, such as a ReLU, and k is the layer number ($k = 0, \cdots K$). Note that

$\mathbf{H}^{(0)} = \mathbf{N}$ and $\mathbf{H}^{(K)} = \mathbf{Z}$. An effective improvement on this propagation rule has been recently proposed [15].

$$\mathbf{H}^{(k+1)} = \sigma(\mathbf{D}^{-\frac{1}{2}}(\mathbf{A}+\mathbf{I})\mathbf{D}^{-\frac{1}{2}}\mathbf{H}^{(k)}\mathbf{W}^{(k)}) \quad (2)$$

where $\mathbf{D}$ is the degree matrix of $\mathbf{A}$, and $\mathbf{I}$ is an $M \times M$ identity matrix. Note that the terms within the parenthesis in Eq. (2) is simply a linear projection, and can be re-written as

$$\mathbf{H}^{(k+1)} = \sigma(\hat{\mathbf{A}}\mathbf{H}^{(k)}\mathbf{W}^{(k)}) \quad (3)$$

where $\hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A}+\mathbf{I})\mathbf{D}^{-\frac{1}{2}}$.

We present a new GCN architecture, called L-GrIN (see Fig. 2), for joint graph learning and classification. It has the following four new components:

• **Non-linear spectral graph convolution ($\mathcal{G}^*$conv).** Motivated by a recent work on spatial graph neural network [43], we replace the linear projection in (3) by a multi-layer perceptron (MLP) layer, and replace $\hat{\mathbf{A}}$ by a learnable $\mathbf{A}$. Thus, instead of the linear layer in (3), we define a new spectral graph convolution layer $\mathcal{G}^*(\cdot)$ as follows:

$$\mathcal{G}^*(\mathbf{H}^{(k)}) = \sigma\Big(\mathrm{MLP}^{(k)}\big(\mathrm{ReLU}(\mathbf{A})\mathbf{H}^{(k)}\big)\Big) \quad (4)$$

where $\mathrm{MLP}(.)$ has two hidden layers with η neurons each, $\mathbf{A}$ is the learnable adjacency matrix and σ is a nonlinear activation function. $\mathbf{A}$ is learned through a joint optimization process described later in this section. The $\mathrm{ReLU}(\cdot)$ in Eq. (4) ensures the non-negativity of $\mathbf{A}$. We refer to the convolution operation defined above as $\mathcal{G}^*$**conv** in the rest of the paper.

• **Graph inception.** We extend the idea of inception layer in traditional CNNs [44] to the graph domain, and introduce a *graph inception* module in our architecture (see Fig. 2). Our graph inception layer consists of two graph convolution layers and one maxpool layer operating on directly connected (1-hop) neighbours only.

Given an input $\mathbf{H}^{(k)}$, the proposed graph inception layer is defined as follows:

$$\mathbf{H}^{(k+1)} = \Big[\mathcal{G}_1^*(\mathbf{H}^{(k)}) \,|\, \mathcal{G}_2^*(\mathbf{H}^{(k)}) \,|\, \mathrm{maxpool}(\mathbf{H}^{(k)})\Big] \quad (5)$$

where $|$ denotes concatenation of the node features, and $\mathcal{G}_1^*$ and $\mathcal{G}_2^*$ are two $\mathcal{G}^*$**conv** layers (see Eq. (4)) with different size of their MLP layers ($\eta = 128$ for $\mathcal{G}_1^*$ and $\eta = 64$ for $\mathcal{G}_2^*$). Hence, for an input of $\mathbf{H}^{(k)} \in \mathbb{R}^{M\times P}$, the inception layer produces an output $\mathbf{H}^{(k+1)} \in \mathbb{R}^{M\times(128+64+P)}$.

The motivation behind the inception layer is to be able to capture the emotion dynamics at multiple temporal scales. The two $\mathcal{G}^*$**conv** layers that yield embeddings of different dimensions can be interpreted as a *multiscale analysis* on graphs in spectral domain. Like a traditional inception layer in CNN, our graph inception layer also combines features from multiple scales allowing the network to have both width and depth. Our graph inception layer has fewer parameters (compared to inception networks in CNNs) enabling us to feed the input directly to the inception layer.

The maxpool function in Eq. (5) operates on every node separately. For each node v_i, we only consider its directly connected neighbors (1-hop), and maxpool over the embeddings along feature dimension. Note that as we start with a fully-connected graph, initially this operation is the same as maxpooling over all nodes, but this changes quickly as we start learning the graph structure.

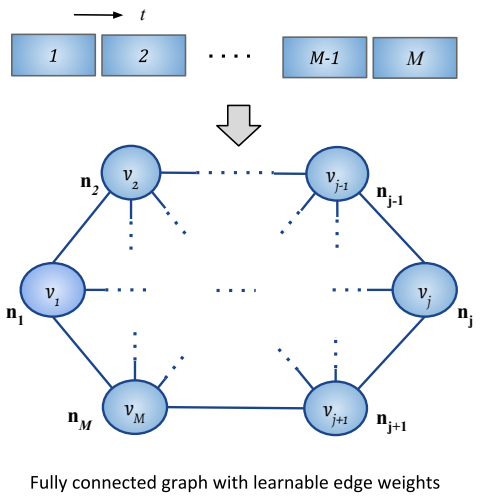


Fig. 3. Graph construction: Given a dynamic input sequence of M segments, a fully-connected graph with M nodes is constructed without making any assumption. The edge weights are learned during the training phase. Each node is associated with a node attribute vector $\mathbf{n}_i$.

• **Learnable pooling for graph-level representation.** Our objective is to classify entire graphs, as opposed to the more common task of classifying each node. Hence, we seek a *graph-level* representation $\mathbf{h}_G \in \mathbb{R}^Q$ as the output of our network. This can be obtained by pooling the node-level representations $\mathbf{H}^{(k)}$ at the K-th layer before passing them to the classification layer (see Fig.2). Common choices for pooling functions in graph domain are mean, max and sum pooling. Max and mean pooling often can not preserve the underlying information about the graph structure while sum pooling is shown to be a better alternative [43]. However, all these pooling functions treat every neighboring node with equal importance, which may not be optimal. To this end, we propose to *learn* a pooling function Ψ that combines the node embeddings from the K-th layer to produce an embedding for the entire graph. Additionally, we also use maxpool and meanpool and combine all the graph-level embeddings together. The pooling layer is thus defined as follows:

$$\mathbf{h}_G = \big[\mathrm{maxpool}(\mathbf{H}^{(K)}) \,|\, \Psi(\mathbf{H}^{(K)}) \,|\, \mathrm{meanpool}(\mathbf{H}^{(K)})\big] \quad (6)$$

$$\Psi(\mathbf{H}^{(K)}) = \mathbf{H}^{(K)}\mathbf{p}$$

where $\mathbf{p}$ has learnable weights to combine node-level embeddings to obtain a graph-level embedding.

• **Learnable adjacency** ($\mathbf{A}$). Recall that in our task the graph structure is not known. Although we can define such structure manually, results are sub-optimal. An effective approach would be to learn the graph structure (adjacency matrix) itself by jointly optimizing over a classification loss and graph learning loss. We assume that all videos have the same underlying graph structure containing the same number of nodes and edges. This largely simplifies our task of graph structure learning. The overall loss $\mathcal{L}$ for joint graph learning and

classification is composed of two components: (i) $\mathcal{L}_{GC}$: a graph classification loss, and (ii) $\mathcal{L}_{GL}$: a graph learning loss. The classification loss $\mathcal{L}_{GC}$ is defined as the cross-entropy loss:

$$\mathcal{L}_{GC} = -\sum_{n=1}^{N} \mathbf{y}_n \log \hat{\mathbf{y}}_n \tag{7}$$

where $\hat{\mathbf{y}}_n$ is the predicted label for the n^{th} sample. The graph learning loss, $\mathcal{L}_{GL}$, is designed to facilitate learning the pooling vector $\mathbf{p}$ and the adjacency matrix $\mathbf{A}$. This is defined as follows:

$$\mathcal{L}_{GL} = \underbrace{\lambda_1 \mathbf{e}^T(\mathbf{A}_d \odot \mathbf{A})\mathbf{e} + \lambda_2 \|\mathbf{A}\|_F^2}_{\text{graph structure loss}} + \underbrace{\lambda_3 \|\mathbf{p}\|_2^2}_{\text{learnable pooling}} \tag{8}$$

where $\odot$ denotes element-wise multiplication, $\mathbf{e}$ is an all-ones vector, $\|\cdot\|_F$ denotes Frobenius norm, λ_1, λ_2, and λ_3 control the relative weights of the three terms, and $\mathbf{A}_d$ is a structure matrix defined as follows:

$$(\mathbf{A}_d)_{ij} = (i-j)^2 \tag{9}$$

The structure matrix $\mathbf{A}_d$ forces the nodes that are temporally close to each other to have stronger relationship, i.e. higher weights in the $\mathbf{A}$. The larger the squared distance between two nodes v_i and v_j (frames), the smaller will be the weights in $(\mathbf{A})_{ij}$. The ReLU operation (see Eq. (4)) ensures the non-negativity of the elements in $\mathbf{A}$. The overall optimization is thus as follows:

$$\min_{\mathbf{A},\mathbf{p},\boldsymbol{\Theta}^{(1:k)}} \mathcal{L} = \min_{\mathbf{A},\mathbf{p},\boldsymbol{\Theta}^{(1:k)}} \left[\mathcal{L}_{GC} + \mathcal{L}_{GL}\right]$$

where, Θ denotes all other learnable network parameters across all graph convolution layers including its constituent MLP layers. Every term in the overall loss function $\mathcal{L}$ is differentiable, thereby allowing an end-to-end optimization.

IV. Experiments

We now present extensive experimental results and analysis to evaluate the performance of the proposed architecture for facial, speech and body emotion recognition.

A. Facial emotion recognition

Video databases: We use three large video emotion recognition databases for our experiments. The databases are chosen based on their popularity in emotion recognition literature.

The **RML** database [45] contains 720 videos of 6 basic emotions: *anger, disgust, fear, joy, sadness*, *surprise* collected when the subjects speak. The subjects are from various ethnic groups and speak different languages.

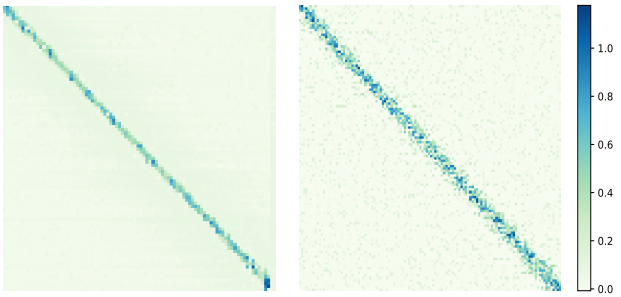
The **eNTERFACE** [46] is contains 1170 videos of 42 subjects with six basic emotion classes as RML. These emotions are the reactions after listening to six different short stories, where each person reads out 5 phrases based on their emotional reaction.

The **RAVDESS** database [47] contains 4904 videos labeled with 8 classes: *anger, calmness, disgust, fear, joy, neutral, sadness* and *surprise*. This is the largest video emotion database currently available.

TABLE I
FACIAL EMOTION RECOGNITION RESULTS ON THREE VIDEO DATABASES.

Model	Accuracy (%)			Params
	RML	eNTERFACE	RAVDESS	
*BLSTM	60.00	58.67	56.14	$\sim$ 1M
*GCN [15]	76.57	69.81	69.34	$\sim$ 102K
*PATCHY-SAN [19]	80.00	67.49	73.52	$\sim$ 52K
*PATCHY-Diff [48]	85.59	76.96	79.83	$\sim$ 71K
SENet [25]	71.20	79.22	71.06	$\sim$ 26M
AVEF [6]	82.48	85.69	-	-
KCFA [49]	82.22	76.00	-	-
OKL [50]	90.83	86.67	-	-
TJE [51]	-	-	72.30	-
***L-GrIN**	**94.11**	**87.49**	**85.65**	$\sim$ 120K

*use same node features



RAVDESS (90×90) **MPI** (120×120)

Fig. 4. *Learned* adjacency matrices for facial and body emotion recognition showing strong temporal dependency between neighboring segments. Darker values indicate higher weights.

Node features: The databases we use provide only raw video clips. We choose to use facial landmark points extracted from the video frames as node attributes. This is because landmark points are known to effectively capture the facial dynamics [52]. We extract 68 landmark points at every video frame using a state-of-the-art landmark detection method [53], resulting into node feature vectors of dimension $P = 136$.

Implementation details: We use a 10-fold cross-validation for all three databases, and report the average recognition accuracy in Table I. We fix the length of each input video to 90 frames yielding a graph with $M = 90$ nodes. The shorter videos are simply padded by duplicating frames from the beginning of the video (cyclic padding). Our network weights are initialized following the Xavier initialization. We set $\lambda_1 = \lambda_2 = 0.1$ and $\lambda_3 = 1 \times 10^{-4}$ (see Eq. 8). We used Adam optimizer with a learning rate of 0.01 and decay rate of 0.5 after each 50 epochs for all experiments. To initialize the learnable adjacency matrix $\mathbf{A}$, we generate a random matrix whose elements are drawn from a Normal distribution with zero mean and unit variance. We used Pytorch for implementing our model and the baselines, and an NVIDIA RTX-2080Ti GPU for all experiments.

Baselines, state-of-the-art: We compare our model against two competitive and relevant baselines as follows:

BLSTM. The first baseline is a Bidirectional LSTM (BLSTM), an extension of the traditional LSTMs [54], [55]. LSTM and its variants have been successfully used in sentiment analysis in language and speech. This BLSTM comprises 1-layered bidirectional cells with embedding size 300 followed by a fully connected layer.

GCN [15]. A natural baseline to compare with our model is a spectral GCN in its standard form (as in Eq. (3)). The original network [15] is designed for node classification and only yields node-level embeddings. To obtain a graph-level embedding, we used max and mean pooling at the end of convolution layers. The GCN uses a binary adjacency matrix constructed following the method used in graph-based action recognition [13].

In addition to the baselines, we compare with two state-of-the-art graph classification architectures:

PATCHY-SAN [19] is a recent architecture that learns CNNs for arbitrary graphs. This architecture is originally developed for graph classification.

PATCHY-Diff [48] is referred to an architecture where PATCHY-SAN is used in combination with the differentiable pooling layer between graph convolution layers proposed recently [48].

SENet [25], Squeeze and Excitation net is a state-of-the-art CNN architecture recently proposed for facial emotion recognition in videos.

Comparisons are also made with other existing works on the respective databases: AudioVisual Emotion Fusion (AVEF) [6], Kernel Crossmodal Factor Analysis (KCFA) [49], Optimized Kernel-Laplacian (OKL) [50] and Temporal Joint Embeddings (TJE) [51].

Results: Table I compares the performance of L-GrIN with all the methods mentioned above. Clearly, the proposed model outperforms all the existing methods by a significant margin, including the graph-based state-of-the-art architectures, such as PATCHY-SAN and PATCHY-Diff. Our model performs better than BLSTM - a class of models most commonly used in video-based emotion recognition. SENet is a very recent CNN architecture developed for emotion recognition, which also trails our model in terms of performance. When compared to the GCN baseline [15], L-GrIN improves the recognition accuracy by more than 10% on RML and eNTERFACE, and more than 5% on RAVDESS.

Also note that KCFA, OKL and TJE use both audio and visual information for recognition. Our model, even though uses only visual information, shows significant improvement over the audiovisual methods.

Fig. 4 shows the learned adjacency matrix for the RAVDESS database. The learned graph structure shows higher values closer to the diagonal i.e., the weights shared among the neighboring nodes. This indicates higher temporal dependencies locally and weaker dependency as we go further from a node.

B. Speech emotion recognition

Databases: We use the popular IEMOCAP database [57] for evaluating the performance of our model on speech emotion recognition. This database contains a total of 12 hours of data recorded in 5 sessions, where each session contains utterances from two speakers. The final database contains a total of 5531 utterances: 1103 *angry*, 1708 *neutral*, 1636 *happy* and 1084 *sad*.

TABLE II
SPEECH EMOTION RECOGNITION RESULTS ON IEMOCAP DATABASE.

Model	Accuracy (%)	Params
*BLSTM (baseline)	58.04	$\sim$ 0.8M
*GCN (baseline)	56.14	$\sim$ 78K
*PATCHY-SAN [19]	60.34	$\sim$ 60K
*PATCHY-Diff [48]	63.23	$\sim$ 68K
CNN [35]	58.52	$\sim$ 0.45M
CNN-LSTM [35]	59.23	$\sim$ 0.6M
Rep learning [56]	50.40	-
LSTM-CTC [4]	64.20	$\sim$ 0.4M
***L-GrIN**	**65.50**	$\sim$ 92K

* use same node features

Node features: We extract a set of low-level descriptors (LLDs) from the raw speech utterances as proposed for Interspeech2009 emotion challenge [58] using the OpenSMILE toolkit [59]. The feature set includes Mel-Frequency Cepstral Coefficients (MFCCs), zero-crossing rate, voice probability, fundamental frequency (F0) and frame energy. For each sample, we use a sliding window of length 25ms with a stride length of 10ms to extract the LLDs locally. Each feature is then smoothed using a moving average filter, and the smoothed version is used to compute their respective first order delta coefficients. In addition, motivated by a recent work on speech emotion recognition [60], we also add *spontaneity* as a binary feature. The spontaneity information comes with the database. Altogether this produces node feature vectors of dimension $P = 35$.

Implementation details: Each audio sample produces a graph of $M = 120$ nodes, where each node corresponds to a (overlapping) speech segment of length 25ms. Cyclic padding is used to make the samples of equal length, as before. We perform a 5-fold cross-validation and report the average unweighted accuracy in Table II. The unweighted accuracy, a standard evaluation strategy for IEMOCAP, does not take into account the class imbalances. It simply computes the total number of correct classifications across all classes. All other parameters and settings remain the same as before to show the generalizability of our model.

Baselines, state-of-the-art: Our model is compared with two baselines (BLSTM and GCN), two state-of-the-art graph-based architectures (PATCHY-SAN and PATCHY-Diff) as before. In addition, we also compare our model with four state-of-art methods in speech emotion recognition: CNN [35], CNN-LSTM [35], representation learning [56] and LSTM with Connectionist Temporal Modeling (LSTM-CTC) [4].

Results: Table II shows that our model performs better than the baselines and state-of-the-art methods on IEMOCAP. Our

Fig. 5. Motion capture recording set-up for the MPI database showing an actor posing for (left to right) T pose (reference), neutral and pride pose.

TABLE III
BODY EMOTION RECOGNITION RESULTS ON THE MPI DATABASE.

Model	Accuracy (%)	Parameters
*BLSTM	45.52	$\sim$ 0.9M
*GCN	56.03	$\sim$ 92K
*PATCHY-SAN [19]	48.42	$\sim$ 80K
*PATCHY-Diff [48]	55.29	$\sim$ 71K
Trajectory learning [5]	50.00	-
***L-GrIN**	**58.59**	$\sim$ 110K

* use same node features

model's performance may seem only slightly better (1.3%) compared to LSTM-CTC, but it requires 4 times more parameters than ours. LSTM-CTC uses 238-dimensional feature vectors where our feature dimension is only 35. Although PATCHY-Diff yields a competitive accuracy with a smaller model size on IEMOCAP, it trails L-GrIN by large margin on other databases. Note that PATCHY-SAN and PATCHY-Diff perform better than BLSTM and CNN-LSTM methods, indicating the effectiveness of graph-based methods in general.

C. Body emotion recognition

Databases: We use the MPI emotional body expression database [61] for our experiments. This database contains 1447 body motion samples of actors narrating coherent stories labeled with 11 emotions: *amusement, anger, disgust, fear, joy, neutral, pride, relief, sadness, shame,* and *surprise*. During their performance, a mocap system (device model: Xsens MVN) recorded the human motion using miniature inertial sensors. The system recorded dynamic 3D postures from 22 joints with a sampling rate of 120Hz.

Node features: For this database, we use the raw information provided by the mocap system. Each node contains the 3D positions and orientations (measure in terms of the Euler angles, pitch, yaw and roll) at a given time-step. These measurements come with the database. The feature consists of Euler angles from 22 joints and additional location information of the reference point. We use all the information (without any preprocessing) as node features, resulting into a vector of dimension $P = 72$.

Implementation details: Each input sample produces a graph of $M = 120$ nodes, where each node corresponds to a temporal segment of 120^{th} of a second. Cyclic padding is used as before. We perform a 5-fold cross-validation and report the average accuracy in Table III. All other network parameters remain the same as before.

TABLE IV
COMPARISON BETWEEN LEARNABLE AND FIXED POOLING STRATEGIES ON THE RML DATABASE. ALL EXPERIMENTS IN THIS TABLE USE THE SAME (BINARY) ADJACENCY MATRIX FOR FAIR COMPARISON.

Pooling	Accuracy (%)
Maxpool	89.76
Meanpool	90.23
Sortpool [62]	83.66
Learnable pool	**91.50**

TABLE V
COMPARISON BETWEEN LEARNABLE AND MANUALLY CONSTRUCTED GRAPH STRUCTURES. FOR FAIR COMPARISON, ALL EXPERIMENTS USE MAXPOOL TO CONVERT NODE EMBEDDINGS TO GRAPH EMBEDDINGS.

	Accuracy (%)			Params		
	RML	IEMOCAP	MPI	RML	IEMOCAP	MPI
Binary	89.5	61.4	53.6	113K	78K	96K
Weighted	62.4	54.3	49.0	113K	78K	96K
Learnable	**91.5**	**65.5**	**58.9**	120K	92K	110K

Baselines, state-of-the-art: Our model is compared with the baselines (BLSTM and GCN), the state-of-the-art graph-based architectures (PATCHY-SAN and PATCHY-Diff), and a recent work on this database, i.e., trajectory learning [5]. The trajectory learning system [5] models neural motion and analyzes the spectral difference between an expressive motion and a neutral motion in order to recognize the body expressions.

Results: Table III shows that L-GrIN outperforms the baselines and state-of-the-art methods on the MPI body expression database. Graph-based methods continue to perform well, indicating the effectiveness of graph-based methods for such tasks. Fig. 4 shows the learned adjacency $\mathbf{A}$ for the MPI database. As before, the learned graph structure exhibit higher temporal dependencies among the neighboring nodes.

D. Network analysis

Network size: Tables I, II and III list the number of learnable network parameters for the baselines, state-of-the-art graph-based architectures and the proposed L-GrIN. As mentioned earlier, a graph network largely reduces the number of learnable parameters as compared to the BLSTM or CNN architectures such as SENet (see Table I) without compromising the recognition accuracy. Our model has more parameters than the baseline GCN due to the inception layers and other learnable parameters, but also improves the recognition accuracy significantly. PATCHY-SAN and PATCHY-Diff have smaller network size compared to L-GRIN, but both trail L-GrIN in terms of performance on all databases. In case of facial emotion recognition, we discount the model size of the landmark detector in the comparison as it is common to all except SENet. For speech and body emotion recognition, no additional network was required as we used hand-crafted features and raw data.

TABLE VI
ABLATION STUDY ON THE RML DATABASE. EACH NEW COMPONENT IN L-GRIN CONTRIBUTES TOWARDS ITS PERFORMANCE.

$\mathcal{G}^*$conv	Inception	Learned $\mathbf{A}$	Learned $\mathbf{p}$	Accuracy (%)
-	-	-	-	76.57
✓	-	-	-	80.12
-	✓	-	-	87.58
-	-	✓	-	79.78
-	-	-	✓	82.86
-	-	✓	✓	84.21
✓	✓	-	-	90.65
✓	✓	✓	-	91.50
✓	✓	-	✓	91.50
✓	✓	✓	✓	**94.11**

TABLE VII
ANALYZING INCEPTION LAYER SETTINGS ON THE RML DATABASE.

Effect of filter size (η)	
Size of the two filters	**Accuracy (%)**
$(16, 32)$	90.82
$(32, 64)$	92.47
$(\mathbf{64}, \mathbf{128})$	**94.11**
$(128, 256)$	93.13
Effect of number of inception layers	
Number of layers	**Accuracy (%)**
1	91.77
2	**94.11**
3	90.78

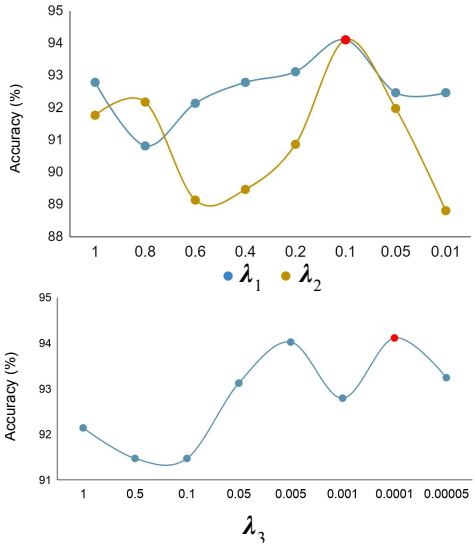


Fig. 6. Effect of the weight parameters in the loss function; experiments on the RML database.

Learnable vs. fixed pooling: Recall that to obtain a graph-level embedding from node-level embeddings, L-GrIN learns a pooling function (see Fig. 2). To show if learnable pooling indeed improves the recognition performance, we compare its performance with various fixed pooling strategies: max pooling, mean pooling and sort pooling (sortpool) [62]. Table IV presents the comparisons on the RML database in terms of facial emotion recognition accuracy, which clearly shows the advantage of learnable pooling over fixed pooling strategies. Similar trend is observed for other databases.

Learnable vs. manually constructed adjacency: An adjacency matrix represents the pairwise relationship between the graph nodes. When this information is not available naturally, a common practice is to manually construct an adjacency matrix. We argued earlier that this may result in sub-optimal graph structures which in turn affects the classification performance. We now compare the performance of leranable adjacency with two fixed adjacency matrices:

(i) *Binary adjacency:* a natural choice is a binary adjacency matrix as used for graph-based action recognition [13]. This is defined as $(\mathbf{A}_b)_{ij} = 1$ if $|i - j| = 1$ and 0 otherwise, i.e., a node (frame) is connected only to its subsequent and preceding node in the temporal direction.

(ii) *Weighted adjacency:* Another adjacency matrix is formed by using the squared ℓ_2 distance between two node attributes as their edge weight. This is defined as $(\mathbf{A}_w)_{ij} = \|\mathbf{n}_i - \mathbf{n}_j\|_2^2$.

Table V compares the performance of the proposed learnable adjacency with the two fixed adjacency matrices described above on the RML, IEMOCAP and the MPI databases. We chose one database from every modality. For this set of experiments we used only maxpooling to obtain the graph-level embeddings for fair comparison. Clearly, the learnable adjacency matrix shows consistent improvement in accuracy across all databases for a relatively small increase in model complexity (only 6% additional parameters). The results show that a learnable adjacency has better at generalizing across databases and modalities.

Ablation study: We performed exhaustive ablation experiments to investigate the contribution of each component we proposed to build L-GrIN. Table VI presents the ablation results on the RML database. We observe that each new component brings significant improvement (row 2 to row 5) over the performance of standard GCN [15] which has 76.57% recognition accuracy (the top row in Table VI). The introduction of the graph inception layer increases the recognition rate by 11%; when combined with our new graph convolution layer $\mathcal{G}^*$conv (Eq. (4)), the accuracy increases to 90.65%. Adding the learnable graph structure (learned $\mathbf{A}$) and learnable pooling bring the accuracy up to 94.11% both contributing to the accuracy. Removing either of the leanrable components reduces the accuracy by 2.61%. The ablation results show that each of the proposed components in our architecture is important, and contributes positively towards its superior performance. Similar ablation trend was observed for other databases.

Inception layer settings: We also investigate the effects of the graph inception layer hyperparameters: (i) the parameter η corresponding to the size of the graph convolution filters $\mathcal{G}_1^*$ and $\mathcal{G}_2^*$ in Eq. (5), and (ii) the number of graph inception layers in L-GrIN. First, we vary the filter dimensions (can be interpreted as scales) in the two inception layers and note how this correspond to the model's performance. Results for

TABLE VIII
CROSS-CORPUS PERFORMANCE OF OUR MODEL (L-GRIN) FOR FACIAL EMOTION RECOGNITION.

Trained on	Evaluated on	Accuracy (%)
RAVDESS	RML	81.94
	eNTERFACE	75.80
RML	RAVDESS	75.42
	eNTERFACE	61.71
eNTERFACE	RML	79.86
	RAVDESS	77.51

the RML database is presented in Table VII; similar trends have been observed for other databases. Results in Table VII show that we achieve the best performance for the combination of (64, 128), which is used in our model. Next, we vary the number of inception layers in the model, each with (64, 128) filter combination (see Table VII. We observe that reducing or increasing the number of inception layers from 2 results in a drop in performance. We chose to use two inception layers in the proposed model. It is obvious that the model size increases significantly as we add more inception layers or increase filter sizes within the layers. We notice a small drop in performance with larger filter sizes and with higher number of inception layers. This could be possibly due to over-smoothing and over-mixing of the node features. However, the over-smoothing effect is not as prominent as in many node classification tasks.

Analysis of the control weights: We also examine the impact of the weights controlling the various components of the loss function in Eq. (8), i.e., λ_1, λ_2 and λ_3. Fig. 6 shows that highest performance is achieved for $\lambda_1 = \lambda_2 = 0.1$ and $\lambda_2 = 0.0001$ (marked red in the plots) on the RML database. We use these λ values in our experiments.

Cross-corpus performance: Methods exhibiting superior performance on one corpus, often fall short when tested on another corpus having different statistical distributions. We investigated the ability of our model to generalize across databases by evaluating its cross-corpus performance. To this end, we trained L-GrIN on one database, followed by fine-tuning a fully-connected layer on the target database, without changing the graph structure (or other parameters) learned from the training database.

Results in Table VIII shows that our model can generalize well producing consistent results under cross-corpus evaluation. Our cross-corpus results higher accuracy compared to the same-corpus GCN accuracy. Cross-corpus results are comparable with the same-corpus performance of PATCHY-SAN. This shows the strength of the proposed architecture. It is worth noticing that the RML database (when used for training) does not have *neutral* and *calmness* emotion classes, but our model still recognizes those emotions on RAVDESS with 67.2% and 73.4% accuracy.

Network visualization: To get an insight into the learning process of our model, we visualized how it attends to different nodes. The video data are the most suitable for the visualization. We use our trained model, and then feed-forward each test video sample through the network, and identify the node (each node corresponds to a video frame) that responded most strongly towards the maxpooling layer. This yields a *salient* node corresponding to each input. We present the corresponding video frames - one example per emotion class for RML, eNTERFACE and RAVDESS databases in Fig. 7. The results show that the proposed model is able to learn the salient information from the input graphs such that it is representative of each emotion.

V. CONCLUSION

We proposed a novel, generalized graph architecture that can recognize emotion in a variety of dynamic input sequence. Our proposed architecture, L-GrIN, learns to detect emotion while jointly learning the underlying graph structure (adjacency matrix) and a pooling function to yield graph-level representation from node-level embeddings. We proposed a new spectral graph convolution operation and introduced the idea of inception in the graph domain. The advantage of our model lies in its state-of-the-art performance spanning three different modalities (video, audio and motion capture), with significantly fewer parameters compared to the CNNs and RNNs. This indicates that our model is suitable for applications in resource-constrained devices, such as smartphones.

We used both modality-specific features and even raw data as node features in this work. Our approach is not tied to any particular feature. In fact, our model can be trained end-to-end by combining it with modality-specific networks (e.g., a CNN) for feature extraction. The architecture we developed, although focuses on emotion recognition, is fairly generic. It will be applicable to a variety of classification tasks involving dynamic data, such as pose estimation, action recognition and visual speech recognition. Since our model makes no assumption about the graph structure, this is also applicable to common unstructured graphs. Future work will be directed towards building multimodal graph architectures taking advantage of the modality-agnostic architecture.

REFERENCES

[1] H. Li, J. Sun, Z. Xu, and L. Chen, "Multimodal 2d+ 3d facial expression recognition with deep fusion convolutional neural network," *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2816–2831.
[2] X. Pan, G. Ying, G. Chen, H. Li, and W. Li, "A deep spatial and temporal aggregation framework for video-based facial expression recognition," *IEEE Access*, vol. 7, pp. 48 807–48 815, 2019.
[3] S. Parthasarathy and C. Busso, "Semi-supervised speech emotion recognition with ladder networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2697–2709, 2020.
[4] W. Han, H. Ruan, X. Chen, Z. Wang, H. Li, and B. W. Schuller, "Towards temporal modelling of categorical speech emotion recognition." in *Interspeech*, 2018, pp. 932–936.
[5] A. Crenn, A. Meyer, R. A. Khan, H. Konik, and S. Bouakaz, "Toward an efficient body expression recognition based on the synthesis of a neutral movement," in *International Conference on Multimodal Interaction*, 2017, pp. 15–22.
[6] Y. Ma, Y. Hao, M. Chen, J. Chen, P. Lu, and A. Košir, "Audio-visual emotion fusion (avef): A deep efficient weighted approach," *Information Fusion*, vol. 46, pp. 184–192, 2019.
[7] W. Zhang, X. He, and W. Lu, "Exploring discriminative representations for image emotion recognition with cnns," *IEEE Transactions on Multimedia*, vol. 22, no. 2, pp. 515–523, 2019.
[8] H. Zhang and M. Xu, "Recognition of emotions in user-generated videos with kernelized features," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2824–2835, 2018.

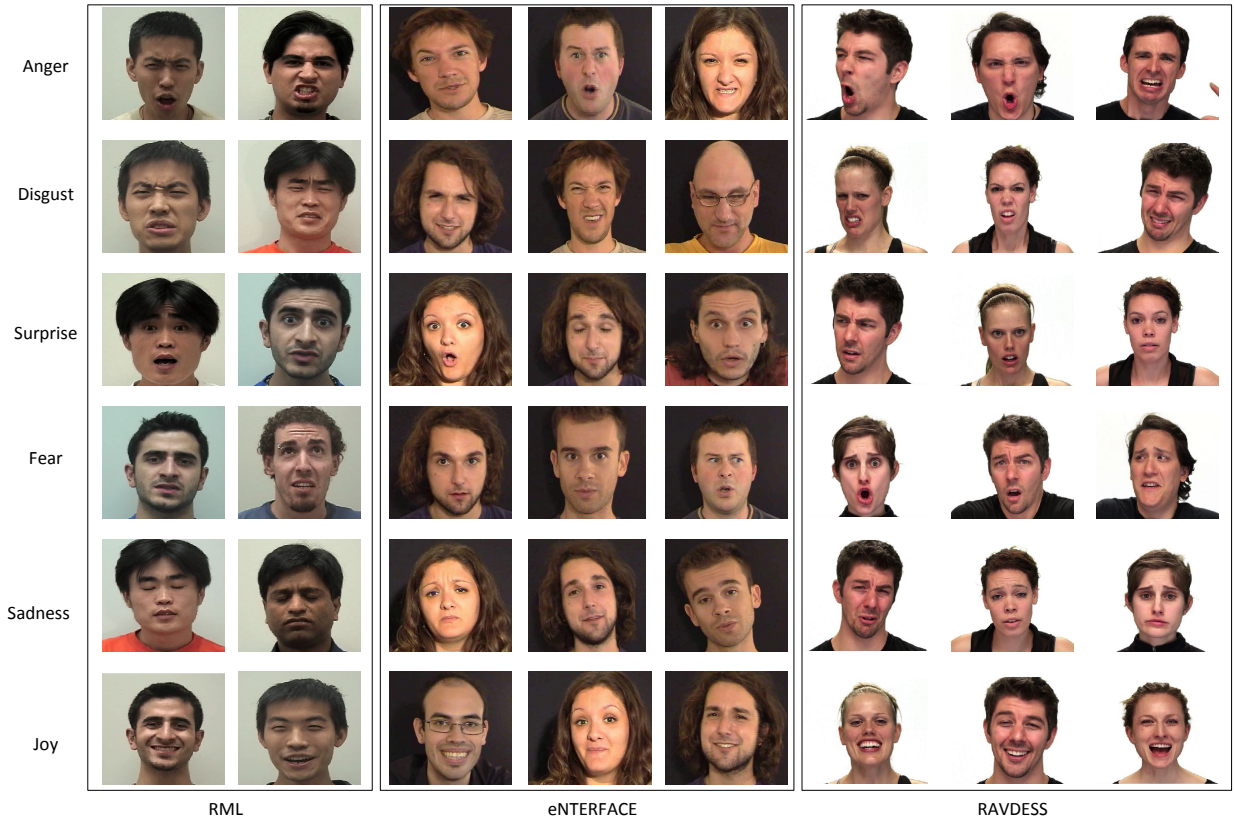


Fig. 7. Qualitative results showing the node (frame) for a graph input that generated the strongest response in our network. One result is displayed per class for the three databases. This shows that L-GRIN is able to learn the salient information for each emotion.

[9] G. Verma, E. G. Dhekane, and T. Guha, "Learning affective correspondence between music and image," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3975–3979.

[10] M. K. Lee, D. Y. Choi, D. H. Kim, and B. C. Song, "Visual scene-aware hybrid neural network architecture for video-based facial expression recognition," in *International Conference on Automatic Face & Gesture Recognition (FG)*, 2019, pp. 1–8.

[11] F. Mao, X. Wu, H. Xue, and R. Zhang, "Hierarchical video frame sequence representation with deep convolutional graph network," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 262–270.

[12] Y. Seo, M. Defferrard, P. Vandergheynst, and X. Bresson, "Structured sequence modeling with graph convolutional recurrent networks," in *International Conference on Neural Information Processing (NeurIPS*, 2018, pp. 362–373.

[13] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 7444–7452.

[14] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning (ICML)*, 2017, pp. 1263–1272.

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

[16] Z. Wang, L. Zheng, Y. Li, and S. Wang, "Linkage based face clustering via graph convolution network," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1117–1125.

[17] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 652–660.

[18] Y. Liu and K. Kirchhoff, "Graph-based semisupervised learning for acoustic modeling in automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 1946–1956, 2016.

[19] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *International Conference on Machine Learning (ICML)*, 2016, pp. 2014–2023.

[20] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 1024–1034.

[21] B. Xu, H. Shen, Q. Cao, Y. Qiu, and X. Cheng, "Graph wavelet neural network," in *International Conference on Learning Representations (ICLR)*, 2019.

[22] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun, "Spectral networks and locally connected networks on graphs," in *International Conference on Learning Representations (ICLR)*, 2014.

[23] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2016, pp. 3844–3852.

[24] R. Vemulapalli and A. Agarwala, "A compact embedding for facial expression similarity," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 5683–5692.

[25] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7132–7141.

[26] B. Pan, S. Wang, and B. Xia, "Occluded facial expression recognition enhanced through privileged information," in *ACM Multimedia*, 2019, pp. 566–573.

[27] J. Lee, S. Kim, S. Kim, J. Park, and K. Sohn, "Context-aware emotion recognition networks," in *International Conference on Computer Vision (ICCV)*, 2019, pp. 2893–2901.

[28] P. D. Marrero Fernandez, F. A. Guerrero Pena, T. Ren, and A. Cunha, "Feratt: Facial expression recognition with attention net," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019.

[29] M. O. Cordel, S. Fan, Z. Shen, and M. S. Kankanhalli, "Emotion-aware human attention prediction," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4026–4035.

[30] S. Jaiswal and M. Valstar, "Deep learning the dynamic appearance and shape of facial action units," in *Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1–8.

[31] Y. Fan, X. Lu, D. Li, and Y. Liu, "Video-based emotion recognition using cnn-rnn and c3d hybrid networks," in *International Conference on Multimodal Interaction (ICMI)*, 2016, pp. 445–450.

[32] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, "Joint fine-tuning in deep neural networks for facial expression recognition," in *International Conference on Computer Vision (ICCV)*, 2015, pp. 2983–2991.
[33] J. Yan, W. Zheng, Z. Cui, C. Tang, T. Zhang, and Y. Zong, "Multi-cue fusion for emotion recognition in the wild," *Neurocomputing*, vol. 309, pp. 27–35, 2018.
[34] Z. Huang, M. Dong, Q. Mao, and Y. Zhan, "Speech emotion recognition using cnn," in *ACM Multimedia*, 2014, pp. 801–804.
[35] S. Latif, R. Rana, S. Khalifa, R. Jurdak, and J. Epps, "Direct modelling of speech emotion from raw speech," *Interspeech*, pp. 3920–3924, 2019.
[36] S. Mirsamadi, E. Barsoum, and C. Zhang, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2227–2231.
[37] Z. Peng, Z. Zhu, M. Unoki, J. Dang, and M. Akagi, "Auditory-inspired end-to-end speech emotion recognition using 3d convolutional recurrent neural networks based on spectral-temporal representation," in *International Conference on Multimedia & Expo (ICME)*, 2018, pp. 1–6.
[38] C.-W. Huang and S. S. Narayanan, "Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition," in *International Conference on Multimedia and Expo (ICME)*, 2017, pp. 583–588.
[39] Y. Gu, X. Lyu, W. Sun, W. Li, S. Chen, X. Li, and I. Marsic, "Mutual correlation attentive factors in dyadic fusion networks for speech emotion recognition," in *ACM Multimedia*, 2019, pp. 157–166.
[40] L. Tarantino, P. N. Garner, and A. Lazaridis, "Self-attention for speech emotion recognition," *Proc. Interspeech 2019*, pp. 2578–2582, 2019.
[41] T. Randhavane, A. Bera, K. Kapsaskis, U. Bhattacharya, K. Gray, and D. Manocha, "Identifying emotions from walking using affective and deep features," *CoRR*, vol. abs/1906.11884, 2019.
[42] U. Bhattacharya, T. Mittal, R. Chandra, T. Randhavane, A. Bera, and D. Manocha, "STEP: spatial temporal graph convolutional networks for emotion perception from gaits," *CoRR*, vol. abs/1910.12906, 2019.
[43] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *International Conference on Learning Representations (ICLR)*, 2019.
[44] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.
[45] Y. Wang and L. Guan, "Recognizing human emotional state from audiovisual signals," *IEEE Transactions on Multimedia*, vol. 10, no. 5, pp. 936–946, 2008.
[46] O. Martin, I. Kotsia, B. Macq, and I. Pitas, "The enterface'05 audio-visual emotion database," in *International Conference on Data Engineering Workshops (ICDEW)*, 2006, pp. 8–8.
[47] S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english," *PloS one*, vol. 13, no. 5, p. e0196391, 2018.
[48] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018, pp. 4800–4810.
[49] Y. Wang, L. Guan, and A. N. Venetsanopoulos, "Kernel cross-modal factor analysis for information fusion with application to bimodal emotion recognition," *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 597–607, 2012.
[50] K. P. Seng, L.-M. Ang, and C. S. Ooi, "A combined rule-based & machine learning audio-visual emotion recognition approach," *IEEE Transactions on Affective Computing*, vol. 9, no. 1, pp. 3–13, 2016.
[51] E. Ghaleb, M. Popa, and S. Asteriadis, "Multimodal and temporal perception of audio-visual cues for emotion recognition," in *International Conference on Affective Computing & Intelligent Interaction (ACII)*, 2019.
[52] J. Gu, X. Yang, S. De Mello, and J. Kautz, "Dynamic facial analysis: From bayesian filtering to recurrent neural network," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1548–1557.
[53] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks)," in *International Conference on Computer Vision (ICCV)*, 2017, pp. 1021–1030.
[54] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.
[55] M. Tan, C. dos Santos, B. Xiang, and B. Zhou, "Lstm-based deep learning models for non-factoid answer selection," *International Conference on Learning Representations workshop*, 2016.
[56] S. Ghosh, E. Laksana, L.-P. Morency, and S. Scherer, "Representation learning for speech emotion recognition." in *Interspeech*, 2016, pp. 3603–3607.
[57] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "Iemocap: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, no. 4, p. 335, 2008.
[58] B. Schuller, S. Steidl, and A. Batliner, "The interspeech 2009 emotion challenge," in *Interspeech*, 2009.
[59] F. Eyben, M. Wollmer, and B. Schuller, "Openear—introducing the munich open-source emotion and affect recognition toolkit," in *International Conference on Affective Computing and Intelligent Interactions (ACII)*, 2009, pp. 1–6.
[60] K. Mangalam and T. Guha, "Learning spontaneity to improve emotion recognition in speech," in *Interspeech*, 2018, pp. 946–950.
[61] E. Volkova, S. De La Rosa, H. H. Bulthoff, and B. Mohler, "The mpi emotional body expressions database for narrative scenarios," *PloS One*, vol. 9, no. 12, 2014.
[62] M. Zhang, Z. Cui, M. Neumann, and Y. Chen, "An end-to-end deep learning architecture for graph classification," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 4438–4445.

Amir Shirian is currently a PhD student in the Department of Computer Science at the University of Warwick, UK. He has received his BSc (2015) and MSc (2018) degrees in Electrical Engineering from the University of Tehran, Iran. His research interests include multimodal signal processing and machine learning with applications to emotion and behavior understanding.

Subarna Tripathi is a Research Scientist at Intel Labs, Intel Corporation, San Diego, US. She received the PhD degree in Electrical and Computer Engineering from the University of California San Diego (UCSD). Her current research involves building computation models in the area of computer vision with a focus on scene graphs, graph embeddings, and scene parsing. She has been an Area Chair of Women in Machine Learning (WiML). She has served in the Program Committee of several conferences including CVPR, ICCV, ECCV and AAAI.

Tanaya Guha is an Assistant Professor in the Department of Computer Science at the University of Warwick, UK. She has received her PhD in Electrical and Computer Engineering from the University of British Columbia (UBC), Vancouver, Canada. Her research is focused on modeling and analysis of multimedia data combining machine learning and signal processing with applications in media content analysis, healthcare and smart surveillance. She was an Area Chair for INTERSPEECH'16 and ICME'20.