Geometric Multimodal Deep Learning With Multiscaled Graph Wavelet Convolutional Network

Maysam Behmanesh, Peyman Adibi, Sayyed Mohammad Saeed Ehsani, and Jocelyn Chanussot, Fellow, IEEE

Abstract—Multimodal data provide complementary information of a natural phenomenon by integrating data from various domains with very different statistical properties. Capturing the intramodality and cross-modality information of multimodal data is the essential capability of multimodal learning methods. The geometry-aware data analysis approaches provide these capabilities by implicitly representing data in various modalities based on their geometric underlying structures. Also, in many applications, data are explicitly defined on an intrinsic geometric structure. Generalizing deep learning methods to the non-Euclidean domains is an emerging research field, which has recently been investigated in many studies. Most of those popular methods are developed for unimodal data. In this article, a multimodal graph wavelet convolutional network (M-GWCN) is proposed as an end-to-end network. M-GWCN simultaneously finds intramodality representations by applying the multiscale graph wavelet transform to provide helpful localization properties in the graph domain of each modality and cross-modality representation by learning permutations that encode correspondences among various modalities. M-GWCN is not limited to either the homogeneous modalities with the same number of data or any prior knowledge indicating correspondences between modalities. Several semi-supervised node classification experiments have been conducted on three popular unimodal explicit graph-based datasets and five multimodal implicit ones. The experimental results indicate the superiority and effectiveness of the proposed methods compared with both spectral graph domain convolutional neural networks and state-of-the-art multimodal methods.

Index Terms—Geometric deep learning, graph convolution neural networks, graph wavelet transform, multimodal learning, spectral approaches.

I. INTRODUCTION

MULTIMODAL data are generally composed of several heterogeneous data gathered from multiple real-world phenomena. Different modalities provide parts of the description of a phenomenon from various source domains usually with very different statistical properties. Unlike the unimodal data that may indicate only partial information of an entity, multimodal data provide complementary information by leveraging and fusing modality-specific knowledge [1]. Although each modality has its distinct statistical properties, the modalities are usually semantically correlated. Multimodal learning tries to improve the performance in applications with multimodal data by discovering the hidden intramodality and cross-modality correlations.

However, although recent multimodal models have been focused on Euclidean data, there are two major situations in which data should be processed in non-Euclidean domains: first, in the cases in which data in various modalities are implicitly represented based on their geometric structures, and second, when data are generated in non-Euclidean geometric domains and inherently defined, for example, as a graph. These applications represent complex relationships and interdependencies among objects [2], including social networks, citation networks, networks of the spread of epidemic diseases, e-commerce networks, brain’s neuronal networks, and biological regulatory networks.

With the emergence of geometric structural data in real-world applications, many works have investigated generalizing deep learning methods to the non-Euclidean domains [2], [3]. As the most popular challenges for the graphs domain data, graph neural networks (GNNs) perform filtering operations directly on the graph via the graph weights [3] and graph convolutional networks (GCNs) learn the local meaningful stationary properties of the input signals through specifically designed convolution operator on graphs [4]. However, complex geometric structures in graphs can be encoded with more powerful mathematical tools in many spatial or spectral graph-based methods [5]. Nevertheless, most of these popular methods are developed for unimodal data and have difficulties in coping with multimodal problems.

One of the remarkable deficiencies of the previous multimodal data analysis methods is their limitation to completely model the real-world situations. More specifically, these methods are generally restricted by considering two simplifying assumptions: 1) there is the same number of data samples in each modality (the homogeneity assumption) and 2) at least partial correspondences and/or noncorrespondences between modalities are given as prior knowledge. Obviously, these assumptions are unrealistic in practical machine learning scenarios [6].

The main purpose of this work is to introduce a novel spectral GCN on graph-based multimodal data in a more practical scenario, in which there are heterogeneous data in each modality, the same number of data samples may not be available in different modalities, and correspondences between modalities are unknown. It is only known that the data points of various modalities are sampled from the same phenomena.
In this work, we generalize GCNs to be applicable on multimodal data in a more practical scenario as an important problem that has rarely been addressed. Specifically, we proposed a multimodal graph wavelet convolutional network (M-GWCN), with an end-to-end learning framework. The proposed M-GWCN model takes advantage of graph wavelet transforms as a multiresolution analysis of graph signals. Therefore, it includes the stacked architecture that utilizes graph wavelets convolution with multiple scaling parameters in parallel. The main problem for multimodal problems in the practical scenario is to explore various localities in different modalities to find the potential correspondences among them. Thus, the proposed architecture tries to find these localities by simultaneously localizing signal content in both spatial and spectral domains (graph wavelets from the spectral graph theory point of view are studied in [7]).

One of the superiorities of the M-GWCN structure is that it is equipped with initial residual connections, which makes the representation at each layer to contain a portion of the initial features and is less prone to oversmoothing. To avoid the expensive spectral decomposition for diagonalizing the graph Laplacian, we apply the Chebyshev polynomial method for approximating the graph wavelet bases. This approximation enables the M-GWCN to be applicable to large graphs.

As will be seen, the proposed M-GWCN model consists of two main parts. In the first part, a graph convolution is conducted by applying multiscaled graph wavelet transforms in each modality. Graph wavelet bases obtained with different scales provide valuable localization properties in the graph domain of each modality. These properties are applied for obtaining intramodality representation of feature vectors of each graph. In the second part, cross-modality representations are computed through the data fusion layer. In this layer, permutations for encoding the cross-modality correlations are learned by applying new loss function and regularization terms.

The main contributions of this article are briefly summarized as follows.

1) We generalize the spectral GCN model to multimodal graph-based data domains as a problem that has rarely been addressed.
2) We consider a general scenario for multimodal problems with unpaired data in the heterogeneous modalities, which needs no prior knowledge.
3) We present a new stacked spectral GCN architecture equipped with initial residual connections that in parallel applies graph wavelet transforms with different scaling parameters and makes the representation at each layer less prone to oversmoothing.
4) We develop a new efficient network with end-to-end learning that simultaneously applies feature mapping on each modality and explores permutations for representing the cross-modality correlations among various modalities.
5) We experimentally demonstrate the superiority and effectiveness of the proposed M-GWCN model on three unimodal explicit graph-based datasets and five multimodal datasets, which are not explicitly defined as graphs, but the graphs are constructed based on them implicitly.

The remainder of this article is organized as follows. Related works are reviewed in Section II. In Section III, we overview the background and the basic notations. Our proposed methods are introduced in Section IV. Experimental results are presented and discussed in Section V. Finally, Section VI concludes this article.

II. RELATED WORKS

A. Multimodal Learning

Many multimodal learning methods are limited to the problems with homogeneous data, which have the same number of data samples in all modalities. For example, MVML-LA is a multiview method, which learns a common discriminative low-dimensional latent space by preserving the geometric structure of the original input [8]. MVDL-CV is a new multiview method, which obtains the sparse representation of the sample by learning a particular dictionary for each view and determines the similarity of samples using a regularization term between two dictionaries [9]. GPLVM represents multiple modalities in a common subspace using the Gaussian process latent variable model [10]. Another work provides a unifying framework for multiclass classification by encompassing vector-valued manifold regularization and coregularized multiview learning [11].

Moreover, some efforts have also been made to apply multimodal data analysis techniques with heterogeneous assumption about modalities to several practical applications, inspired by capturing different information from various modalities. These types of multimodal approaches try to provide more confident estimation of the problems, e.g., using multifeature fusion models to cope with abundant unlabeled data [12] or generalizing spectral clustering [13]. Some multimodal learning methods are geometry-aware and try to extend different diffusion and spectral methods to the multimodal setting. A nonlinear multimodal data fusion method captures the intrinsic structure of data and relies on minimal prior model knowledge [14]. A multimodal image registration method uses the graph Laplacian to capture the intrinsic structure of data in each modality by introducing a new structure preservation criterion based on the Laplacian commutativity [15]. Although this type of approaches improves the model performance, there is an implicit assumption that the pointwise correspondence between different views must be predetermined.

However, regardless of whether data samples in various modalities are homogeneous or heterogeneous, these methods assume that they are completely paired, and all modalities have the same number of data samples. Since the correspondence knowledge between various modalities, which is essential for the above learning multimodal methods, may not be available in many practical scenarios, working independent of this prior knowledge is of key importance. Thus, it has been focused in continue on several methods, which try to reduce dependence on the expert manipulation process.
As one of the first efforts, Eynard et al. [16] proposed a multimodal approach based on weakly paired data samples, in which this partial correspondence information is predetermined using an expert manipulation process. The method presented in [17] first extends the given correspondence information between modalities using functional mapping idea on the data manifolds of the respected modalities and then uses all correspondence information to simultaneously learn an underlying low-dimensional common manifold by aligning the manifolds of different modalities. Another multimodal manifold learning approach, called local signal expansion for joint diagonalization (LSEJD), was proposed in [18], which uses the intrinsic local tangent spaces to expand the initial correspondence knowledge.

Although these two later methods greatly expand correspondence information, they still depend on the little basic prior knowledge of correspondences. In another recent work [6], we proposed a multimodal learning method that is independent of any prior expert knowledge even about partial alignment/misalignment between different modalities. The main idea of this work consists in finding a functional map between local descriptors in different modalities. For this purpose, the spectral graph wavelet transform is applied to extract local descriptors on graph Laplacian of each modality.

Inspired by the successful efficiency of the spectral graph wavelet transform in exploring various localities [6], we try in this article to take this advantage on GCNs and provide a more advanced application of spectral graph wavelet transform. Unlike [6] that utilized the spectral graph wavelet signatures (SGWSs) to identify the localities of each manifold, in this work, we exploit wavelet transform for providing a proper set of bases and apply them for projecting graph signals from the vertex domain into the spectral domain. The key benefit of wavelet bases with different scales is the exploration of cross-modality correlations among various modalities, which enables the spectral GCN model to be generalized to multimodal graph-based data domains. We compare the performances of the proposed method with this recent method [6] in Section V, demonstrating the superiority of using graph wavelet bases in a GCN model compared with using them as a descriptor in a functional map problem in [6].

B. Graph Convolutional Neural Networks

Inspired by the success of CNNs in the Euclidean domain, a large number of methods are proposed to generalize CNNs to non-Euclidean and especially graph domains. These methods are classified into two categories: spatial methods and spectral ones.

In spatial graph-based methods, the convolution on each vertex is directly defined by aggregating feature information from all its neighbors. Different spatial methods provide various weighted average functions for characterizing the node influences in the neighborhoods. NN4G [19] performs graph convolutions by summing up the nodes neighborhood information directly. The message-passing neural network (MPNN) [20] treats graph convolution as a message-passing process in which information can be passed from one node to another along edges directly. To overcome the limitations of MPNN-based methods in distinguishing different graph structures based on the graph embedding, the graph isomorphism network (GIN) [21] is proposed. GIN adjusts the weight of the central node in a neighborhood by a learnable parameter. GraphSage [22] performs graph convolution by adopting a sampling strategy to obtain a fixed number of neighbors for each node. Graph attention network (GAT) [23] applies graph convolution by adopting attention mechanisms to learn the relative weights between two connected nodes.

Spectral graph-based methods define convolution by leveraging graph Fourier transform to convert signals defined in the vertex domain into the spectral domain using the convolution theorem. Spectral CNN (SCNN) [24] is the first spectral method, in which its graph convolution layer projects the input graph signal to a new space using graph Fourier transform. To overcome the limitations of the basic SCNN, Chebyshev SCNN (ChebyNet) [25] is introduced, which applies as a fast localized filter approximation to find the desired approximate filter response through the Chebyshev expansion. CayleyNet is another spectral method that uses Cayley polynomials filters. Unlike ChebyNet, CayleyNet can detect narrow frequency bands with a small number of filter parameters [26]. To address the limitations of spectral graph CNN methods in providing beneficial localization properties, GWNN is presented to apply graph wavelets as a set of bases instead of eigenvectors of graph Laplacian [27]. A new spectral method is proposed in [28] that offers a more flexible frequency response. It captures a better global graph structure using the autoregressive moving average (ARMA) filter.

C. Multimodal GNNs

Generalizing GNN to graph-based multimodal data is an important problem that has rarely been addressed. In a recent effort, a multimodal GNN method is presented for visual question-answering tasks [29]. This method represents the image as three graph-based modalities and refines the features of nodes by passing a message from one graph to another. Inspired by the message-passing idea of GNNs, a multimodal GCN (MMGCN) framework is proposed in [30]. MMGCN enriches the representation of each node by leveraging information interchange in various modalities to capture user preferences in a recommender system. Motivated by a graph-based structure in addressing the long unaligned sequences, an MMGCN-based method is proposed in [31] to investigate the effectiveness of GNN in modeling the multimodal sequential data. The edge-adaptable GCN (EA-GCN) method for disease prediction is presented in [32]. EA-GCN represents various modalities as a population graph using an edge adapter and applies GCN for semisupervised node classification.
an undirected weighted graph $G_m = (V_m, E_m, A_m)$, where $V_m$ is a set of $N_m$ vertices, $E_m$ is the set of $E_m$ edges, and $A_m \in \mathbb{R}^{N_m \times N_m}$ is a weighted adjacency matrix representing connection weights between vertices.

Let graph signal $x_m = [x_{m,1}, \ldots, x_{m,N_m}]^T \in \mathbb{R}^{N_m \times d_m}$ be a collection of all feature vectors associated with $l_m$ labeled and $u_m$ unlabeled vertices, where $N_m = l_m + u_m$.

Consider $y_m = [y_{m,1}, \ldots, y_{m,N_m}]^T \in \{0, 1\}^{N_m \times C}$ as the label matrix of data sample in modality $m$ for a C-class classification problem. If the vertex $x_{m,j}$ belongs to the $k$th class, then $y_{m,j}$ contains 1 in the $k$th location and 0 in all others. For unlabeled data sample $x_{m,k}$, the vector $y_{m,k}$ has $-1$ in all $C$ locations.

In this article, it is assumed that the sample correspondences information across different modalities is unknown. Also, whenever there is no emphasis on a specific modality, symbols are considered in the generic notation, e.g., $G$ is used to show the data graph instead of $G_m$, which indicates a specific modality $m$.

B. Graph Spectral Geometry

The symmetric normalized Laplacian matrix of graph $G_m$ is defined as $L_m = D_m^{-1/2}(D_m - A_m)D_m^{-1/2}$, where $D_m = \text{diag}(\sum_{k \neq l} a_{m}(k, l))$. For Laplacian matrices $[L_m \in \mathbb{R}^{N_m \times N_m}]_{m=1}^M$ of all modalities, there are unit eigenspaces $U_m$s, such that $L_m = U_m \Lambda_m U_m^T$, where matrices $U_m = [u_{m,1}, \ldots, u_{m,N_m}]$ and $\Lambda_m = \text{diag}(\lambda_{m,1}, \ldots, \lambda_{m,N_m})$ contain the orthogonal eigenvectors and their corresponding eigenvalues (spectrum), respectively. The eigenvectors play the role of Fourier basis in a classical harmonic analysis, and the eigenvalues can be interpreted as frequencies. For a given graph signal $f = (f_{m,1}, \ldots, f_{m,N_m})^T \in \mathbb{R}^{N_m}$ on the vertices of $G_m$, $\hat{f} = U_m^T f$ performs the graph Fourier transform, and $f = U_m \hat{f}$ is its inverse.

C. Spectral GCN

According to the convolution theorem, the spectral graph convolution of signal $x$ with the filter $g \in \mathbb{R}^N$ on the graph $G$ can be defined as an elementwise product of their Fourier transform as $x * g = U((U^T x) \odot (U^T g))$, where $U$ is the eigensapce of Laplacian matrix of graph $G$. By denoting $g_0 = \text{diag}(U^T g)$, the spectral graph convolution can be written in the form of matrix multiplication as $x * g_0 = U g_0 U^T x$ [25].

The spectral convolution layer $k$ is defined by extending CNN to graph $G$ as follows [24]:

$$X^{(k)}(:,j) = \sigma \left( \sum_{i=1}^{f_k} \sum_{r=1}^{R} \theta_{i,j}^{(k)}(\lambda_r)u_i u_i^T X^{(k-1)}(:,i) \right)$$

$$= \sigma \left( \sum_{i=1}^{f_k} U_R \theta_{i,j}^{(k)}(\Lambda_R) U_R^T X^{(k-1)}(:,i) \right)$$

where $U_R$ is an $N \times R$ matrix of first $R$ eigenvectors of Laplacian matrix $L$ (corresponding to its least eigenvalues); $g_0 = \theta_{i,j}^{(k)}(\Lambda_R)$ is an $R \times R$ diagonal matrix of spectral multipliers representing the learnable parameters in layer $k$;

$$X^{(k)} = \{X^{(k)}(:,1), \ldots, X^{(k)}(:,f_k)\} \in \mathbb{R}^{N \times f_k}$$

is the input signal, including $f_k$ features (channels); $X^{(k)} = \{X^{(k)}(:,1), \ldots, X^{(k)}(:,f_k)\} \in \mathbb{R}^{N \times f_k}$ is the output signal, including $f_k$ features; and $\sigma(\cdot)$ is a nonlinear activation function [e.g., rectified linear unit (ReLU)].

In this equation, parameter $R$ keeps the locality of filter in the spectral domain using the $R$ lowest frequency harmonics.

The main drawback of the SCNN is its computational complexity because the eigendecomposition problem of Laplacian matrix $L$ is too expensive. Furthermore, the resulting dense eigenvectors $U$ prevent achieving the sparse multiplications.

As one of the main efforts, ChebyNet [25] model is tried to address these challenges through approximating the desired filter response $g_0$ using the Chebyshev expansion as follows:

$$g_0 = \sum_{i=0}^{R} \theta_i T_i(\bar{\Lambda})$$

where $\bar{\Lambda} = 2\Lambda/\lambda_{\text{max}} - 1$ is the rescaled spectrum in $[-1, 1]$, $\theta = [\theta_0, \ldots, \theta_R]$ is the $(R+1)$-dimensional vector of the polynomial coefficients parametrizing the filter and is optimized during the training, and $T_i(\bar{\Lambda}) = 2\bar{\Lambda}T_{i-1}(\bar{\Lambda}) - T_{i-2}(\bar{\Lambda})$ is the Chebyshev polynomial of order $i$ that defines recursively with $T_0(\bar{\Lambda}) = 1$ and $T_1(\bar{\Lambda}) = \bar{\Lambda}$.

The convolution of graph signal $x$ with this defined filter $g_0$ is obtained as follows:

$$x * g_0 = U(\sum_{i=0}^{R} \theta_i T_i(\bar{\Lambda})) U^T x = \sum_{i=0}^{R} \theta_i T_i(\bar{\Lambda}) x$$

where $\bar{\Lambda} = 2\Lambda/\lambda_{\text{max}} - 1$. The resulting convolution layer $k$ is now defined as

$$X^{(k)} = \sigma \left( \sum_{i=0}^{R} T_i(\bar{\Lambda}) X^{(k-1)} W_i^{(k)} \right)$$

where $W_i^{(k)} \in \mathbb{R}^{f_{k-1} \times f_k}$ indicates the $i$th trainable weight matrix in layer $k$. A specific polynomial of order $i$ in (4) covers the $i$-hop neighborhood and ignores the impact of the farther neighbors.

GCN [33] presents a simplified version of the Chebyshev filter. It reduces complexity and overfitting by setting $R = 1$, $\lambda_{\text{max}} = 2$, and $W_i = -W_i = W$, and substituting $\bar{\Lambda}$ by $\tilde{\Lambda} = D^{-1/2} \tilde{A} D^{-1/2}$ in (4), where $\tilde{\Lambda} = \Lambda + 1$ and $\tilde{A}_{ij} = \sum_j \tilde{A}_{ij}$. Thus, the convolution layer $k$ of GCN is obtained as

$$X^{(k)} = \sigma (\tilde{\Lambda}X^{(k-1)} W^{(k)})$$

This model is able to cover the large structure of graph with high-order neighborhoods by applying multiple GCN layers.

D. Spectral Graph Wavelet Transform

Wavelet transform is a powerful multiresolution analysis tool that expresses a signal as a combination of several localized, shifted, and scaled bases (wavelet bases) [34]. Spectral graph wavelet transform refers to projecting graph signals from the vertex domain into the spectral domain using a proper set of bases provided by wavelet transform. This transformation provides valuable localization property by applying a series of appropriate scaling operations of graph signals [7].
As initially shown in [7], the spectral graph wavelet localized at vertex \( i \) with scale parameter \( s \) is shown by \( \psi_{s,i} \), whose \( j \)-th element is given by

\[
\psi_{s,i}(j) = \sum_{l=1}^{N} g(s\lambda_l)u_l^*(i)u_l(j)
\]

(6)

where \( N \) is the number of vertices, \( \lambda_l \) is the \( l \)-th eigenvalue of the normalized graph Laplacian matrix, \( u_l \) is the Laplacian's associated eigenvector that its \( j \)-th element \( u_l(j) \) is the value of the Laplace–Beltrami operator eigenfunction at vertex \( j \), the symbol \( * \) denotes complex conjugate operator, and \( g : \mathbb{R}^+ \rightarrow \mathbb{R}^+ \) is the spectral graph wavelet generating kernel.

Wavelet bases are defined by \( \Psi_s = [\psi_{1,s}, \ldots, \psi_{N,s}] \in \mathbb{R}^{N \times N} \), where each wavelet basis \( \psi_{s,i} \) corresponds to a signal on vertex \( i \) and scale \( s \). According to wavelet bases, (6) can be written in the form of matrix multiplication

\[
\Psi_s = \mathbf{UG_sU}^T
\]

(7)

where \( \mathbf{G} \) is the adjacency matrix, \( \mathbf{U} \) is the graph Fourier transform, and \( \Psi_s \) is a matrix of wavelet bases.

Computing the wavelet bases is still dependent on eigen-decomposition, which is inefficient for large graphs. As mentioned in [7], the graph wavelet bases can be approximated using Chebyshev polynomials as follows:

\[
\Psi_s = \frac{1}{2}e_{0,s} + \sum_{i=1}^{Q} e_{i,s} T_i(\tilde{\mathbf{L}})
\]

(8)

where \( T_i(\tilde{\mathbf{L}}) = 2\tilde{\mathbf{L}}T_{i-1}(\tilde{\mathbf{L}}) - T_{i-2}(\tilde{\mathbf{L}}) \) is the Chebyshev polynomial of order \( i \) for approximate \( \Psi_s \) with \( T_0(\tilde{\mathbf{L}}) = 1 \), \( T_1(\tilde{\mathbf{L}}) = \tilde{\mathbf{L}} \), \( Q \) is the number of Chebyshev polynomials, and \( J_i(-s) \) is the Bessel function of the first kind [36].

IV. Proposed Method

As mentioned in Section III, the computational complexity of graph Fourier transform for obtaining Fourier bases is the main drawback of the most spectral methods. In addition, graph Fourier transform, as a global transformation, does not provide helpful localization properties in the vertex domain.

Inspired by the superiority of spectral graph wavelet transforms in approximating with highly sparse wavelet bases that have more helpful localization properties, we developed a novel M-GWCN for analyzing multimodal data.

We observed that the sparse wavelet bases used in the proposed M-GWCN model provide more efficient computations than the models that are based on graph Fourier bases. Furthermore, since each wavelet basis is related to a signal on the graph that diffused away from a central node, the M-GWCN model with small-scale values is localized in the vertex domain of each modality. Thus, different scales of wavelet bases enable this model to represent feature vectors of each modality based on the different levels of localities in an efficient way.

The M-GWCN model also takes advantage of the complementary information provided by various modalities by learning cross-modal representations. Cross-modality representation is the process of representing feature vectors of each modality based on wavelet bases of the other modalities. Finding the correspondences among different vertices in various graphs is essential for cross-modality representation. The M-GWCN model explores these correspondences by learning permutation matrices, which encode them among various modalities.

The proposed M-GWCN method has three advantages that distinguish it from previous networks.

1) M-GWCN introduces a stacked architecture that utilizes graph wavelets convolution with multiple scaling parameters in parallel. Integration of embedded features, obtained using these different scales, provides more useful intramodality localization properties.

2) M-GWCN adopts residual connections, which makes the representation at each layer less prone to oversmoothing.

3) M-GWCN generalizes the benefit of graph wavelet transform for cross-modality representation by finding permutations encoded cross-modality correlations among various modalities.

A. Multiscale Adaptive Graph Wavelet

We define an adaptive graph wavelet (AGW) as a building block of the proposed network. AGW consists of a graph wavelet transformation with the desired scale. AGWs with different scales are concatenated in parallel and combined with a residual connection, as an adaptive component, to form a multiscale AGW (MAGW) or a stack of AGWs. Multiple scales of wavelets provide more helpful localization properties by decomposing a graph signal on components at different scales or frequency ranges.

The designed graph filter with MAGW approximates graph frequency response with different scales, without knowing the underlying graph structure. The residual connection tries to compensate for the limitation of MAGW in building deeper layers by making the representation at each layer less prone to oversmoothing. Fig. 1 shows a scheme of MAGW.

According to the superiority of rational filters in approximating various shapes of filters, compared with polynomial filters, we apply a rational filter, as a more versatile graph filter, for each AGW. Inspired by frequency response of rational filters mentioned in [37], the response of filtering signal \( x \) in scale \( s \) can be implemented using the following first-order recursion:

\[
x_s^{(i)} = \omega_x M x_s^{(i-1)} + \phi_x x_s^{(0)}
\]

(9)

where \( \omega_x \) and \( \phi_x \) are the filter coefficients in scale \( s \) and \( M \) is any practical graph representing matrix used to capture comprehensive information of the graph.

Inspired by this filter response in graph signal processing, we design a machine learning approach for learning the parameters \( \omega_x \) and \( \phi_x \) in each scale using a new GCN. In the
designed AGW, graph wavelet transform is used for localizing
graph convolution by projecting signals in the vertex domain
designed AGW, graph wavelet transform is used for localizing
k-space. In this phase, the cross-modality correlations on various
multimodal problems considered in the practical scenarios
is to find the pointwise correspondences or cross-modality
correlations among various modalities.

In this phase, the cross-modality correlations on various
modalities are explored and a new convolutional layer is
defined to represent embedded features of each modality based
on the graph wavelet of the other modalities.

To prevent the increase of learnable parameters, in this layer,
we leverage wavelet with one scale. Our GWCN learned in
phase 1 is permutation invariant because the embedded feature
vectors are insensitive to reordering the node index. We utilize
this property to take advantage of applying correlated repre-
sentational information of the other modalities discovered by
cross-modality correlations.

To have a better representation of embedded feature vectors
in modality m obtained after K layers, \( H^{(k)}_m \), based on the
wavelet bases of modality \( e \) \((e \neq m)\), we define the following
cross-modality feature mapping between two modalities \( m \) and \( e \):

\[
\tilde{H}_{m,e} = P_{m,e} \left( \Psi_{e,c} \Theta_{c,s} \Psi_{s}^{-1} \right) P_{m,e}^T H^{(K)}_m
\]

where \( P_{m,e} \) is an \( N_m \times N_e \) permutation matrix that encoded
cross-modality correspondence between modalities \( e \) and \( m \),
and \( \tilde{H}_{m,e} \) is a matrix of embedded feature vectors in
modality \( m \) obtained by representing \( H^{(K)}_m \) based on the
wavelet bases of modality \( e \), while their correlations are
encoded in \( P_{m,e} \).

A permutation matrix is defined as a matrix, including
exactly one single unit value in each row and column, and
zeros elsewhere. This matrix is used to represent the permuta-
tions of elements in an ordered sequence.

For example, for the following square matrix \( A \) with three
rows, a permutation in order of its rows as \((3,1,2)\) is

Applying stochastic dropout to the initial node feature
in (11) encourages each AGW to provide a response different
from the others.

The final embedded feature vectors of layer \( k \) are defined by
averaging the outputs of all \(| S | \) units of AGW in \( k \)th
AGW as

\[
H^{(k+1)}_m = \frac{1}{|S|} \sum_{s=1}^{|S|} \tilde{H}^{(k+1)}_{m,s}.
\]

2) Cross-Modality Correlations: The major challenge in
multimodal problems considered in the practical scenarios
is to find the pointwise correspondences or cross-modality
correlations among various modalities.

In this phase, the cross-modality correlations on various
modalities are explored and a new convolutional layer is
defined to represent embedded features of each modality based
on the graph wavelet of the other modalities.

To prevent the increase of learnable parameters, in this layer,
we leverage wavelet with one scale. Our GWCN learned in
phase 1 is permutation invariant because the embedded feature
vectors are insensitive to reordering the node index. We utilize
this property to take advantage of applying correlated repre-
sentational information of the other modalities discovered by
cross-modality correlations.

To have a better representation of embedded feature vectors
in modality m obtained after K layers, \( H^{(K)}_m \), based on the
wavelet bases of modality e \((e \neq m)\), we define the following
cross-modality feature mapping between two modalities \( m \) and \( e \):

\[
\tilde{H}_{m,e} = P_{m,e} \left( \Psi_{e,c} \Theta_{c,s} \Psi_{s}^{-1} \right) P_{m,e}^T H^{(K)}_m
\]

where \( P_{m,e} \) is an \( N_m \times N_e \) permutation matrix that encoded
cross-modality correspondence between modalities \( e \) and \( m \),
and \( \tilde{H}_{m,e} \) is a matrix of embedded feature vectors in
modality \( m \) obtained by representing \( H^{(K)}_m \) based on the
wavelet bases of modality \( e \), while their correlations are
encoded in \( P_{m,e} \).

A permutation matrix is defined as a matrix, including
exactly one single unit value in each row and column, and
zeros elsewhere. This matrix is used to represent the permuta-
tions of elements in an ordered sequence.

For example, for the following square matrix \( A \) with three
rows, a permutation in order of its rows as \((3,1,2)\) is

Applying stochastic dropout to the initial node feature
in (11) encourages each AGW to provide a response different
from the others.

The final embedded feature vectors of layer \( k \) are defined by
averaging the outputs of all \(| S | \) units of AGW in \( k \)th
AGW as

\[
H^{(k+1)}_m = \frac{1}{|S|} \sum_{s=1}^{|S|} \tilde{H}^{(k+1)}_{m,s}.
\]

2) Cross-Modality Correlations: The major challenge in
multimodal problems considered in the practical scenarios
is to find the pointwise correspondences or cross-modality
correlations among various modalities.

In this phase, the cross-modality correlations on various
modalities are explored and a new convolutional layer is
defined to represent embedded features of each modality based
on the graph wavelet of the other modalities.

To prevent the increase of learnable parameters, in this layer,
we leverage wavelet with one scale. Our GWCN learned in
phase 1 is permutation invariant because the embedded feature
vectors are insensitive to reordering the node index. We utilize
this property to take advantage of applying correlated repre-
sentational information of the other modalities discovered by
cross-modality correlations.

To have a better representation of embedded feature vectors
in modality m obtained after K layers, \( H^{(K)}_m \), based on the
wavelet bases of modality e \((e \neq m)\), we define the following
cross-modality feature mapping between two modalities \( m \) and \( e \):

\[
\tilde{H}_{m,e} = P_{m,e} \left( \Psi_{e,c} \Theta_{c,s} \Psi_{s}^{-1} \right) P_{m,e}^T H^{(K)}_m
\]

where \( P_{m,e} \) is an \( N_m \times N_e \) permutation matrix that encoded
cross-modality correspondence between modalities \( e \) and \( m \),
and \( \tilde{H}_{m,e} \) is a matrix of embedded feature vectors in
modality m obtained by representing \( H^{(K)}_m \) based on the
wavelet bases of modality e, while their correlations are
encoded in \( P_{m,e} \).

A permutation matrix is defined as a matrix, including
exactly one single unit value in each row and column, and
zeros elsewhere. This matrix is used to represent the permuta-
tions of elements in an ordered sequence.

For example, for the following square matrix \( A \) with three
rows, a permutation in order of its rows as \((3,1,2)\) is

Applying stochastic dropout to the initial node feature
in (11) encourages each AGW to provide a response different
from the others.

The final embedded feature vectors of layer \( k \) are defined by
averaging the outputs of all \(| S | \) units of AGW in \( k \)th
AGW as

\[
H^{(k+1)}_m = \frac{1}{|S|} \sum_{s=1}^{|S|} \tilde{H}^{(k+1)}_{m,s}.
\]
represented by the shown permutation matrix $\mathbf{P}$, which yields $\mathbf{A}_p = \mathbf{PA}$ by a simple matrix–vector multiplication as

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{A}_p = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \end{bmatrix}.$$

The permutation of symmetric matrix $\mathbf{A}$ is obtained by matrix–vector multiplication $\mathbf{A}_p = \mathbf{P}\mathbf{A}\mathbf{P}^T$.

Based on the cross-modality feature mapping, for each modality $m$, we conduct a cross-modality convolutional layer as follows:

$$\mathbf{H}^{(K+1)}_m = \sigma \left( \text{con}(\mathbf{H}^{(K)}_m, \hat{\mathbf{H}}_m) \mathbf{W}^{(K)}_m \right)$$

where $\mathbf{H}^{(K+1)}_m \in \mathbb{R}^{N_c \times f_{k+1}}$ is the output embedded feature matrix, $\mathbf{W}^{(K)}_m \in \mathbb{R}^{(M_f x) \times f_{k+1}}$ is the kernel matrix, $\hat{\mathbf{H}}_m = \text{con}(\hat{\mathbf{H}}_{m,1}, \ldots, \hat{\mathbf{H}}_{m,m-1}, \hat{\mathbf{H}}_{m,m+1}, \ldots, \hat{\mathbf{H}}_{m,M})$, $K$ is the total number of layers for intramodality representation, and con(·) function is the concatenation operator that incorporates the extracted feature information in the cross-modality convolution layer.

3) Node Classification: The embedded feature vectors of each modality are obtained through $K$ layers of feature mapping (intramodality representation) and one layer of feature mapping based on the other modalities (cross-modality representation).

The obtained embedded feature vectors are fed into the last layer to conduct node classification as follows:

$$\mathbf{Z}_m = \text{softmax}(\mathbf{H}^{(K+1)}_m \mathbf{W}^{(K+1)}_m)$$

where $\mathbf{W}^{(K+1)}_m \in \mathbb{R}^{f_{k+1} \times C}, \mathbf{Z}_m \in \mathbb{R}^{N_c \times C}$, and $C$ is number of classes.

4) Problem Formulation and Optimization: Considering the architecture of the proposed network, we present a unified regularization-based optimization problem that aims to simultaneously learn network parameters and permutations through network training.

Since the permutations are highly discrete and too costly to enumerate, the stochastic gradient descent (SGD) method is not capable to optimize the proposed networks because it applies for optimizing networks with continuous parameters.

We extend our formulation to the nearest convex surrogate by approximating all permutation matrices $\mathbf{P}_{m,e} (1 \leq m, e \leq M, m \neq e)$ in (13) with doubly stochastic matrices $\tilde{\mathbf{P}}_{m,e}$. A doubly stochastic matrix $\tilde{\mathbf{P}}_{m,e}$ is an $N_m \times N_e$ matrix of nonnegative real numbers, each of whose rows and columns sums to 1, i.e.,

$$\tilde{\mathbf{P}}_{m,e}(i, j) \geq 0, \quad \tilde{\mathbf{P}}_{m,e}1 = 1_e, \quad \tilde{\mathbf{P}}_{m,e}^T 1 = 1_e$$

where $1_e$ is an $N_e$-dimensional column vector of ones.

According to this relaxation, we propose a new network loss function as

$$\mathcal{L} = \mathcal{L}_{CE}(\mathcal{W}, \mathcal{P}) + \alpha \mathcal{L}_{DSM}(\mathcal{P})$$

where $\mathcal{W} = \{\mathbf{W}^{(k)}_1, \ldots, \mathbf{W}^{(k)}_M\}_{k=1}^K$ is the set of network weights, $\mathcal{P} = \{\tilde{\mathbf{P}}_{m,e} \mid 1 \leq m, e \leq M, m \neq e\}$ is the set of doubly stochastic matrices used in extension of (13), $\mathcal{L}_{CE}(\mathcal{W}, \mathcal{P})$ is the categorical cross-entropy loss function, which will be computed by (17), $\mathcal{L}_{DSM}(\mathcal{P})$ is the loss function for doubly stochastic matrix, which will be given in (19), and $\alpha$ is a tradeoff parameter between them.

The categorical cross-entropy loss function, $\mathcal{L}_{CE}(\mathcal{W}, \mathcal{P})$, measures the empirical loss on the training data by summing up the discrepancy between the outputs of the network $\mathbf{Z}_m$ and the ground truth $\mathbf{Y}_m$ for all modalities $m = 1, \ldots, M$

$$\mathcal{L}_{CE}(\mathcal{W}, \mathcal{P}) = \sum_{m=1}^{M} \sum_{i=1}^{l_m} \mathbf{y}_{m,i} \ln \mathbf{z}_{m}(i, :)$$

where $\mathbf{Z}_m$ is the output of the network, obtained in (15). This output is a function of the set of network weights $\mathcal{W}$ in (11), (14), and (15) and the set of doubly stochastic matrices $\mathcal{P}$ in the extension of (13).
The loss function for learning the doubly stochastic matrix \( \tilde{P}_{m,e} \) between two modalities \( m \) and \( e \) is given as follows:

\[
\mathcal{L}_{\text{DSM}}(\tilde{P}_{m,e}) = \frac{1}{N_m \times N_e} \left[ \sum_{i=1}^{N_m} \sum_{j=1}^{N_e} [ \tilde{p}_{m,e}(i, j) - 1 ] \right]
+ \sum_{j=1}^{N_e} \left( \sum_{i=1}^{N_m} [ \tilde{p}_{m,e}(i, j) - 1 ] \right).
\]  

(18)

Therefore, the loss function considering all doubly stochastic matrices is defined as follows:

\[
\mathcal{L}_{\text{DSM}}(\mathcal{P}) = \sum_{m=1}^{M} \sum_{e=1, e < m}^{M} \mathcal{L}_{\text{DSM}}(\tilde{P}_{m,e}).
\]  

(19)

The final optimization problem is also included the following additional regularization terms:

\[
\mathcal{R} = \beta \sum_{m=1}^{M} \sum_{k=1}^{K+1} \left| \mathcal{W}_m^{(k)} \right|_F^2 + \gamma \mathcal{R}_{BM}(\mathcal{P}) + \lambda \mathcal{R}_{WM}(\mathcal{W})
\]  

(20)

where \( \beta, \gamma, \) and \( \lambda \) are regularization parameters. The first term is the L2-norm regularization penalty on the network weights preventing overfitting by constraining the complexity of the learned kernels of all layers. The second term is the between-modality regularization term that restricts the output labels of all modality pairs \( m \) and \( e \) based on cross-modality relations encoded in the permutation matrix \( \tilde{P}_{m,e} \) as follows:

\[
\mathcal{R}_{BM}(\mathcal{P}) = \sum_{m=1}^{M} \sum_{e=1, e < m}^{M} \left( \left\| \mathcal{P}^{T}_{m,e} \mathbf{Z}_m - \mathbf{Z}_e \right\|_F^2 + \left\| \mathbf{Z}_m - \mathcal{P}_{m,e} \mathbf{Z}_e \right\|_F^2 \right).
\]  

(21)

The third term of (20) is the intramodality regularization term that is defined based on preserving manifold constraints of multimodal data based on the following equation:

\[
\mathcal{R}_{WM}(\mathcal{W}) = \sum_{m=1}^{M} \sum_{s=1}^{N_m} a_m(i, j) \left\| \mathbf{H}_m^{(K+1)}(i, :) - \mathbf{H}_m^{(K+1)}(j, :) \right\|^2
\]  

(22)

where \( a_m(i, j) \) is the weight of edge between the \( i \)th and \( j \)th vertices.

According to loss functions and regularization terms defined in (16) and (20), the final optimization problem on the objective function \( \mathcal{J} = \mathcal{L} + \mathcal{R} \) is obtained as follows:

\[
\mathcal{J} = \min_{\mathcal{W}, \mathcal{P}} \mathcal{L}_{\text{CE}}(\mathcal{W}, \mathcal{P}) + \alpha \sum_{m=1}^{M} \sum_{e=1, e < m}^{M} \mathcal{L}_{\text{DSM}}(\tilde{P}_{m,e})
+ \beta \sum_{m=1}^{M} \sum_{k=1}^{K+1} \left| \mathcal{W}_m^{(k)} \right|_F^2
+ \gamma \mathcal{R}_{BM}(\mathcal{P}) + \lambda \mathcal{R}_{WM}(\mathcal{W})
s.t. \quad \tilde{p}_{m,e}(i, j) > 0, \ 1 \leq m, e \leq M.
\]  

(23)

To optimize this problem under its constraints, an iterative optimization algorithm can be adopted to alternatively minimize the loss function with respect to \( \mathcal{W} \) and \( \mathcal{P} \).

In the first pass, \( \nabla \mathcal{W}_{m,e} \mathcal{J} \), that is the gradient of \( \mathcal{J} \) with respect to \( \mathcal{W}_m^{(k)} \), is computed, while all other parameters are considered fixed. According to the SGD optimization method, all network kernel matrices \( \mathbf{W}_m^{(k)}(m = 1, \ldots, M \text{ and } k = 1, \ldots, K + 1) \) can be updated using the following iterative equation until convergence

\[
\mathbf{W}_m^{(k)}(t + 1) = \mathbf{W}_m^{(k)}(t) - \eta \nabla \mathcal{W}_m^{(k)} \mathcal{J}(t + 1)
\]  

(24)

where \( \eta \) is the learning rate of the SGD method and \( t \) is the iteration number.

At the second pass, \( \nabla \mathcal{P}_{m,e} \mathcal{J} \) is computed as the gradient of \( \mathcal{J} \) with respect to \( \tilde{p}_{m,e}(1 \leq m, e \leq M, m \neq p) \), while all other parameters are fixed. Permutation matrices \( \tilde{P}_{m,e} \) can be updated using SGD as

\[
\tilde{P}_{m,e}(t + 1) = \tilde{P}_{m,e}(t) - \eta \nabla \mathcal{P}_{m,e} \mathcal{J}(t + 1)
\]  

(25)

while the nonnegative constraint is maintained by thresholding \( \tilde{P}_{m,e} = \max(\tilde{P}_{m,e}, 0) \).

Practically, for optimizing the abovementioned problems, we take advantage of Keras\(^1\) library by adding a new loss function and defining regularization terms on each layer, in which these penalties are summed into the loss function during optimization.

The proposed M-GWCN is summarized in Algorithm 1.

5) Computational Complexity Analysis: The computations of M-GWCN, as shown in Algorithm 1, mainly evolve two parts: computing graph wavelet bases (step 1) and computing network outputs (steps 2 and 3).

Since M-GWCN employs the Chebyshev polynomials for approximating graph wavelet bases, it takes its linearity advantage with computational complexity \( \mathcal{O}(E_m \mathcal{Q}) \), where \( E_m \) is the number of edges of \( \mathcal{G}_m \) and \( \mathcal{Q} \) is the order of the used Chebyshev polynomial.

For each specific scale \( s \) in modality \( m \), the computational complexity of each layer \( k \) in the intramodality localization phase of M-GWCN is computed based on two parts of matrix multiplications. The first part is the multiplication between the sparse square matrix \( \Psi_{m,s} \theta_{m,s} \Psi_{m,s}^{-1} \) and the embedded feature matrix \( \mathbf{H}_m^{(k)} \), which has incurred linear complexity, due to the sparsity structure of the wavelet bases. In the second part, there are two matrix multiplications between the embedded feature matrices and the kernel matrices \( \mathbf{W}_m^{(k)} \) and \( \mathbf{V}_m^{(k)} \). Therefore, the computational complexity in this phase is \( \mathcal{O}(N_m \mathbf{f}_k \mathbf{f}_{k+1} + N_m d_m \mathbf{f}_{k+1}) \), where \( d_m \) is the number of initial features. Inspired by [27] and by detaching the feature transformation from graph convolution, the computational complexity is reduced to \( \mathcal{O}(N_m + (f_k + d_m) f_{k+1}) \), which for all \( |\mathcal{S}| \) scales is equal to \( \mathcal{O}(|\mathcal{S}|N_m + |\mathcal{S}|(f_k + d_m) f_{k+1}) \). Accordingly, computing the feature mapping phase for each modality \( m \) requires \( \mathcal{O}(|\mathcal{S}|N_m + |\mathcal{S}|C_m) \), where \( C_m = \sum_{k=1}^{K+1} (f_k + d_m) f_{k+1} \). By generalizing to all \( M \) modalities, the computational complexity is \( \mathcal{O}(K|\mathcal{S}|N + |\mathcal{S}|B) \), where \( N = \sum_{m=1}^{M} N_m \) and \( B = \sum_{m=1}^{M} C_m \).

\(^1\)https://keras.io/api/losses/
**Algorithm 1 Multimodal GWCN (M-GWCN)**

**INPUTS**
1. $M$: number of modalities
2. $G_m = (V_m, E_m, A_m)$: Undirected weighted graph of modality $m$ ($1 \leq m \leq M$)
3. $Y_m = [y_{m,1}, \ldots, y_{m,N_m}]^T$: Label vector of data samples in modality $m$ ($1 \leq m \leq M$)

**HYPER-PARAMETERS**
1. $S$: Set of scales
2. $K$: Number of layers
3. $f_k$: Number of features in layer $k$
4. $r$: Dropout rate for initial node feature
5. $\sigma()$: Nonlinearity activation function
6. $Q$: Number of Chebyshev polynomials
7. $\alpha, \beta, \gamma, \lambda$: Regularization parameters
8. $T$: Maximum iteration number

**STEPS**
1. Approximate $\Psi_{m,s}$ for each modality $m$ in scale $s$ using equation (8).
2. Initialize $W_m^{(0)}(0)$ and $\hat{P}_{m,e}(0)$ randomly for all modalities $m$, $e = 1, \ldots, M$ and all layers $k = 1, \ldots, K + 1$.
3. For ($t = 1$ to $T$) do:
   a) Compute $\nabla W_m^{(t)}, \nabla J$ and update kernel matrix $W_m^{(t)}$ using equation (24).
   b) Compute $\nabla \hat{P}_{m,e}, \nabla J$ and update doubly stochastic matrices $\hat{P}_{m,e}$ using equation (25).
   c) Maintain non-negative constraint by thresholding $\hat{P}_{m,e} = \max(\hat{P}_{m,e}, 0)$.

**END for**

**OUTPUTS**
1. Network parameters $W_m^{(t)}(m = 1, \ldots, M$ and $k = 1, \ldots, K + 1).
2. Doubly stochastic matrices $\hat{P}_{m,e}$ ($1 \leq m, e \leq M$).

The computations of the cross-modality phase can perform efficiently by decoupling the permutation computation of sparse square matrix $\Psi_{s,m} \theta_{s,m} \Psi^{-1}_{s,m}$ from the permutation computation of input features $\Psi_{s,m}$. Since the computation of the doubly stochastic matrix $\hat{P}_{m,e}$ is also $O(N_e N_m)$, the computation of this phase for $M = 2$ modalities takes $O((2 + f_K)N_e N_m)$ operations. For $M > 2$ modalities, the overall computation complexity of cross-modality phase in the worst case is $O((2 + f_K)N_e N_m)$, where $N_e$ and $N_m$ are the number of data points in the first and second largest modalities, respectively, since computations are performed on pairs of modalities. In a similar way, the computational complexity of cross-modality convolutional layer and node classification phase is $O(N + M f_K f_{K+1})$ and $O(N + C f_{K+1})$, respectively.

Based on the above discussion, the overall complexity of the proposed framework is $O(E Q + (K |S|N + |S|B) + (N + M f_K f_{K+1}) + (N + C f_{K+1}) + ((2 + f_K)N_e N_m) + (\sum_{m=1}^{M} E_m)$, where $E = \sum_{m=1}^{M} E_m$. Since the number of modalities $M$, scales $|S|$, and layers $K$ are not necessarily small values, the overall complexity is dominated by the complexity of cross-modality phase, that is, $O(N_e N_m)$.

The importance of using sparse wavelet bases becomes more prominent when the graph size is increased. According to this computational complexity, by increasing the graph size, the complexity grows linearly.

To evaluate the computational performance of this model, we compare its computational complexity with the $M^2$CPC-u [6], as a state-of-the-art multimodal model in the practical scenario. Although the $M^2$CPC-u model efficiently find pointwise correspondence in $O(N_e |S| \log N_m)$, due to the computation of a dense matrix inverse in its label prediction phase in time $O(N^3)$, its overall complexity is not as efficient as M-GWCN model.

Among the other state-of-the-art multimodal methods, including LSJD [18], CD [16], and SCSMM [17], since they are still somewhat dependent on correspondences among modalities, they perform efficiently in the linear order of number of matching and mismatching samples. Nevertheless, the aforementioned comparison is not fair due to this dependency to prior knowledge.

To further evaluate the computational performance, we compare the computational complexity of GWCN model (Section IV-C1), as a unimodal version of M-GWCN, with state-of-the-art GNN models. For a graph $\hat{G} = (\hat{V}, \hat{E}, \hat{A})$ with $\hat{N} = |\hat{V}|$ nodes and $\hat{E} = |\hat{E}|$ edges, the computational complexity of GWCN is $O(\hat{E} Q + (K |S|\hat{N} + |S|C_m) + (\hat{N} + C f_{K+1}))$. The overall computational complexity of GWCN is dominated by the number of edges, $O(\hat{E})$, which is similar to the computational complexity of state-of-the-art GNN models, including GIN, GAT, ChebyNet, GCN, CayleyNet, GNN-ARMA, and GWNW, as reported in [2].

**C. Other Versions of M-GWCN**

1. **Graph Wavelet Convolutional Network**: Since most of the explicit graph-based data are unimodal, we simplify the proposed M-GWCN for unimodal tasks. GWCN, as a unimodal version of M-GWCN, is designed using the first $K$ layers integrated with the last classification layer, without cross-modality correlations.

2. **Multiview GWCN**: A special case of multimodal data, usually called multiview data, takes advantage of the prior knowledge about fully or partially correspondence information among modalities. We redesigned our proposed M-GWCN, called multiview GWCN (MV-GWCN), to cope with multiview problems. In MV-GWCN, permutation matrices encode correspondence information among various modalities, such that if sample $i$ in modality $m$, $x_{m,i}$, corresponds to sample $j$ in modality $e$, $x_{e,j}$, then $p_{m,e}(i, :)$ contains $1$ in the $j$th entry and $0$ otherwise.

MV-GWCN is trained in a similar way to M-GWCN, while permutation matrices are considered as nonlearning parameters.

**V. EXPERIMENTS**

We investigate the effectiveness of the proposed network with two types of experiments on unimodal explicit graph-based data and multimodal implicit graph-based ones.
The first experiment examines the efficiency of the proposed network on inherently graph-based data, including citation datasets, considering semisupervised node classification tasks. The purpose of the second experiment is to evaluate the effectiveness of the proposed method on multimodal data, implicitly considered as a graph, compared with state-of-the-art semisupervised multimodal problems. Public implementations with the open-source GNN libraries Spektral [38] (TensorFlow/Keras) are available in https://github.com/maysambehmanesh/M-GWCN.

### A. Evaluation on Unimodal Explicit Graph-Based Data

To evaluate the performance of M-GWCN on explicit graph-based data, we focus on semisupervised node classification of popular citation datasets. Since these datasets are unimodal, we apply the unimodal version of M-GWCN (GWCN) for semisupervised node classification.

Specifications of three benchmark datasets used in this experiment, including the number of nodes, edges, node features, and classes, are reported in Table I.

We compare the efficiency of GWCN with most popular state-of-the-art GCNs. These baselines include ChebyNet [25], GCN [33], CayleyNets [26], GNN-ARMA [28], and GWNN [27] as spectral methods, in addition to GAT [23], GraphSAGE [22], and GIN [21] as spatial methods.

In the semisupervised problem, the features of all vertices are known, but only 20 labels per class are given for training. Also, 500 and 1000 labeled nodes are considered for validation and test, respectively.

The task is learning a network that takes the feature vectors as inputs and assigns a label to each vertex as outputs. The evaluation results of the learned network on testing nodes are reported as classification accuracies.

Table II reports the experimental results of GWCN on citation datasets described in Table I and compares them with state-of-the-art methods. The accuracies of CayleyNets, GraphSAGE, and GIN have been reported from the respective papers, while others are obtained by running the authors codes with the hyperparameters given in their papers. In this table, GWCN-1 indicates the GWCN model with only one and the first scale, mentioned in Table III, e.g., 0.7 for Cora, and GWCN-2 considers both scales.

The maximum number of training epochs is set to 20000. Training of network will be terminated if the validation loss does not decrease for 50 consecutive epochs. The learning rate is $\eta = 0.01$ for all methods.

For maintaining the sparsity structure of graph wavelet bases, a threshold $t$ is defined to refine the values of $\Psi_{m,s}$ and $\Psi^{-1}_{m,s}$ such that the value of entries that are smaller than threshold $t$ is set to 0. This parameter in all experiments is set to $t = 10^{-4}$.

Using small values for scale parameters, graph wavelet bases induce locality properties in such a way that each basis represents the neighborhood structure of a specific vertex, as proved in [7]. To avoid model complexity in this experiment, we use two scales for graph wavelets, $|S| = 2$. The values of these scales are chosen among small values between 0 and 1 via grid search to ensure the locality of convolution in the vertex domain. The values of all parameters used in the above experiments are summarized in Table III. Parameters of other methods are chosen according to their respected papers.

According to classification accuracies reported in Table II, GWCN provides the best accuracies among all baselines. The second-best accuracies for Cora and Pubmed are provided by GWNN and for Citeseer are reported with GCN. These results demonstrate the capability of graph wavelet bases in achieving a better representation of vertex domain compared to other spectral methods.

Furthermore, GWCN performs better than all spatial methods, which reflects the promising ability of spectral methods to achieve good performance. Since GAT assigns a self-attention weight to each edge, it captures the local similarity among neighborhoods more effectively and provides better accuracy compared to other spatial methods. However, computing the attention weights is inefficient in the dense graphs.

Among spectral methods, GCN not only consistently outperforms others but also achieves a significant accuracy compared with GWCN. A key benefit of GCN compared to other spectral methods is that it leverages the Laplacian matrix as a weighted matrix in its formulation (see (5)), which in terms of sparsity, Laplacian matrix is sparser than Fourier bases and is similar to wavelet bases.

GWNN presents closer results to GWCN because it employs wavelet bases to localized graph convolution. Nevertheless, GWCN has two main advantages that enable it to consistently outperform GWNN. First, aggregating features with different values of scaling parameters makes GWCN flexible in locally exploring each subgraph according to an appropriate scale,
Instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.

Table V reports the training time and parameter complexity of GNN methods. Since in the citation network, Laplacian is sparser than wavelet bases [27], two graph wavelet-based methods, GWN and GWCN, have a little more time complexity compared with Laplacian-based methods, ChebyNet, GCN, and GNN-ARMA.

Most GNN models have a moderate number of parameters in their architecture because increasing the parameters by adding multiple layers will lead to the oversmoothing problem. Nevertheless, due to two main reasons, increasing the parameters of the GWCN model does not lead to oversmoothing: 1) instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.

Table V reports the training time and parameter complexity of GNN methods. Since in the citation network, Laplacian is sparser than wavelet bases [27], two graph wavelet-based methods, GWN and GWCN, have a little more time complexity compared with Laplacian-based methods, ChebyNet, GCN, and GNN-ARMA.

Most GNN models have a moderate number of parameters in their architecture because increasing the parameters by adding multiple layers will lead to the oversmoothing problem. Nevertheless, due to two main reasons, increasing the parameters of the GWCN model does not lead to oversmoothing: 1) instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.

Table V reports the training time and parameter complexity of GNN methods. Since in the citation network, Laplacian is sparser than wavelet bases [27], two graph wavelet-based methods, GWN and GWCN, have a little more time complexity compared with Laplacian-based methods, ChebyNet, GCN, and GNN-ARMA.

Most GNN models have a moderate number of parameters in their architecture because increasing the parameters by adding multiple layers will lead to the oversmoothing problem. Nevertheless, due to two main reasons, increasing the parameters of the GWCN model does not lead to oversmoothing: 1) instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.

Table V reports the training time and parameter complexity of GNN methods. Since in the citation network, Laplacian is sparser than wavelet bases [27], two graph wavelet-based methods, GWN and GWCN, have a little more time complexity compared with Laplacian-based methods, ChebyNet, GCN, and GNN-ARMA.

Most GNN models have a moderate number of parameters in their architecture because increasing the parameters by adding multiple layers will lead to the oversmoothing problem. Nevertheless, due to two main reasons, increasing the parameters of the GWCN model does not lead to oversmoothing: 1) instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.

Table V reports the training time and parameter complexity of GNN methods. Since in the citation network, Laplacian is sparser than wavelet bases [27], two graph wavelet-based methods, GWN and GWCN, have a little more time complexity compared with Laplacian-based methods, ChebyNet, GCN, and GNN-ARMA.

Most GNN models have a moderate number of parameters in their architecture because increasing the parameters by adding multiple layers will lead to the oversmoothing problem. Nevertheless, due to two main reasons, increasing the parameters of the GWCN model does not lead to oversmoothing: 1) instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.

Table V reports the training time and parameter complexity of GNN methods. Since in the citation network, Laplacian is sparser than wavelet bases [27], two graph wavelet-based methods, GWN and GWCN, have a little more time complexity compared with Laplacian-based methods, ChebyNet, GCN, and GNN-ARMA.

Most GNN models have a moderate number of parameters in their architecture because increasing the parameters by adding multiple layers will lead to the oversmoothing problem. Nevertheless, due to two main reasons, increasing the parameters of the GWCN model does not lead to oversmoothing: 1) instead of using only one scale parameter for the whole graph. Second, due to the sparsity of wavelet bases, after a few convolutions, the node features in GWN are tended to oversmoothing. Since the GWCN formulation is adopted in each layer with different scaling parameter values and not respected to more layers, 2) residual connection adopted for GWCN makes the representation less prone to oversmoothing.

To confirm the statistical significance of the accuracy improvement of our proposed GWCN models over state-of-the-art methods and prove that mean accuracy provides statistically significant results, we conduct t-test experiments using 95% confidence interval and 20 random data splits, which are highlighted in Table IV. The t-score for the compared models A and B is computed by $t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma^2_A}{n_A} + \frac{\sigma^2_B}{n_B}}}$, where $\mu$ and $\sigma^2$ are the mean and variance of the performances of each model, respectively, and $n$ is the number of runs of experiment. The t-score of the compared models for Cora, Citeseer, and Pubmed datasets is shown in Table IV.
maximum number of training epochs is 20000, and the training phase will be terminated if the validation loss does not decrease for 100 consecutive epochs. Experimental results over ten randomly data splits are repeated in terms of mean accuracy and standard deviations. Parameters of other methods are chosen according to their respective paper.

Table VIII reports the performance of M-GWCN in terms of mean and standard deviation of accuracies in comparison with the state-of-the-art results. Among these compared methods, M²CPC-u has the most similar experimental settings with M-GWCN because it takes prior correspondence knowledge. Although other methods have been developed for multimodal datasets, they are still somewhat dependent on prior knowledge about correspondences among modalities. In the respective papers of these methods, including CD (pos), CD (pos+neg), m-LSJD, and M²-LSJD, the rate of dependency is defined based on the ratio of the number of given corresponding samples to the total number of samples (in percent). To get closer to a fair comparison, we consider the correspondence ratio as minimum as possible (10%) in all mentioned methods.

Classification accuracies reported in Table VIII show that the M-GWCN model achieves a significant improvement and consistently outperforms other models. These results indicate the superiority of the proposed GCN in effectively finding the cross-modal correlations in the absence of given coupling/decoupling information between various modalities.

The M-GWCN is able to represent feature vectors of each modality based on the correlated wavelet bases on other modalities. This capability enables it to be not only successfully applicable to the multimodal graph-based data but also usable efficiently independent of initial correspondence information.

Since in many multimodal methods, the correspondences among modalities are predetermined, we evaluate the performance of our proposed network on the multiview datasets. Therefore, in the second experiment, we use MV-GWCN as a particular version of M-GWCN model and conduct it on multiview graph-based datasets, according to parameters listed in Table X. The other settings are similar to the first experiment.

To evaluate the performance of the proposed method on multiview datasets, we compare the efficiency of MV-GWCN with some existing methods evaluated on mentioned multiview datasets. The state-of-the-art methods used for comparison include MLDA [39], MLDA-m [39], MULDA [39], MULDA-m [39], MvMDA [40], OGMA [38], OMLDA [38], OMvMDA [38], and M²CPC-p [6].

To confirm the statistical significance of the accuracy improvement of our proposed M-GWCN model with respect to state-of-the-art techniques, we conduct t-test using 95% confidence interval and summarize the obtained results in Table IX. The highlighted entries in this table indicate passing the test with t-scores greater than or equal to 2.101. These results demonstrate that the mean accuracy always provides statistically significant results in this experiment.
TABLE IX
SUMMARY OF \( t \)-TEST OF MEAN ACCURACY USING 95% CONFIDENCE INTERVAL ON TWO MULTIMODAL DATASETS. EACH ENTRY IS A \( t \)-SCORE FOR TWO DIFFERENT MODELS AND HIGHLIGHTED ENTRIES INDICATE PASSING THE TEST.

| Datasets | Model      | CD (pos) | CD (pos-neg) | SCSSMM | m-LSD | m^2-LSD | M-CPC-u |
|----------|------------|----------|--------------|--------|--------|---------|---------|
| Caltech  | M-GWCN     | 37.01    | 60.81        | -      | 14.12  | 4.03    | 22.75   |
| NUS      | M-GWCN     | 20.68    | 31.47        | 6.58   | 12.43  | 4.65    | 10.36   |

Fig. 4. Visualizing the results using \( t \)-SNE on the MNIST dataset and comparing the quality of results of various GCNs in a multiview scenario.

Table X reports the parameters of MV-GWCN model.

| Parameter | Caltech 101-7 | Caltech 101-20 | MNIST |
|-----------|---------------|----------------|-------|
| \( K \)   | 2             | 2              | 2     |
| \( S \)   | \( \{0.5, 0.7\} \) | \( \{0.7, 0.8\} \) | \( \{0.3, 0.6\} \) |
| \( t \)   | \( 10^{-6} \) | \( 10^{-4} \) | \( 10^{-5} \) |
| \( Q \)   | 40            | 30             | 35    |
| \( r \)   | 0.75          | 0.75           | 0.75  |
| \( \eta \) | \( 10^{-2} \) | \( 10^{-2} \) | \( 10^{-2} \) |

Table XI reports the classification accuracy obtained by the M-GWCN compared with different multimodal problems on three multiview datasets. In this table, MV-GWCN-1 indicates the proposed method with only the first scale mentioned in Table X. Similarly, MV-GWCN-2 is based on two scales.

As can be seen from Table XI, MV-GWCN consistently outperforms other methods. According to these results, although our proposed method achieves state-of-the-art results in all datasets, the differences between their accuracies are not as significant as multimodal datasets. The main reason is that the knowledge of correspondence is predetermined in multiview datasets and corresponded samples can be fused into a joint latent space for boosting the classification, which improves accuracy without needing to explore the cross-modality correlations. When exploring the cross-modality correlations for discovering the correspondences among various modalities is a necessity, the superiority of our proposed method is proven to be more significant because of the lack of this type of prior knowledge.

To demonstrate that the mean accuracy provides statistically significant results in this experiment, we apply a \( t \)-test with 95% confidence interval for multiview datasets. The results that are obtained in a similar way to multimodal datasets are summarized in Table XII.

In the last experiment, we develop new network architectures based on several conventional GCNs for applying to multiview datasets. The architecture of these networks, which are named with a prefix “M” in Table XIII, is straightforward. Similar to MV-GWCN, each modality is first represented through \( K \) layers using a specific GCN, and then, embedded features in each modality are fused into a joint layer to have better representation based on all modalities.

This experiment evaluates the abilities of various GCNs in feature mapping of each modality and in cross-modal feature fusion simultaneously. Table XIII compares the performance of these networks with MV-GWCN, which takes advantage of graph wavelet bases abilities.

Fig. 4 visualizes the results of this experiment on the MNIST dataset, where we can see that the MV-MGWC-1 model achieves the best qualitative results when mapping using the \( t \)-SNE technique.

The classification results of various GCNs on multiview datasets are shown in Table XIII. The results reported in this table demonstrate the superiority of applying wavelet bases on GNNs with similar modality fusion ideas. These results confirm the effectiveness of applying wavelet bases in simultaneously representing each modality and utilizing correlation among all modalities efficiently.
VI. CONCLUSION

Extending the convolutional neural networks to multimodal geometrically structured and/or graph-based data is an important problem that has been rarely addressed to the best of authors’ knowledge. This article introduced a novel graph convolutional neural network based on wavelet bases to learn the representation of multimodal graph-based data in the spectral domain. Compared to Fourier bases used in spectral GNNs, wavelet bases represent the feature vectors in each modality utilizing scaling parameter more effectively. Besides, due to the ability of Chebyshev polynomials in approximating wavelet bases without requiring the costly eigendecomposition and also the sparse structure of obtained wavelet bases, computations are performed efficiently.

The proposed M-GWCN simultaneously provides intramodal localization by applying multiscaled graph wavelet convolution and estimates the cross-modal correlations between various modalities. We also introduced two additional particular versions of the proposed network for conducting unimodal and multiview tasks.

The proposed network evaluated on both unimodal explicit graph-based datasets as well as multimodal implicit graph-based data. Extensive experiments demonstrated that M-GWCN outperforms state-of-the-art GNNs, including unimodal and multimodal cases, without any prior knowledge.

To the best of our knowledge, our proposed M-GWCN model is the first work in developing GCNs for multimodal graph-based data defined in a fully practical scenario.

According to the efficiency, generality, and flexibility of the spatial methods, analyzing the multimodal graph-based data using a spatial network can be a valuable work for future. Since multimodal data have a wide range of applications, developing multimodal graph-based networks for other tasks, including cross-modal retrieval, multimodal clustering, and domain adaptation, can also be our other future works.

REFERENCES

[1] T. Baltrusaitis, C. Ahuja, and L.-P. Morency, “Multimodal machine learning: A survey and taxonomy,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 2, pp. 423–443, Feb. 2019.

[2] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, “A comprehensive survey on graph neural networks,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 1, pp. 4–24, Jan. 2021.

[3] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric deep learning: Going beyond Euclidean data,” IEEE Signal Process. Mag., vol. 34, no. 4, pp. 18–42, Jul. 2017.

[4] F. Scarselli, M. Gori, A. Chung Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” IEEE Trans. Neural Netw., vol. 20, no. 1, pp. 61–80, Jan. 2009.

[5] F. R. K. Chung, Lectures on Spectral Graph Theory. Providence, RI, USA: AMS, 2001.

[6] M. Behmanesh, P. Adibi, J. Chauussot, and S. M. S. Ehsani, “Cross-modal and multimodal data analysis based on functional mapping of spectral descriptors and manifold regularization,” 2021, arXiv:2105.05631.

[7] D. K. Hammond, P. Vandergheynst, and R. Gribonval, “Wavelets on graphs via spectral graph theory,” Appl. Comput. Harmon. Anal., vol. 30, no. 2, pp. 129–150, Mar. 2011.

[8] Y. Zhao et al., “Multi-view manifold learning with locality alignment,” Pattern Recognit., vol. 78, pp. 154–166, Jan. 2018.

[9] B. Liu, X. Chen, Y. Xiao, W. Li, L. Liu, and C. Liu, “An efficient dictionary-based multi-view learning method,” Inf. Sci., vol. 576, pp. 157–172, Oct. 2021.

[10] J. Li, Z. Li, G. Lu, Y. Xua, B. Zhang, and D. Zhang, “Asymmetric Gaussian process multi-view learning for visual classification,” Inf. Fusion, vol. 65, pp. 108–118, Jan. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S156625352030350X

[11] H. Q. Minh, L. Bazzani, and V. Murino, “A unifying framework in vector-valued reproducing kernel Hilbert spaces for manifold regularization and co-regularized multi-view learning,” J. Mach. Learn. Res., vol. 17, pp. 769–840, Jan. 2016.

[12] R. Zhou, X. Chang, L. Shi, Y-D. Shen, Y. Yang, and F. Nie, “Person reidentification via multi-feature fusion with adaptive graph learning,” IEEE Trans. Neural Netw. Learn. Syst., vol. 33, no. 5, pp. 1592–1601, May 2020.

[13] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, “Dynamic affinity graph construction for spectral clustering using multiple features,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 12, pp. 6323–6332, Dec. 2018.

[14] O. Katz, R. Talmon, Y-L. Lo, and H-T. Wu, “Alternating diffusion maps for multimodal data fusion,” Inf. Fusion, vol. 45, pp. 346–360, Jan. 2019.

[15] V. A. Zimmer, M. Á. González Ballester, and G. Piella, “Multimodal image registration using Laplacian commutators,” Inf. Fusion, vol. 49, pp. 130–145, Sep. 2019.

[16] D. Eynard, A. Kovnatsky, M. M. Bronstein, K. Glasshoff, and A. M. Bronstein, “Multimodal manifold analysis by simultaneous diagonalization of Laplacians,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 12, pp. 2505–2517, Dec. 2015.

[17] A. Pournemat, P. Adibi, and J. Chauussot, “Semisupervised charting and co-regularized multi-view learning,” J. Mach. Learn. Res., vol. 17, no. 1, pp. 2764–2804, 2016.

[18] M. Behmanesh, P. Adibi, J. Chauussot, C. Jutten, and S. M. S. Ehsani, “Geometric multimodal learning based on local signal expansion for joint diagonalization,” IEEE Trans. Signal Process., vol. 69, pp. 1271–1286, 2021.

[19] A. Micheli, “Neural network for graphs: A contextual constructive approach,” IEEE Trans. Neural Netw., vol. 20, no. 3, pp. 498–511, Mar. 2009.

[20] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, “Neural message passing for quantum chemistry,” in Proc. 34th Int. Conf. Mach. Learn. (ICML), vol. 3, 2017, pp. 1263–1272.

TABLE XII
SUMMARY OF T-TEST OF MEAN ACCURACY USING 95% CONFIDENCE INTERVAL ON THREE MULTIVIEW DATASETS. EACH ENTRY IS A T-Score FOR TWO DIFFERENT MODELS AND HIGHLIGHTED ENTRIES INDICATE PASSING THE TEST

| Datasets      | Model  | MULDA-M | M-OMDA | OOMA | OMLDA | OM-OMDA | M²CPC-p | MV-GWCN-1 | MV-GWCN-2 |
|---------------|--------|---------|--------|------|-------|---------|---------|-----------|-----------|
| Caltech101-7  | MV-GWCN-1  | 6.47    | 6.32   | 0.58 | 0.66  | 1.31    | 0.78    | 2.10      |          |
|               | MV-GWCN-2  | 16.64   | 16.17  | 5.51 | 5.69  | 6.87    | 4.49    | 2.10      | -         |
| Caltech101-20 | MV-GWCN-1  | 22.37   | 28.97  | 7.23 | 3.87  | 21.94   | 3.23    | 1.46      | -         |
|               | MV-GWCN-2  | 18.08   | 22.88  | 7.07 | 4.63  | 17.77   | 4.11    | 1.46      | -         |
| MNIST         | MV-GWCN-1  | 3.00    | 6.03   | 0.81 | 1.67  | 1.04    | -       | 1.54      | -         |
|               | MV-GWCN-2  | 9.44    | 15.49  | 5.06 | 6.78  | 5.51    | -       | 1.54      | -         |

TABLE XIII
CLASSIFICATION ACCURACIES ON MULTIVIEW DATASETS (MEAN ± STANDARD DEVIATION)

| Method       | Caltech101-7 | Caltech101-20 | MNIST |
|--------------|--------------|---------------|-------|
| M-CherryNet  | 91.49±0.3    | 85.53±0.4     | 95.50±0.8 |
| M-GAT        | 93.59±0.4    | 82.59±0.5     | 96.00±0.4 |
| M-GIN        | 92.51±0.7    | 84.27±0.3     | 95.25±0.6 |
| M-GCN        | 91.15±0.3    | 82.80±0.6     | 92.43±0.3 |
| M-NN-AARMA   | 94.89±1.1    | 86.37±0.5     | 95.75±0.3 |
| MV-GWCN-1    | 95.25±1.3    | 87.82±0.8     | 96.45±1.4 |
Maysam Behmanesh received the Ph.D. degree in artificial intelligence from the Faculty of Computer Engineering, University of Isfahan, Isfahan, Iran, in 2022.

He is currently a Post-Doctoral Researcher with the LIR Research Laboratory, GeoViC Group, École Polytechnique, Paris, France. He has been a Visiting Scholar with the GIPSA-lab, Grenoble Institute of Technology, Grenoble, France, from 2019 to 2020. His research is mainly focused on multimodal processing, machine learning, geometric analysis, and deep neural networks.