Multiscale Graph Scattering Transform

Genjia Liu
Shanghai Jiao Tong University
Shanghai, China
LGJ1zed@sjtu.edu.cn

Maosen Li
Shanghai Jiao Tong University
Shanghai, China
maosen_li@sjtu.edu.cn

Siheng Chen
Shanghai Jiao Tong University
Shanghai, China
sihengc@sjtu.edu.cn

***Abstract*—Graph scattering transform (GST) is mathematically-designed graph convolutional model that iteratively applies graph filter banks to achieve comprehensive feature extraction from graph signals. While GST performs excessive decomposition of graph signals in the graph spectral domain, it does not explicitly achieve multiresolution in the graph vertex domain, causing potential failure in handling graphs with hierarchical structures. To address the limitation, this work proposes novel multiscale graph scattering transform (MGST) to achieve hierarchical representations along both graph vertex and spectral domains. With recursive partitioning a graph structure, we yield multiple subgraphs at various scales and then perform scattering frequency decomposition on each subgraph. MGST finally obtains a series of representations and each of them corresponds to a specific graph vertex-spectral subband, achieving multiresolution along both graph vertex and spectral domains. In the experiments, we validate the superior empirical performances of MGST and visualize each graph vertex-spectral subband.**

***Index Terms*—graph scattering transform, graph multiresolution, vertex-spectral decomposition**

I. Introduction

Graph-structured data widely exists in real-world applications, and two frameworks have emerged to analyzing graph data: graph signal processing [30] and graph neural networks [5], [28], [32]. Graph signal processing (GSP) extends classical signal processing to irregular non-Euclidean domain and provides a mathematical framework to analyze graph data. Graph neural networks (GNN) adopt the data-driven manner and provide an effective framework to learn from graph data. Comparing these two, GSP emphasizes theoretical analysis; while GNN uses nonlinearity to promote better empirical performances.

To takes advantages of both GSP and GNN, graph scattering transform (GST) is proposed to serve as a powerful feature extractor and is also amenable to theoretical analyses. GST is a non-trainable graph convolutional model that iteratively applies graph filter banks followed by nonlinear activation functions [6], where filter banks are mathematically-designed and scatter the input signal to multiple frequency bands. The mathematically designed filter bank endows GST with stability to both graph topological deformation and graph signal perturbation [7], [8]. Furthermore, GST is competitive to well-trained GCN methods in various empirical tasks, especially with small training data sets [8].

Multiresolution is a common and powerful technique in both GSP and GNN [10], [15], [31], [33]. It enables comprehensive data analysis at multiple scales and precisely capture distinct patterns at one or more scales [10]–[14], especially for those graphs with hierarchical structures. For example, in citation network data sets [9], documents in the same research field are closely referenced and can be locally clustered. In Protein data set, local structure of molecules may correspond to different functional groups. In map data, structure of local vertexes set help with understanding the information transmission between closely adjacent cities [10]. Based on the GSP framework, Irion [10] constructed Haar-like wavelet by iterative partition on graph structure. Shuman et al. [15] presented a modular framework to generalize Laplacian pyramid transform for graph data, which could be apply for graph signal compression coding. Based on the GNN framework, Gao proposed Graph u-nets [33] which gathers the multiscale features via pooling through the importance of vertices. Li et al. proposed graph cross network [31] that achieve multiple scales feature learning based on novel vertex infomax pooling and feature-crossing layer. However, multiresolution techniques for GST has not been well studied yet, which could potentially provide deeper theoretical analysis than multiscale GNN as well as better empirical performance than multiscale GSP.

To achieve this goal, we propose multiscale graph scattering transform (MGST) framework. MGST essentially achieves multiresolution as well as multiscale along two domains: graph vertex domain via hierarchical partitioning and graph spectral domain via graph scattering. MGST obtains a series of representations and each of them corresponds to a specific vertex-spectral subband in a specific resolution layer. The spectral subband here indicates those frequency components selected by a filter, and a vertex subband indicates those vertices contained in a subgraph. The resulting representation thus provides rich possibilities to capture significant features in graph data. Our MGST framework will be introduced in (Section II). We also discuss the performance of MGST in the learning task of graph classification in (Section III), results show that MGST outperforms GST all data sets and is comparable to state-of-the-art GCN approaches.

II. Multiscale Graph Scattering Transform

We now introduce our multiscale graph scattering transform (MGST), producing rich possibilities to capture hierarchical representations along both graph vertex and spectral domains. We first construct a series of subgraphs with multiple graph scales and then perform scattering over each subgraph.

A. Multiscale graph decomposition via recursive partitioning

We construct subgraphs via recursive partitioning, and the subgraphs are expected to have the following properties. First, the each one of those subgraphs should be clustered, dividing the whole graph into several clusters helps us study the properties of each local part. Second, subgraphs with different scales should be included to bring us more sufficient structural information. Considering these assumptions, we perform recursive partitioning by solving RatioCut [34] problem for clusters $k = 2$, and a relaxed solution for RatioCut will be presented.

Given a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$, we first compute the complete eigenvalue and eigenvector pairs of laplacian matrix: $\{(\lambda_\ell, \mathbf{u}_\ell)\}_{\ell=0,1,\ldots,N-1}$. The number of zero eigenvalue depends on the number of components in the graph, and we denote the least positive eigenvalue as λ_{min} with its corresponding eigenvector as $\mathbf{u}_{min}$. Let the complete vertex set serves as the initial set $\mathcal{V}_{0,0}$, we partition it into two complementary subsets as $\mathcal{V}_{1,0}, \mathcal{V}_{1,1}$ based on the polarity of element in $\mathbf{u}_{min}$:

$$\mathcal{V}_{1,0} = \mathcal{V}^+ := \{v_i \in \mathcal{V} : \mathbf{u}_{min}(i) \geq 0\}, \quad (1a)$$

$$\mathcal{V}_{1,1} = \mathcal{V}^- := \{v_i \in \mathcal{V} : \mathbf{u}_{min}(i) < 0\}. \quad (1b)$$

In this way, we form the node sets of two parted subgraphs, whose structures are further inherited from the internal structures among the corresponding nodes in the input graph, thus obtaining the two disjoint subgraphs $\mathcal{G}^+ := \{\mathcal{V}^+, \mathcal{E}^+, \mathbf{W}^+\}$, $\mathcal{G}^- := \{\mathcal{V}^-, \mathcal{E}^-, \mathbf{W}^-\}$. Repeatedly partition on those generated graphs results in hierarchical subgraphs with decreasing scale, and each subgraph is obtained from a graph in upper layer.

This partition manner based on $\mathbf{u}_{min}$ is the relaxed solution of RatioCut [20] as well as unnormalized spectral clustering for clusters $k = 2$, it approximately solve the NP hard problem in a simple way. In fact, spectral clustering algorithms for clusters $k \geq 3$ worth considering as well, we perform two clusters partitioning here for calculation convenience.

B. Graph scattering on each scale

We then perform scattering transform on graphs to achieve layered frequency decomposition. Let $\mathbf{S}(\mathcal{G})$ and $\mathbf{x} \in \mathbb{R}^{N\times F}$ denote the shift matrix and graph signal with F channels respectively, where $\mathbf{S}(\mathcal{G})$ is typically chosen as laplacian matrix of graph structure. Frequency decomposition in each layer is realized by the linear operator described in (2):

$$\mathbf{H} : \mathbf{x} \mapsto (h_j(\mathbf{S})\mathbf{x})_{j=1,\ldots,J}. \quad (2)$$

each $h_j(\mathbf{S})$ denotes a filter in the form of polynomial of shift operator $\mathbf{S}$, and J denotes the size of filter bank. Typical choices of filter banks $\{\psi_j : h_j(\mathbf{S})\}_{j=1,\ldots,J}$ include diffusion wavelets [17] and tight Hann wavelets [19].

Graph scattering transform $\mathbf{\Phi}_\mathcal{G}(x)$ is then cascaded by wavelet decomposition operator $\mathbf{H}$, pointwise nonlinear function ρ and global average pooling operator U. The representation obtained from GST with L layers is in (3)

$$\mathbf{\Phi}_\mathcal{G}(\mathbf{x}) = \left\{U(\rho\mathbf{H})^k\mathbf{x}; k = 0, \ldots, L-1\right\}, \quad (3)$$

where $|\mathbf{\Phi}_\mathcal{G}(\mathbf{x})| = \sum_{\ell=0}^{L} J^\ell$. We denote $\mathbf{z}_{(0)} = \mathbf{x}$ and the transformation coefficient in the mth layer $U(\rho\mathbf{H})^m\mathbf{x}$ equals to $\left\{\mathbf{z}_{(j_1,\ldots,j_m)} : U\rho\psi_{j_m}\ldots\rho\psi_{j_1}\mathbf{x}\right\}_{1\leq j_1,\ldots,j_m\leq J}$. While the filter bank roughly partitions the spectrum into several subbands, the nonlinear function $\rho(\cdot)$ changes the distribution of spectrum and enables GST to be more discriminating.

C. Multiscale graph scattering

MGST finally performs graph scattering tranform on every subgraphs in every scale layer. Algorithm 1 describes MGST in detail, note that signal partition is realized by sampling from partitioned nodes.

The process of multiscale graph filtering performed by MGST is illustrated in Fig. 1. We define $\mathbf{\Psi}_{\mathcal{G},P}(\mathbf{x})$ as the multiscale graph scattering transform of graph signal $\mathbf{x}$. The first subscript of scattering coefficient vector $\mathbf{\Phi}_{i,j}$ represents the number of scale layer, and the scale of graph decreases with layer. The second number represents the sequence number in that layer, the lth layer contains 2^l graphs, and the volume of MGST is computed as $|\mathbf{\Psi}_{\mathcal{G},P}(\mathbf{x})| = (2^P - 1)\sum_{\ell=0}^{L} J^\ell$. Each $\mathbf{\Phi}_{i,j}$ is located on a specific subgraph $\mathcal{G}_{i,j}$, and each coefficient $(U\rho\psi_{j_m}\ldots\rho\psi_{j_1}\mathbf{x})$ contains energy extracted from a specific spectral subband.

The computational complexity of (2) is $\mathcal{O}(KE)$ [8] where K represents the polynomial order of the filter and E represents the number of edges in $\mathcal{G}$. The computational complexity for diagonalization of laplacian matrix before (1a)(1b) is $\mathcal{O}(N^3)$. Since the two calculation steps are decoupled, we express their computational complexity respectively.

Algorithm 1 Multiscale graph scattering transform $\mathbf{\Psi}$

Input: $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$, graph signal $\mathbf{x}$, size of filter bank J, layers for graph scattering transform L, layers for hierarchical graph partition $P \geq 2$.

Output: $\mathbf{\Psi}_{\mathcal{G},P}(\mathbf{x})$

Initialize: $\mathcal{G}_{0,0} := \mathcal{G}$, $\mathbf{x}_{0,0} := \mathbf{x}$, $\mathbf{\Phi}_{0,0} := \mathbf{\Phi}_{\mathcal{G}_{0,0}}(\mathbf{x}_{0,0})$

for $p = 1$ *to* $P - 1$ **do**
  for $j = 0$ *to* $2^{p-1} - 1$ **do**
    Graph partition: $\mathcal{G}_{p-1,j} \to \mathcal{G}^+_{p-1,j}, \mathcal{G}^-_{p-1,j}$
    $\mathcal{G}_{p,2j} := \mathcal{G}^+_{p-1,j}$, $\mathcal{G}_{p,2j+1} := \mathcal{G}^-_{p-1,j}$
    Signal partition: $\mathbf{x}_{p-1,j} \to \mathbf{x}^+_{p-1,j}, \mathbf{x}^-_{p-1,j}$
    $\mathbf{x}_{p,2j} := \mathbf{x}^+_{p-1,j}$, $\mathbf{x}_{p,2j+1} := \mathbf{x}^-_{p-1,j}$
    Scattering transform: $\mathbf{\Phi}_{p,2j} := \mathbf{\Phi}_{\mathcal{G}_{p,2j}}(\mathbf{x}_{p,2j})$, $\mathbf{\Phi}_{p,2j+1} := \mathbf{\Phi}_{\mathcal{G}_{p,2j+1}}(\mathbf{x}_{p,2j+1})$.
  end
end

return $\mathbf{\Psi}_{\mathcal{G},P}(\mathbf{x}) = \{\mathbf{\Phi}_{i,j}\}_{0\leq i<P,\ 0\leq j<2^i}$

D. Stability of MGST to graph data perturbations

In this section, we explore the stability of MGST when the input graph data $\mathbf{x} \in \mathbb{R}^N$ is disturbed. When analysing the stability of MGST, the frame bounds of the utilized wavelets is considered, that is $A^2\|\mathbf{x}\|^2 \leq \sum_{j=1}^{J} \|h_j(\mathbf{S})\mathbf{x}\|^2 \leq B^2\|\mathbf{x}\|^2$, where the frame bounds A and B describe how much the filter bank amplify frequency components [8].

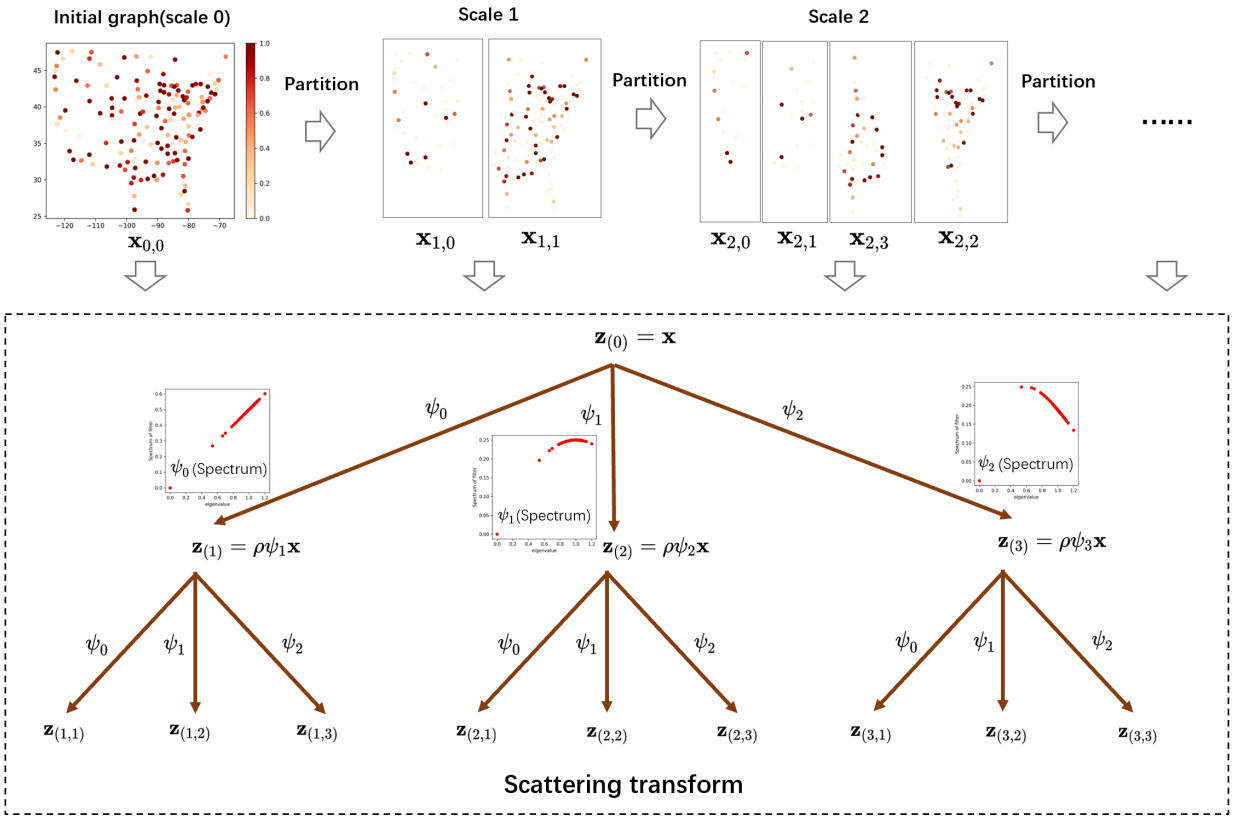

Fig. 1. We illustrate the process of scattering transform on a specific subgraph. $\mathbf{x}_{0,0}$ is graph signal with all pass frequency band and $\mathbf{x}_{3,4}$ is extracted from $\mathbf{x}_{0,0}$ by three times of partition, the color reflects the signal amplitude. ψ_0, ψ_1, ψ_2 denote three filters defined by diffusion wavelets, and their spectrums are shown upon. Iterative wavelet decomposition followed by non-linear function ρ constructs the scattering tree. This figure only shows the graph spectral analysis on one single subgraph, while MGST perform spectral decomposition on every subgraph that provides more comprehensive feature extraction.

We concern about how much is the output of model effected by the disturbance upon input signal, and we now show the stability of the proposed MGST.

Theorem 1: Consider MGST operator $\mathbf{\Psi}(\cdot)$ constructed on graph $\mathcal{G}$ with L layers, J filters and P scale levels, suppose that the graph filter bank forms a frame with bound B. Given signal $\mathbf{x} \in \mathbb{R}^N$ and disturbed signal $\tilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\delta} \in \mathbb{R}^N$, it then holds that

$$\frac{\|\mathbf{\Psi}_{\mathcal{G},\mathcal{P}}(\mathbf{x}) - \mathbf{\Psi}_{\mathcal{G},\mathcal{P}}(\tilde{\mathbf{x}})\|_2}{\sqrt{|\mathbf{\Psi}_{\mathcal{G},\mathcal{P}}(\mathbf{x})|}} \leq \sqrt{\frac{\sum_{\ell=0}^{L}(B^2J)^\ell}{\sum_{\ell=0}^{L}J^\ell}}\|\boldsymbol{\delta}\|_2. \quad (4)$$

The proof is similar to Theorem 1 in [8]. The numerator on the left of the inequality represents the square difference of the characteristics of MGST before and after disturbance, and $|\mathbf{\Psi}_{\mathcal{G},\mathcal{P}}(\mathbf{x})| = (2^P - 1)\sum_{\ell=0}^{L} J^\ell$ in denominator is the total number of scattering features. Notice that stability bound in (4) is not related to the number of scale levels P, and is linearly related to the square root of disturbance energy $\|\boldsymbol{\delta}\|$. While MGST performs recursive partitioning, the union of sub-vertex set in the pth ($p \geq 1$) scale layer equals to the complete vertex set: $\cup_{j=0}^{2^p-1}\mathcal{V}_{p,j} = \mathcal{V}_{0,0}$, therefore the disturbance signal $\boldsymbol{\delta}$ is partitioned in the same way that $\|\boldsymbol{\delta}\|_2^2 = \|\boldsymbol{\delta}_{0,0}\|_2^2 = \Sigma_{j=0}^{2^p-1}\|\boldsymbol{\delta}_{p,j}\|_2^2$.

III. EXPERIMENTAL RESULTS

This section validates the proposed MGST from both quantitative and qualitative ways.

A. Graph classification

The graph classification task requires to predict the category label y of the input graph $\mathcal{G}$ given the graph signal $\mathbf{x}$ and graph structure $\mathbf{S}(\mathcal{G})$. We use MGST as effective non-trainable representation extractor and utilize the extracted $\mathbf{\Psi}_{\mathcal{G},P}(x)$ for prediction task. The experiment of graph classification task was carried out on four benchmark data sets, including three proteins data sets: DD, Proteins, Enzymes, and one scientific collaboration data set Collab, the data sets are described in detail in Table I. Owing to the lack of graph signal feature $\mathbf{x}$ in data set DD and Collab, we thus construct node degree vector $\hat{\mathbf{x}} = \mathbf{W1}$ as fake signal. We employ random forest classifier to predict the label y according to the transformation coefficients $\mathbf{\Psi}_{\mathcal{G},P}(x)$ obtained from MGST. The hyperparameters of MGST and classifier are adjusted on the validation set. The graph scattering transform employs diffusion wavelets with $L = 5, J = 5$. We set the number of graph partition layers P no more than 5 and adjusted it on different data set. In some cases the graph contains only one node, we deal with this by dividing the graph into a empty set and itself.

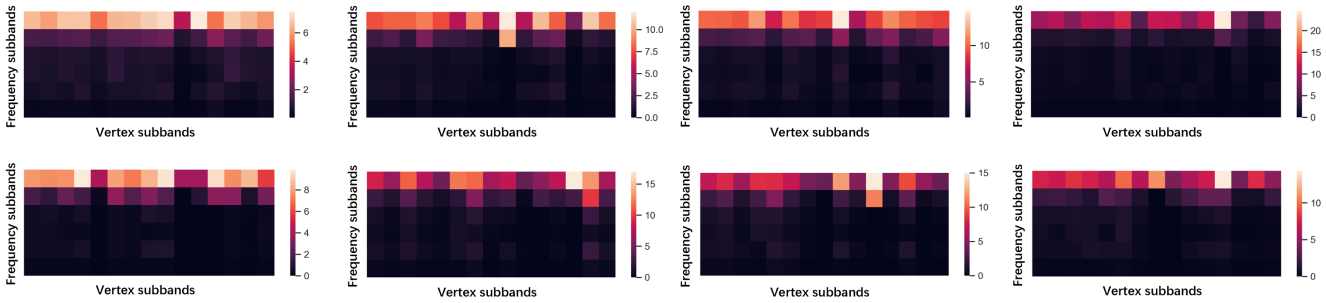

Fig. 2. Visualization of vertex-frequency subbands in eight graph data in Proteins dataset, where MGST is realized with $L = 3, J = 3, P = 4$. The vertex subbands correspond to subgraphs $\mathcal{G}_{0,0}, \{\{\mathcal{G}_{p,j}\}_{0\le j<2^p}\}_{0<p\le 3}$ from left to right. Each column located on a vertex subband corresponds to the first six scattering coefficients $\mathbf{z}_{(0)}, \mathbf{z}_{(1)}, \mathbf{z}_{(2)}, \mathbf{z}_{(3)}, \mathbf{z}_{(1,1)}, \mathbf{z}_{(1,2)}$ from up to down, the rest coefficients are omitted since they preserve little energy.

TABLE I
DATASET CHARACTERISTICS

Data set	Graphs	Avg. nodes	Avg. edges	Features
Proteins	1113	39	72	1
DD	1178	284	715	-
Collab	5000	74	2457	-
Enzymes	600	32	62	3

TABLE II
GRAPH CLASSIFICATION ACCURACY

	Method	Data set		
		Proteins	DD	Collab
Kernel	SHORTEST-PATH [21]	76.43	78.86	59.10
	WL-OA [22]	75.26	79.04	80.74
GNNs	PATCHYSAN [23]	75.00	76.27	72.60
	GRAPHSAGE [24]	70.48	75.42	68.25
	ECC [25]	72.65	74.10	67.79
	SET2SET [26]	74.29	78.12	71.75
	SORTPOOL [27]	75.54	79.37	73.76
	DIFFPOOL-DET [28]	75.62	75.47	82.13
	DIFFPOOL-NOLP [28]	77.42	79.98	75.63
	DIFFPOOL [28]	78.10	81.15	75.50
Scattering	GSC [29]	74.03	76.57	76.88
	GST [6]	76.65	74.20	76.48
	MGST(ours)	**78.85**	**79.13**	**77.49**

Table II lists the classification accuracy and we compare our MGST method with state-of-the-art approaches, including kernel methods [21], [22], deep graph network methods [23]–[28], scattering methods like geometric scattering classifier (GSC) [29] and base line graph scattering transform (GST) [6]. The bold numbers represent the best performance achieved by non-trainable scattering methods, and the numbers with gray background represent the best performance among all approaches. The results show that our MGST outperforms the previous non-trainable models on each data set. Even without learning process in the representation extraction phase, the performance of MGST is comparable to those deep learning GCN methods.

We also explored the impact of different hierarchical scales on different data sets and analyze the characteristics of graph distribution. Fig. 3 shows the variation of classification accuracy with scale layer P on each data set. In this experiment, we only select the original GST coefficient $\Phi_{0,0}$ and the coefficient of the Pth layer $\{\Phi_{P,j}\}_{0\le j<2^P}$ for classification, so the effects of different scale layers are considered separately.

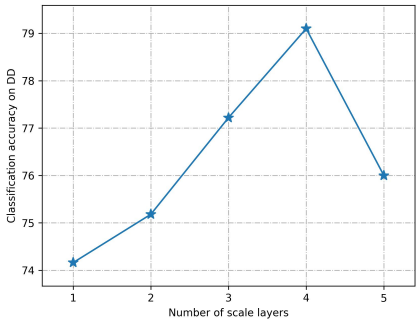

Fig. 3. Effect of scale layers. Adding scale layers enhance the discrimination ability, but this gain may decrease when the graph scale is too small.

The results show that adding some specific scale layers will significantly improve the accuracy, while other layers will not, which indicates that the structural characteristics of specific data sets are contained in specific scale levels.

B. Graph vertex-spectral subband visualization

We visualize the different distribution of vertex-spectral subbands in different graph data in Fig. 2, and we can figure out on which subbands the graph data mainly focuses. The spectral subband here indicates those frequency components selected by one or multiple filters, and a vertex subband indicates those vertices contained in a subgraph. The energy decreases with the depth of scattering from up to down due to the frequency decomposition. Those highlighted blocks indicate the main component of graph data.

We take the second figure in the first row as an example for analysis. The large value in the first line means that most of the energy of the signal is distributed in the corresponding frequency band of the first filter in $\mathbf{z}_{(1)}$. The large value in the ninth column means that the nodes in subgraph $\mathcal{G}_{3,1}$ have higher energy. We can locate the regions with larger signal values in this way. The result implies that the feature captured by MGST is located in both vertex and spectral domain, and different graph data highlights different subbands.

IV. Conclusions

This work proposes a novel multiscale graph scattering transform that achieve layered vertex-spectral decomposition. The multiscale graphs is obtained via recursive partitioning and the frequency decomposition is performed by scattering transform. Furthermore, when dealing with perturbed input signal, the stability of MGST is derived. Experiments demonstrate that MGST provides better empirical performance than other scattering approaches and is competitive to the state-of-the-art GCN methods.

References

[1] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in Proc. Intl. Conf. Mach. Learn. (ICML), 2016.

[2] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," IEEE Data Engineering Bulletin, 2017.

[3] M. Simonovsky and N. Komodakis, "Dynamic edge-conditioned filters in convolutional neural networks on graphs," in IEEE Conf. on Comp. Vis. and Pat. Rec., 2017.

[4] 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.

[5] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in Proc. Intl. Conf. on Learn. Representations (ICLR), Toulon, France, Apr. 2017.

[6] Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Diffusion scattering transforms on graphs. In Proc. Intl. Conf. on Learn. Representations (ICLR), 2019b.

[7] Fernando Gama, Joan Bruna, and Alejandro Ribeiro. Stability of graph scattering transforms. In Proc. Advances Neural Inf. Process. Syst. (NeurIPS), 2019a.

[8] Vassilis N. Ioannidis, Siheng Chen, and Georgios B. Giannakis. Pruned graph scattering transforms. In International Conference on Learning Representations (ICLR), Apr. 2020. 3, 4

[9] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. AI magazine, 29(3):93, 2008.

[10] Jeff Irion, Naoki Saito: Hierarchical graph Laplacian eigen transforms. JSIAM Lett. 6: 21-24 (2014)

[11] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in Proc. IEEE INFOCOM, vol. 3, Mar. 2003, pp. 1848–1857.

[12] R. R. Coifman and M. Maggioni, "Diffusion wavelets," Appl. Comput. Harmon. Anal., vol. 21, no. 1, pp. 53–94, Jul. 2006.

[13] D.I Shuman, S.K. Narang, P. Frossard, A. Ortega, P. Vandergheynst, The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains, IEEE Signal Process. Mag. 30 (2013) 83–98.

[14] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi-supervised learning," in Proc. Int. Conf. Mach. Learn., Haifa, Israel, Jun. 2010, pp. 367–374.

[15] David I. Shuman, Mohammad Javad Faraji, Pierre Vandergheynst: A Multiscale Pyramid Transform for Graph Signals. IEEE Trans. Signal Process. 64(8): 2119-2134 (2016)

[16] David I. Shuman, Benjamin Ricaud, Pierre Vandergheynst: Vertex-Frequency Analysis on Graphs. CoRR abs/1307.5708 (2013)

[17] Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Diffusion scattering transforms on graphs. In Proc. Intl. Conf. on Learn. Representations (ICLR), 2019b.

[18] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. Applied and Computational Harmonic Analysis, 30(2):129–150, 2011.

[19] David I Shuman, Christoph Wiesmeyr, Nicki Holighaus, and Pierre Vandergheynst. Spectrum-adapted tight graph wavelet and vertex-frequency frames. IEEE Trans. Sig. Process., 63(16):4223–4235, 2015.

[20] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput., 17 (2007), 395–416.

[21] Karsten M Borgwardt and Hans-Peter Kriegel. Shortest-path kernels on graphs. In IEEE Intl. Conf. on Data Mining (ICDM), 2005.

[22] Nils M Kriege, Pierre-Louis Giscard, and Richard Wilson. On valid optimal assignment kernels and applications to graph classification. In Proc. Advances Neural Inf. Process. Syst. (NeurIPS), 2016.

[23] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In Proc. Intl. Conf. Mach. Learn. (ICML), 2016.

[24] William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. IEEE Data Engineering Bulletin, 2017.

[25] Martin Simonovsky and Nikos Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In IEEE Conf. on Comp. Vis. and Pat. Rec., 2017.

[26] Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. Order matters: Sequence to sequence for sets. Proc. Intl. Conf. on Learn. Representations (ICLR), 2015.

[27] Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. An end-to-end deep learning architecture for graph classification. In AAAI Conference on Artificial Intelligence, 2018.

[28] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In Proc. Advances Neural Inf. Process. Syst. (NeurIPS), 2018.

[29] Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In Proc. Intl. Conf. Mach. Learn. (ICML), pp. 2122–2131, 2019.

[30] Antonio Ortega, Pascal Frossard, Jelena Kovacevic, José M. F. Moura, Pierre Vandergheynst: Graph Signal Processing: Overview, Challenges, and Applications. Proc. IEEE 106(5): 808-828 (2018)

[31] Maosen Li, Siheng Chen, Ya Zhang, Ivor W. Tsang: Graph Cross Networks with Vertex Infomax Pooling. NeurIPS 2020

[32] P. Veli ˇckovi ´c, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in Proc. Intl. Conf. on Learn. Representations (ICLR), 2018.

[33] Hongyang Gao and Shuiwang Ji. Graph u-nets. In Proceedings of the International Conference on Machine Learning (ICML), July 2019.

[34] Hagen, L., Kahng, A.: New spectral methods for ratio cut partitioning and clustering. IEEE Trans. Comput.-Aided Des. 11(9), 1074–1085 (1992)