Kumaraswamy Wavelet for Heterophilic Scene Graph Generation

Lianggangxu Chen[1], Youqi Song[1], Shaohui Lin[1], Changbo Wang[1*], Gaoqi He[1, 2*]

[1]School of Computer Science and Technology, East China Normal University, Shanghai, China
[2]Chongqing Key Laboratory of Precision Optics, Chongqing Institute of East China Normal University, Chongqing, China
{lgxchen, youqisong}@stu.ecnu.edu.cn, {shlin, cbwang, gqhe}@cs.ecnu.edu.cn

Abstract

Graph neural networks (GNNs) has demonstrated its capabilities in the field of scene graph generation (SGG) by updating node representations from neighboring nodes. Actually it can be viewed as a form of low-pass filter in the spatial domain, which smooths node feature representation and retains commonalities among nodes. However, spatial GNNs does not work well in the case of heterophilic SGG in which fine-grained predicates are always connected to a large number of coarse-grained predicates. Blind smoothing undermines the discriminative information of the fine-grained predicates, resulting in failure to predict them accurately. To address the heterophily, our key idea is to design tailored filters by wavelet transform from the spectral domain. First, we prove rigorously that when the heterophily on the scene graph increases, the spectral energy gradually shifts towards the high-frequency part. Inspired by this observation, we subsequently propose the Kumaraswamy Wavelet Graph Neural Network (KWGNN). KWGNN leverages complementary multi-group Kumaraswamy wavelets to cover all frequency bands. Finally, KWGNN adaptively generates band-pass filters and then integrates the filtering results to better accommodate varying levels of smoothness on the graph. Comprehensive experiments on the Visual Genome and Open Images datasets show that our method achieves state-of-the-art performance.

Introduction

Scene graph generation (SGG) aims to generate a structured representation of a scene by detecting objects and expressing their relationships through predicates in images (Cong, Yang, and Rosenhahn 2023). Therefore, SGG plays a big role in subsequent scene understanding tasks, including visual question answering (Lei et al. 2023) and image captioning (Yang, Liu, and Wang 2022).

Existing SGG methods typically use graph neural networks (GNNs) in the spatial domain (Li et al. 2018) to successively pass visual features for obtaining reliable context clues. Specifically, according to the trend of predicate nodes on a graph having consistent or different (homophily or heterophily) classes as their neighbors, GNNs used in SGG can be categorized into two classes : (1) ***Resampling based strategies designed for homophily***, as shown in Figure 1(c), to selectively aggregate neighborhood information (Tang et al. 2019; Yang et al. 2018; Zhou et al. 2022), and (2) ***Attention mechanism driven strategies designed for heterophily***, as shown in Figure 1(d), to correlate different neighbors (Li et al. 2021; Lin et al. 2020, 2022).

Unfortunately, existing spatial GNNs in SGG suffer from the over smoothing issue in the heterophilic scene graphs. Over smoothing in GNN will manifest as: the features of all nodes within the same connected component tend to be consistent. In general, GNNs update node representations by aggregating information from neighbors, which can be seen as a special form of low-pass filter (Bo et al. 2021). The low-pass filter in GNNs mainly retains the commonalities of node features. This mechanism works well for homogeneous graphs (Zhang et al. 2022), *i.e.*, similar nodes tend to connect with each other. However, in heterophilic scene graphs, nodes of different classes are often adjacent, which also means that fine-grained predicates are always connected to the coarse-grained predicates. Employing a low-pass filter to force the representation of connected predicates to be similar would prevent its applicability in heterogeneous scenes.

Specifically, the above over smoothing leads to inaccurate recognition of fine-grained predicates in the heterophilic scene graphs. For example, as shown in Figure 1(c) and Figure 1(d), existing variant models of GNNs (Tang et al. 2019; Li et al. 2021; Xu et al. 2017; Lin et al. 2022) misclassify fine-grained predicates. The fine-grained predicates **standing on/parked on/part of** are over smoothed to the coarse-grained predicate **on**. Figure 2(a) shows the misclassification ratio of fine-grained predicates by HL-Net (Lin et al. 2022). As shown in Figure 2(a), many fine-grained predicates have prediction error rates of up to about 10%.

To further support the previous analysis of over smoothing issue, scene graph features are decomposed into spectral features in Figure 2(b). We find one important phenomenon: when the heterophily on the graph increases, the spectral energy gradually shifts towards the high-frequency part. That is, frequency is positively associated with heterophily. Meanwhile, we observed that the features of different frequency bands contribute differently to the prediction accuracy of scene graphs with different heterophily ratios. This suggests that it is necessary to employ both high-pass filters to capture the differences between neighboring

*Changbo Wang and Gaoqi He are the corresponding authors.

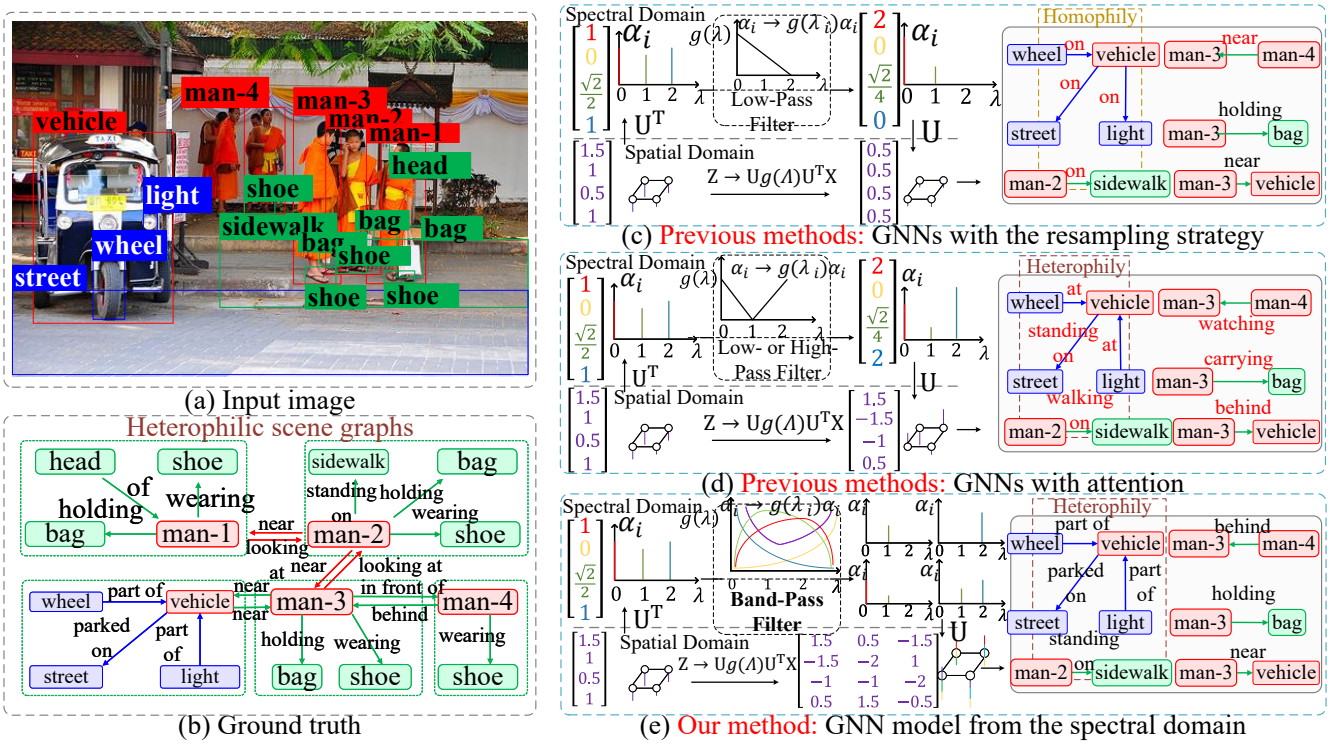


Figure 1: The illustration of handling fine-grained predicates for complex heterophilic scenarios. (a) An input image. (b) The ground truth of SGG. In (c), GNNs with resampling are prone to predict coarse predicates (Tang et al. 2019; Yang et al. 2018; Zhou et al. 2022). In (d), GNNs with attention prefer tail classes but still predicts fine-grained predicates incorrectly (Li et al. 2021; Lin et al. 2020, 2022). In (e), our KWGNN from the spectral domain can appropriately handle fine-grained predicates in heterophilic scenarios, e.g., standing on/walking on/parked on/part of. In (c), (d), and (e), homophily and heterophily of predicates are indicated in the yellow dashed box and the burgundy dashed box.

nodes and low-pass filters to maintain smoothness in the extracted signals (Bo et al. 2021). However, traditional filters (Lin et al. 2022; Xu et al. 2017) are typically designed for specific signals, which cannot effectively extract signals of different frequencies simultaneously.

In this paper, we propose Kumaraswamy Wavelet Graph Neural Network (KWGNN) to solve the problem of inaccurate fine-grained predicate recognition in heterophilic SGG. The key idea of KWGNN lies in better capturing the feature components of various frequencies in SGG. To achieve this, the Kumaraswamy distribution is used as the graph kernel function, benefiting from its excellent closed-form representation of probability density function controlled by learnable parameters. We rigorously prove the equivalence of Kumaraswamy wavelets in the spatial and spectral domains of graphs. This proof indicates that Kumaraswamy graph wavelets can be customized for any specific frequency band in SGG. Unlike from GNNs (Xu et al. 2017) that use a layer-by-layer message passing mechanism, KWGNN uses complementary multi-group wavelet kernels in parallel and then aggregates their filtering results for aggregating multi-band features. Finally, the updated graph features are used to predict class labels. Comprehensive experiments are conducted on the Visual Genome (VG) dataset. The results demonstrate the proposed method works well for fine-grained predicates in the heterophilic scene graphs.

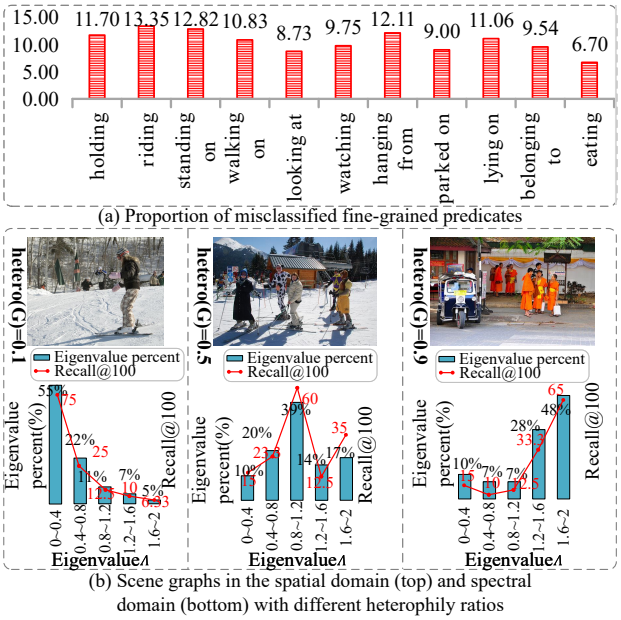
Overall, the main contributions are summarized as:

- To the best of our knowledge, we are the first to use spectral graph features for solving the over smoothing issue in SGG. We prove rigorously that frequency is strongly correlated with heterophily, which provides theoretical and empirical support for fine-grained predicates learning.
- A novel KWGNN is proposed to address heterophily via complementary multi-group band-pass filters. KWGNN explicitly establishes the connection between the heterophily in the spatial and spectral domains.
- Experimental results validate that our method can address the heterophily to effectively improve the prediction accuracy of fine-grained predicates. (*e.g.*, HL-Net, BGNN+HLB, and HetSGG improved by 111.2%, 51.98%, 44.13% of Mean Recall@100 on VG dataset).

Related Work

GNNs Based SGG Methods

The existing GNN-based SGG methods can be categorized into two classes. 1) Using the resampling strategy to ag-



(a) Proportion of misclassified fine-grained predicates

(b) Scene graphs in the spatial domain (top) and spectral domain (bottom) with different heterophily ratios

Figure 2: In (a), HL-Net (Lin et al. 2022) performs poorly in predicting fine-grained predicates in heterophilic scenarios. In (b), we illustrate the percentage of eigenvalues in different groups and how they contribute to the accuracy.

gregate neighborhood information selectively. Yang *et al.* (Yang et al. 2018) filtered out object pairs that may have relationships by pre-training correlation coefficient. However, they ignored the similarity of local structures. Zellers *et al.* (Zellers et al. 2018) generated structured scene representation from an image by learning prior motifs. Tang et al. (Tang et al. 2019) constructed a richer dynamic tree structure based on (Zellers et al. 2018), and then adopted bidirectional TreeLSTM to encode the visual contexts. 2) Applying attention mechanisms to correlate different neighbors. Tian *et al.* (Tian et al. 2020) and Zheng *et al.* (Zheng et al. 2020) refined local features via the attention-based intra-object message passing. However, they ignored multi-scale contextual information. Lin *et al.* (Lin et al. 2020) designed a direction-aware message passing module to enhance the node feature with multi-scale contextual information. Zareian *et al.* (Zareian, Karaman, and Chang 2020) bridged scene and commonsense graphs by connecting corresponding nodes, and then propagated messages using a Gated Recurrent Unit based attention mechanism. Tripathi *et al.* (Tripathi, Mishra, and Chakraborty 2023) and Wang *et al.* (Wang et al. 2023) designed the visio-lingual message passing method to transfer semantic knowledge into scene graphs.

Spectral GNNs

GNNs are developed to perform signal filtering based on the graph Laplacian eigendecomposition (Kipf and Welling 2016; Zheng, Pan, and Wu 2019). Balcilar *et al.* (Balcilar et al. 2021) pointed out that current GNNs can only obtain features of a small frequency segment. Bo *et al.* (Bo et al. 2021) further discovered and demonstrated the importance of high-frequency signals in graph classification tasks. To capture rich graph spectrum with large bandwidth, graph wavelet transform (GWT) and related models are developed (Wu et al. 2022). GWT generalizes the image-based scattering transforms (Gama, Ribeiro, and Bruna 2019), combining various graph signal filters with theoretically justified designs in terms of spectrum properties. Recently, researchers have studied to learn a filter that can approximate any shape by learning the parameters in the filter (He et al. 2021; Chien et al. 2020). One recent work HL-Net (Lin et al. 2022) theoretically showed traditional GNNs methods in SGG are essentially low pass filters.

Heterophilic SGG

In recent SGG research, He *et al.* (He et al. 2022) proposed a plug-and-play Heterogeneous Learning Branch to enhance the independent representation capability of predicate features. Yoon *et al.* (Yoon et al. 2022) proposed an unbiased heterogeneous SGG framework that captures relation-aware context using message passing neural networks. Lin *et al.* (Lin et al. 2022) proposed a novel Heterophily Learning Network to comprehensively explore the homophily and heterophily between objects/predicates. However, our work differs from theirs in: (1) our analysis are more comprehensive as we focus on all spectral graph features and investigate how distinct spectral graph features contribute to SGG rather than specific GCN designs; (2) we demonstrate our proposed KWGNN significantly outperforms (Lin et al. 2022) under fine-grained predicates setting in complex scenes.

Method

Problem Formulation and Definitions

Scene graph generation: Given an input image I, the task of SGG involves parsing I to obtain an unclassified scene graph $\mathcal{G}_s = \{\mathcal{V}_s = (\mathcal{V}_o \cup \mathcal{V}_p), \mathcal{E}_s\}$, where $\mathcal{V}_o = \{\mathbf{v}_o^i\}_{i=1}^{N_o}$, $\mathbf{v}_o^i$ denotes the feature of object node and $\mathcal{V}_p = \{\mathbf{v}_p^i\}_{i=1}^{M_p}$, $\mathbf{v}_p^i$ denotes the feature of predicate node. N_o is the number of objects and M_p represents the number of predicates between objects. There are two types of undirected edges in $\mathcal{E}_s$. Each edge connects a predicate node to its corresponding subject node or object node. During the prediction stage, each object node $\mathbf{v}_o^i$ is assigned a class label from the set of object classes C_o and a corresponding image location represented by a bounding box $\mathbf{b}^i$, where $\mathbf{b}^i = \left[b_x^i, b_y^i, b_w^i, b_n^i\right]$, such that b_x, b_y, b_w, b_h are the x coordinate, y coordinate, width and height of b^i, respectively. Each predicate node is assigned a class label from the set of predicate classes C_p. The resulting labeled graph is referred to as the **scene graph** $\mathcal{G}$.

Spectral GNNs based SGG: Let $\mathbf{A}$ be the adjacency matrix of $\mathcal{G}$, then graph laplacian $\mathbf{L}$ can be expressed as $\mathbf{D} - \mathbf{A}$ or as $\mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, where $\mathbf{I}$ is the identity matrix and $\mathbf{D}$ is the diagonal degree matrix (Opolka et al. 2022). $\mathbf{A}$ and $\mathbf{D}$ are pre-trained by VCTREE (Tang et al. 2019). Since $\mathbf{L}$ is positive semi-definite and symmetric, it has an eigendecomposition $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where $\mathbf{\Lambda} = \{\lambda_1, \cdots, \lambda_N\} \in [0, 2]$

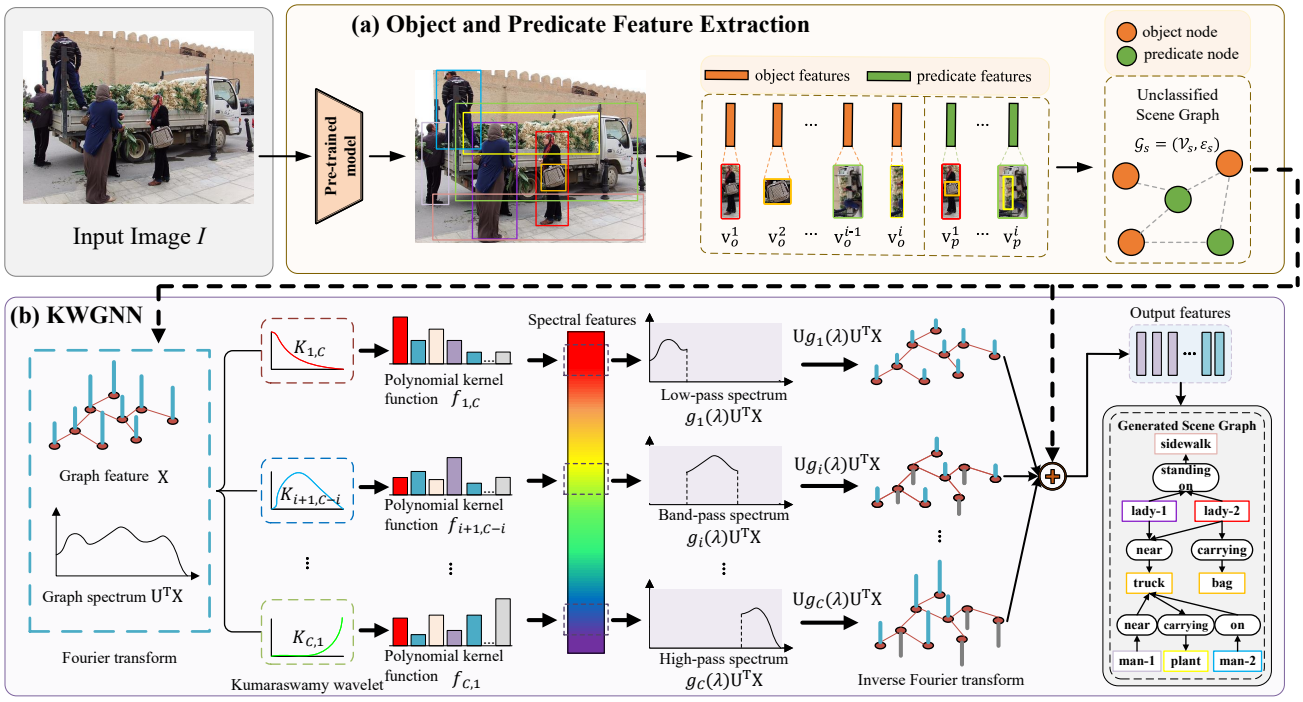


Figure 3: Architecture of the KWGNN. The KWGNN first applies Fourier transform to convert the scene graph features to the spectral domain. Then, graph wavelet transform are built for feature extraction at different frequencies. Finally, we build a skip-connection between input and output features and use inverse Fourier transform to generate the scene graph.

are eigenvalues ($N = N_o + M_p$) and $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_N]$ are unit eigenvectors (Luan et al. 2022). Next, the symbolic representations of the scene graph are simplified. Assuming $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]$ is a graph signal, then we call $\mathbf{U}^T\mathbf{X}$ as the spectral graph Fourier transform of signal $\mathbf{X}$. The objective of spectral GNN based SGG aims to identify a response function $g(\cdot)$ on $\mathbf{\Lambda}$ to learn node representations $\mathbf{Z} = \mathbf{U}g(\mathbf{\Lambda})\mathbf{U}^T\mathbf{X}$. $g(\mathbf{\Lambda})$ is called the filter, which implements the scaling of eigenvalues $\mathbf{\Lambda}$.

Definition 1. (Spectral density in a graph (Zhang et al. 2022)). Spectral density is a probability distribution that characterizes the power distribution of a graph signal $\mathbf{X}$ in the spectral domain. Given the graph Fourier transform results $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_N\}^T = \mathbf{U}^T\mathbf{X}$, the spectral density at a particular frequency λ_k is defined as:

$$S_k(\mathbf{X}, \mathbf{L}) = 1 - \frac{\sum_{i=1}^{k} \alpha_k^2}{\sum_{i=1}^{N} \alpha_i^2}, \tag{1}$$

where $\mathbf{L}$ is the Laplacian matrix of the graph. As $\alpha_k = \mathbf{u}_k^T\mathbf{X}$, a smaller S_k indicates that the spectral power concentrates more on the first k frequencies $\{\lambda_1, \lambda_2, \cdots, \lambda_k\}$.

Definition 2. (Heterophily (Lin et al. 2022)). Heterophily refers to the tendency of a node to have neighbors with different classes. The heterophily of a scene graph $\mathcal{G}$ can be defined as $\text{hetero}(\mathcal{G}) = \frac{1}{|\mathcal{V}_s|}\sum_{i\in\mathcal{V}_s}\frac{|\mathcal{N}_i^h|}{|\mathcal{N}_i|}$, where $\mathcal{N}_i$ denotes the set of neighboring nodes for the i-th node. $\mathcal{N}_i^h$ is the set of neighboring nodes with a different label than the i-th node, and $|\cdot|$ represents the cardinality operator. Hence, $\text{hetero}(\mathcal{G}) \to 0$ corresponds to strong homophily, while $\text{hetero}(\mathcal{G}) \to 1$ indicates strong heterophily. According to (Lin et al. 2022), heterophily in the scene graph dataset can be observed with $\text{hetero}(\mathcal{G})$ ranging between 0 and 1.

Proposition 1. In SGG, given Laplacian $\mathbf{L}$, and a scene graph signal $\mathbf{X}$, the expectation of spectral density $\mathbb{E}[S_k(\mathbf{X}, \mathbf{L})]$ is monotonically increasing with the heterophily degree of the scene graph:

$$\mathbb{E}[S_k(\mathbf{X}, \mathbf{L})] \propto \text{hetero}(\mathcal{G}). \tag{2}$$

This interesting finding verifies our conjecture. A low-pass filter is empirically used for a strong homophily scene graph, while a high-pass filter should be assigned if the scene graph is strong heterophily. Proposition 1 motivates us to capture features of different frequencies for scene graphs with the varying degrees of heterophily.

Kumaraswamy Wavelet Transform

Our overall framework is illustrated in Figure 3. We propose our Kumaraswamy Wavelet Transform and prove the equivalence in the spatial and spectral domains of graphs. Based on Kumaraswamy Wavelet Transform, the KWGNN is constructed. Based on proposition 1, we choose the Kumaraswamy distribution as the filter $g(\mathbf{\Lambda})$ to capture features of different frequencies. Here, our filter $g(\mathbf{\Lambda})$ is presented

in the form of graph wavelet transform. In this subsection, we first give the definition of graph wavelet transform and Kumaraswamy wavelet transform. Then we prove the consistency of Kumaraswamy wavelet transform in the spatial domain and the spectral domain and its advantages as band-pass filters in the spectral domain.

Definition 3. (Graph Wavelet Transform). The purpose of graph wavelet transform is to design a set of functions $\mathcal{F} = (f_1, f_2, \cdots, f_i)$ to achieve transformation control of eigenvalues $\mathbf{\Lambda}$. Defined in (Opolka et al. 2022), applying $\mathcal{F}$ on a graph signal $\mathbf{X}$ can be written as

$$\mathcal{F} = \mathbf{U} g(\mathbf{\Lambda}) \mathbf{U}^T \mathbf{X}, \tag{3}$$

where $g(\mathbf{\Lambda}) = \{g_1(\lambda), g_2(\lambda), \cdots, g_i(\lambda)\}$ is called kernel function. The kernel function $g(\cdot)$ needs to meet two requirements (Opolka et al. 2022):

- Kernel function is a real-valued function with special properties. It oscillates and decays rapidly, usually with an integral that satisfies the condition of being zero. It means the frequency spectrum $g(\lambda) = \int_{-\infty}^{\infty} g(t) e^{-i\omega t} d_t$ satisfies the Parseval theorem (Kelkar, Grigsby, and Langsner 1983):

$$\int_0^{\infty} \frac{|g_i(\lambda)|^2}{\lambda} d_\lambda = C_g, \tag{4}$$

  where C_g is a constant. $e^{-i\omega t}$ is a complex exponential function (Qin et al. 2021). The condition can be summarized as: kernel function should be waves that oscillate and decay rapidly.
- To avoid the eigen-decomposition of the graph Laplacian $\mathbf{L}$, the kernel function $g(\cdot)$ has to be a polynomial function, *i.e.*, $\mathbf{U} g(\Lambda) \mathbf{U}^T = g(\mathbf{L})$. Also, the kernel function can be formed by scaling and shifting the basic wavelet function in scale and position (Li et al. 2022):

$$g_{a,b}(\lambda) = \frac{1}{\sqrt{a}} g\left(\frac{\lambda - b}{a}\right), \tag{5}$$

  where a is the scaling coefficient and b is the translation coefficient. The kernel functions are completely determined by a and b, and the scaling transformation not only changes the width of the basic wavelet, but also its amplitude. The kernel function serves as a set of band-pass filters through different scaling functions $\{g_1(\lambda), g_2(\lambda), \cdots, g_i(\lambda)\}$ to cover different frequency bands $\{\lambda_1, \lambda_2, \ldots, \lambda_i\}$.

Definition 4. (Kumaraswamy Wavelet Transform). The Kumaraswamy distribution was originally proposed as a model for hydrological phenomena (Kumaraswamy 1980; Mitnik 2013). One of its main advantages is that it is a closed-form cumulative distribution function. Despite its potential, the Kumaraswamy distribution has not been extensively used in the context of mining spectral GNNs. In this work, we choose the Kumaraswamy distribution as the kernel function and demonstrate that it satisfies the requirements of graph wavelet transform. The standard form of the Kumaraswamy density function is given by:

$$K_{p,q}(t) = pq t^{p-1} \left(1 - t^p\right)^{q-1}, \tag{6}$$

where $t \in [0, 1]$ and $p, q > 0$ are shape parameters. To cover the full spectral range of the normalized graph Laplacian $\mathbf{L}$, we define the wavelet function $\mathcal{F} = \frac{1}{2} K_{p,q}(\frac{t}{2})$. To meet the requirements of graph wavelet transform, we impose the additional constraints $p, q \in \mathbb{N}^+$ to ensure that $K_{p,q}$ is a polynomial. Therefore, the Kumaraswamy wavelet transform can be expressed as:

$$\mathcal{F} = \mathbf{U} K_{p,q}(\mathbf{\Lambda}) \mathbf{U}^T = pq(\frac{\mathbf{L}}{2})^{p-1} \left(1 - (\frac{\mathbf{L}}{2})^p\right)^{q-1}. \tag{7}$$

The Kumaraswamy wavelet transform $\mathcal{F}$ is constructed by a group of $C = p + q$ Kumaraswamy wavelets with:

$$\mathcal{F} = \left(K_{1,C}, K_{2,C-1}, \cdots, K_{C,1}\right), \tag{8}$$

where C is a hyper-parameter. By the definition of the Fourier transform (Qin et al. 2021), the kernel function $K_{p,q}$ satisfies:

$$\int_0^{\infty} \frac{|g_i(\lambda)|^2}{\lambda} d_\lambda = \int_{-\infty}^{\infty} K_{p,q}(t) e^{-i\omega t} d_t = C_g, \tag{9}$$

where $C_g = \frac{(pq)^4}{4} \frac{(p!(q-1)!)^2}{(p+q-1)!} \left(1 - \frac{1}{2^{2p}}\right)^{2(q-1)}$ is a constant. p and q are two control factors. Therefore, the proposed Kumaraswamy wavelet transform $\mathcal{F}$ satisfies both two requirements of GWT in Definition 3.

Kumaraswamy Wavelet Graph Neural Network

Based on the introduced Kumaraswamy wavelet transform, we propose the KWGNN for SGG. Unlike from GNNs (Wang and Zhang 2022) that use a layer-by-layer message passing mechanism, complementary multi-group wavelet kernels in KWGNN are used and then aggregates the corresponding filtering results. Specifically, KWGNN adopts the following process to update node features:

$$\begin{aligned} \mathbf{Z}_i &= f_{i,C-i}(\phi_1(\mathbf{X})), \\ \mathbf{H} &= \phi_2\left([\phi_1(\mathbf{X}), \mathbf{Z}_0, \mathbf{Z}_1, \cdots, \mathbf{Z}_C]\right). \end{aligned} \tag{10}$$

We use three-layer fully connected networks with ReLU activation for trainable heads ϕ. $[\cdot]$ is a concatenation function. $f_{i,C-i}$ from Eqn. (8) denotes our wavelet kernels. We build a skip-connection between input $\mathbf{X}$ and output features $\mathbf{Z}$ to avoid overfitting. $\mathbf{H}$ is used to predict the label of the node in the unclassified scene graph.

Existing work on SGG tends to utilize cross-entropy loss for object and predicate classification (Xu et al. 2017; Zellers et al. 2018), which considers the priority of them to be all equal. Instead, the focal loss is used to handle this problem (Lin et al. 2017):

$$\mathcal{L} = \alpha(1 - p)^{\gamma} \log(p), \tag{11}$$

where p denotes the classification score calculated from $\mathbf{H}$. α and γ are the hyper-parameters.

Training Details

The Adam (Kingma and Ba 2014) optimizer is used with batch size 4 during training. The initial learning rate is 0.0001 for the backbone and 0.0003 for other parts, which

Model	SGDET		SGCLS		PREDCLS		
	mR@50	mR@100	mR@50	mR@100	mR@50	mR@100	Mean
HL-Net (Lin et al. 2022)	-	9.2	-	13.5	-	22.8	-
BGNN+HLB (He et al. 2022)	12.6	15.0	16.7	18.1	28.2	30.4	20.2
HetSGG (Yoon et al. 2022)	12.2	14.4	17.2	18.7	31.6	33.5	21.3
FGPL (Lyu et al. 2022)	17.4	20.3	22.6	24.0	36.4	40.3	26.8
SHA + GCL (Dong et al. 2022)	17.9	20.9	23.0	24.3	41.6	44.0	28.6
PE-Net (Zheng et al. 2023)	12.4	14.5	17.8	18.9	31.5	33.8	21.5
SQUAT (Jung et al. 2023)	14.1	16.5	17.5	18.8	30.9	33.4	21.9
TsCM (Sun et al. 2023)	18.3	21.2	23.7	25.1	40.1	42.3	28.4
A-PFG (Wang, Yuan, and Chen 2023)	18.9	21.7	24.5	25.8	41.9	44.3	29.5
Our method	**22.1**	**23.4**	**25.3**	**27.6**	**42.8**	**45.1**	**31.1**

Table 1: Comparison of mean recall on the three tasks of the VG dataset. Because some results are unavailable ('-'), the mean is only calculated from the complete results.

Model	OI V6-mR@50	Head	Body	Tail
GPS-Net	35.3	30.8	8.5	3.9
VCTREE-TDE	-	24.7	12.2	1.8
BGNN	40.5	34.0	12.9	6.0
RelTR	-	30.6	14.4	5.0
Our method	**44.4**	**37.8**	**17.6**	**9.8**

Table 2: Comparison with other state-of-the-art methods on the Open Images (OI) V6 (Kuznetsova et al. 2020) test set and the SGDET-mR@100 metric is utilized for evaluating the effectiveness of the head, body, and tail categories in VG.

$\mathrm{hetero}(\mathcal{G})$	0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.7-0.8	0.8-0.9	0.9-1
MOTIFS	37.7	19.2	35.3	36.9	37.4	36.7	40.3
Vtranse	33.0	21.4	36.7	36.1	37.5	35.8	39.5
HL-Net	33.8	29.3	29.6	37.0	40.4	40.1	43.8
HetSGG	34.9	30.3	31.2	38.0	40.2	40.0	41.3
Our method	**40.3**	**35.7**	**37.4**	**41.1**	**45.1**	**46.1**	**49.8**

Table 3: Performance comparisons on R@100 for different $\mathrm{hetero}(\mathcal{G})$ in the SGCLS task on the VG dataset.

Exp	Module	SGDET		
		R@20	R@50	R@100
1	KWGNN	**26.9**	**36.7**	**39.3**
2	GNN (Xu et al. 2017)	20.4	26.5	30.2
3	GAT (Li et al. 2021)	22.9	28.9	32.3
4	GPR-GNN (Lin et al. 2022)	26.7	32.5	35.0
5	Graph-rcnn (Yang et al. 2018)	26.7	36.3	39.2
6	VCTREE (Tang et al. 2019)	**26.9**	**36.7**	**39.3**
7	RelMN (Zhou et al. 2022)	26.5	36.2	39.1
8	C=1	23.3	29.5	33.1
9	C=2	26.8	32.9	35.1
10	C=3	**26.9**	**36.7**	**39.3**
11	C=4	26.7	32.9	37.1
12	C=5	25.9	31.9	35.1

Table 4: Ablation studies on the proposed KWGNN.

decays by a factor of 10 for every 5 epochs. The weight decay is set as 0.0001. The training of our full method takes approximately 19-20 hours. We set the dimension of node representations to 1024, and the filter number C in KWGNN is 3. The α and γ are empirically set to 0.1 and 2, respectively.

Experiments

Dataset and Evaluation Settings

We use the large-scale VG dataset (Krishna et al. 2017) to evaluate the effectiveness of the proposed KWGNN. The training set of VG includes 70% images, with 5K images used as a validation subset. The testing set is composed of the remaining 30% of the images. Three standard evaluation modes are used (Lu et al. 2016): (i) Scene Graph Detection (SGDET); (ii) Scene Graph Classification (SGCLS); (iii) Predicate Classification (PREDCLS). The image-wise Recall (R) evaluation metric is used, which measures the fraction of ground truth visual predicates appearing in top-20, top-50 and top-100 confident predictions. Owing to the scarcity of the tail class annotation in VG, previous studies usually achieve a low performance for less frequent classes. Hence, the Mean Recall (mR) is also utilized as an evaluation metric (Chen et al. 2019; Tang et al. 2019).

Comparisons with State-of-the-art Methods

In Table 1, our method performs the best with mR of 31.1 on average. To further analyze the model performance on fine-grained predicates prediction, we compute mR@100 for each predicates group on SGDET in Table 2 followed by RelTR(Cong, Yang, and Rosenhahn 2023). KWGNN achieves the highest mR@100 over all predicates classes and also performs SOTA in Open Images dataset with mR@50.

Moreover, to verify the effectiveness of our KWGNN for SGG in heterophilic scenarios, we compare our method with HetSGG (Yoon et al. 2022), HL-Net (Lin et al. 2022), Motifs (Zellers et al. 2018), and VtransE (Zhang et al. 2017) in Table 3. Following (Lin et al. 2022), we divide the degree of heterophily into 10 intervals based on $\mathrm{hetero}(\mathcal{G})$. As observed, KWGNN consistently achieves the best performance across all intervals. Specifically, KWGNN demonstrates significant superiority over other methods in scenarios with high heterophily (0.7-1). This notable performance

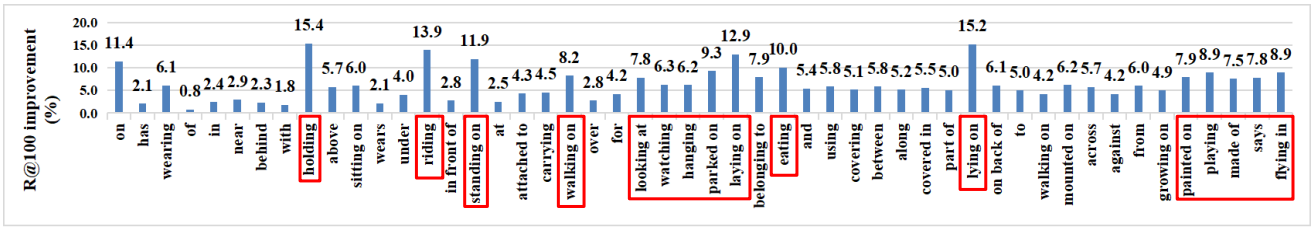


Figure 4: Improvement in PREDCLS of our full model for R@100 in comparison with HL-Net (Lin et al. 2022).

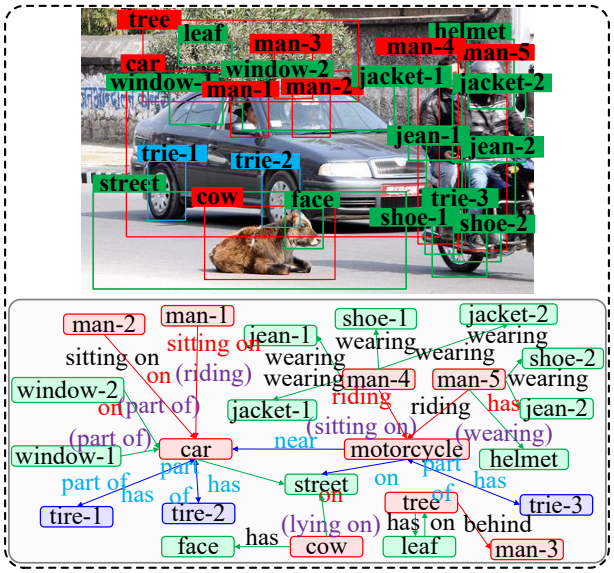


Figure 5: Qualitative examples of the improvement in SGG. The purple are those correctly classified predicates (in ground truth), the red are those incorrectly classified predicates in (Lin et al. 2022). Blue nodes and edges represent reasonable predictions but are not annotated as ground truth.

of KWGNN serves as compelling evidence of its distinct advantages to effectively capture the heterophily of SGG.

Ablation Studies

Effectiveness of KWGNN: According to (Balcilar et al. 2021), GNNs have equivalence between spectral domain and spatial domain. Common variations of spatial GNNs have their own spectral filtering expressions. As shown in Table 4, Exp 1 denotes our full method. Exp 2-4 represent the existing common GNNs with their spectral expressions. We see that a substantial performance gain after our polynomial band-pass filters have been applied.

The Impact of Initial Graph Structure: From Exp 5-7, we can see that different initialized graph structures have little impact on the prediction of scene graphs. Therefore, the choice of specific pre-trained methods, *e.g.*, Graph-rcnn or VCTREE is not the key factor in our framework. By default, we utilize VCTREE to extract the initial graph structure.

The Filter Number C***:*** The filter number C can determine the receptive field of the filters and whether multi-group filters can complement each other. Exp 8-12 in Table 4 present the Recall of KWGNN on the VG when varying C from 1 to 5. We observe that with a higher number C (1 to 3), the performance of our model increases. When further considering an even higher number C (3 to 5), the performance gradually significantly degrades. Although higher filter numbers allow the model to directly capture more information of neighborhoods, noisy messages start to permeate and hamper the prediction. From a frequency domain perspective, three filters are typically sufficient to cover low, intermediate, and high-frequency bands in the graph. Therefore, this suggests that $C = 3$ is the optimal choice.

Qualitative Results

Can our method really help improve the recall of fine-grained predicates? To demonstrate the effectiveness of our model in fine-grained predicates prediction, Figure 4 illustrates how our method affects the performance of each class. Our model predicts well on fine-grained predicates at the tail. Also, our method achieves a 15.4% improvement on **holding**, 13.9% on **riding**, 11.9% on **standing on** and 15.2% on **lying on**. This means that our results are beyond the simple reflection of the statistical bias of a long-tailed space, which achieved a more generalized performance.

In order to verify that our model really improves performance in heterophilic scenarios, Figure 5 shows the qualitative results. The performance of our model on those images is outstanding. Up to 24 objects can be detected in images. The results show that our model can recognize various objects and predicates in even complex scenes (*e.g.*, **cow-lying on-street**, **man-riding-car** and **man-wearing-helmet**). Moreover, from the blue nodes and edges, we can also find that our model can make some reasonable predictions that are missed in the annotation, such as **tire-part of-car**, **motorcycle-near-car** and **motorcycle-on-street**.

Conclusion

This work presents a novel analysis of SGG in the spectral perspective. We find that frequency is strongly associated with the heterophily. To this end, we propose Kumaraswamy Wavelet Graph Neural network to better capture heterophily on scene graphs. Our key idea is to design tailored filters from the spectral domain. KWGNN leverages complementary multi-group Kumaraswamy wavelets to generate band-pass filters and then aggregate the corresponding filtering results. Experimental results on VG dataset show the superiority and scalability of our model. KWGNN takes a step toward understanding the frequency in SGG.

Acknowledgments

This work was supported in part by Natural Science Foundation of Chongqing (No. CSTB2022NSCQ-MSX0552), National Natural Science Foundation of China (No. 62002121, 62102151 and 62072183), Shanghai Sailing Program (No. 21YF1411200), Shanghai Science and Technology Commission (No. 21511100700 and 22511104600), the Open Project Program of the State Key Lab of CAD&CG (No. A2203), and CCF-tencent Rhino-bird Open Research Fund.

References

Balcilar, M.; Guillaume, R.; Héroux, P.; Gaüzère, B.; Adam, S.; and Honeine, P. 2021. Analyzing the expressive power of graph neural networks in a spectral perspective. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Bo, D.; Wang, X.; Shi, C.; and Shen, H. 2021. Beyond low-frequency information in graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 3950–3957.

Chen, T.; Yu, W.; Chen, R.; and Lin, L. 2019. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6163–6171.

Chien, E.; Peng, J.; Li, P.; and Milenkovic, O. 2020. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*.

Cong, Y.; Yang, M. Y.; and Rosenhahn, B. 2023. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Dong, X.; Gan, T.; Song, X.; Wu, J.; Cheng, Y.; and Nie, L. 2022. Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19427–19436.

Gama, F.; Ribeiro, A.; and Bruna, J. 2019. Stability of graph scattering transforms. *Advances in Neural Information Processing Systems*, 32.

He, M.; Wei, Z.; Xu, H.; et al. 2021. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in Neural Information Processing Systems*, 34: 14239–14251.

He, Y.; Ren, T.; Tang, J.; and Wu, G. 2022. Heterogeneous Learning for Scene Graph Generation. In *Proceedings of the 30th ACM International Conference on Multimedia*, 4704–4713.

Jung, D.; Kim, S.; Kim, W. H.; and Cho, M. 2023. Devil's on the Edges: Selective Quad Attention for Scene Graph Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18664–18674.

Kelkar, S.; Grigsby, L.; and Langsner, J. 1983. An extension of Parseval's theorem and its use in calculating transient energy in the frequency domain. *IEEE Transactions on Industrial Electronics*, (1): 42–45.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1): 32–73.

Kumaraswamy, P. 1980. A generalized probability density function for double-bounded random processes. *Journal of hydrology*, 46(1-2): 79–88.

Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Kolesnikov, A.; et al. 2020. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7): 1956–1981.

Lei, S. W.; Gao, D.; Wu, J. Z.; Wang, Y.; Liu, W.; Zhang, M.; and Shou, M. Z. 2023. Symbolic replay: Scene graph as prompt for continual learning on vqa task. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 1250–1259.

Li, M.; Guo, X.; Wang, Y.; Wang, Y.; and Lin, Z. 2022. G^2CN: Graph Gaussian Convolution Networks with Concentrated Graph Filters. In *International Conference on Machine Learning*, 12782–12796. PMLR.

Li, R.; Sheng, W.; Zhu, F.; and Huang, J. 2018. Adaptive Graph Convolutional Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3546–3553.

Li, R.; Zhang, S.; Wan, B.; and He, X. 2021. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11109–11119.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Lin, X.; Ding, C.; Zeng, J.; and Tao, D. 2020. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3746–3753.

Lin, X.; Ding, C.; Zhan, Y.; Li, Z.; and Tao, D. 2022. HL-Net: Heterophily learning network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19476–19485.

Lu, C.; Krishna, R.; Bernstein, M.; and Fei-Fei, L. 2016. Visual relationship detection with language priors. In *European conference on computer vision*, 852–869. Springer.

Luan, S.; Hua, C.; Lu, Q.; Zhu, J.; Zhao, M.; Zhang, S.; Chang, X.-W.; and Precup, D. 2022. Revisiting heterophily for graph neural networks. *arXiv preprint arXiv:2210.07606*.

Lyu, X.; Gao, L.; Guo, Y.; Zhao, Z.; Huang, H.; Shen, H. T.; and Song, J. 2022. Fine-grained predicates learning for scene graph generation. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, 19467–19475.

Mitnik, P. A. 2013. New properties of the Kumaraswamy distribution. *Communications in Statistics-Theory and Methods*, 42(5): 741–755.

Opolka, F.; Zhi, Y.-C.; Lio, P.; and Dong, X. 2022. Adaptive gaussian processes on graphs via spectral graph wavelets. In *International Conference on Artificial Intelligence and Statistics*, 4818–4834. PMLR.

Qin, Z.; Zhang, P.; Wu, F.; and Li, X. 2021. Fcanet: Frequency channel attention networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, 783–792.

Sun, S.; Zhi, S.; Liao, Q.; Heikkilä, J.; and Liu, L. 2023. Unbiased Scene Graph Generation via Two-stage Causal Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Tang, K.; Zhang, H.; Wu, B.; Luo, W.; and Liu, W. 2019. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6619–6628.

Tian, H.; Xu, N.; Liu, A.-A.; and Zhang, Y. 2020. Part-Aware Interactive Learning for Scene Graph Generation. In *Proceedings of the 28th ACM International Conference on Multimedia*, 3155–3163.

Tripathi, A.; Mishra, A.; and Chakraborty, A. 2023. Grounding Scene Graphs on Natural Images via Visio-Lingual Message Passing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 4391–4400.

Wang, L.; Yuan, Z.; and Chen, B. 2023. Learning to Generate an Unbiased Scene Graph by Using Attribute-Guided Predicate Features. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 2581–2589.

Wang, X.; and Zhang, M. 2022. How powerful are spectral graph neural networks. In *International Conference on Machine Learning*, 23341–23362. PMLR.

Wang, Z.; Cheng, B.; Zhao, L.; Xu, D.; Tang, Y.; and Sheng, L. 2023. VL-SAT: Visual-Linguistic Semantics Assisted Training for 3D Semantic Scene Graph Prediction in Point Cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21560–21569.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2022. Beyond low-pass filtering: Graph convolutional networks with automatic filtering. *IEEE Transactions on Knowledge and Data Engineering*.

Xu, D.; Zhu, Y.; Choy, C. B.; and Fei-Fei, L. 2017. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5410–5419.

Yang, J.; Lu, J.; Lee, S.; Batra, D.; and Parikh, D. 2018. Graph r-cnn for scene graph generation. In *Proceedings of the European conference on computer vision (ECCV)*, 670–685.

Yang, X.; Liu, Y.; and Wang, X. 2022. Reformer: The relational transformer for image captioning. In *Proceedings of the 30th ACM International Conference on Multimedia*, 5398–5406.

Yoon, K.; Kim, K.; Moon, J.; and Park, C. 2022. Unbiased Heterogeneous Scene Graph Generation with Relation-aware Message Passing Neural Network. *arXiv preprint arXiv:2212.00443*.

Zareian, A.; Karaman, S.; and Chang, S.-F. 2020. Bridging knowledge graphs to generate scene graphs. In *European Conference on Computer Vision*, 606–623. Springer.

Zellers, R.; Yatskar, M.; Thomson, S.; and Choi, Y. 2018. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5831–5840.

Zhang, H.; Kyaw, Z.; Chang, S.-F.; and Chua, T.-S. 2017. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5532–5540.

Zhang, Y.; Zhu, H.; Song, Z.; Koniusz, P.; and King, I. 2022. Spectral Feature Augmentation for Graph Contrastive Learning and Beyond. *arXiv preprint arXiv:2212.01026*.

Zheng, C.; Lyu, X.; Gao, L.; Dai, B.; and Song, J. 2023. Prototype-based Embedding Network for Scene Graph Generation. *arXiv preprint arXiv:2303.07096*.

Zheng, C.; Pan, L.; and Wu, P. 2019. Multimodal deep network embedding with integrated structure and attribute information. *IEEE transactions on neural networks and learning systems*, 31(5): 1437–1449.

Zheng, Z.; Li, Z.; An, G.; and Feng, S. 2020. Subgraph and object context-masked network for scene graph generation. *IET Computer Vision*, 14(7): 546–553.

Zhou, H.; Yang, Y.; Luo, T.; Zhang, J.; and Li, S. 2022. A unified deep sparse graph attention network for scene graph generation. *Pattern Recognition*, 123: 108367.