PKU-IAI Technical Report: PKU-IAI-2022-T-0001

Review, Replicate, Reflect: A Comprehensive Study of Graph Parsing Neural Network

Tang Jiaqi
Yuanpei College
Peking University
2200017803@stu.pku.edu.cn

Huang Chenyi
Yuanpei College
Peking University
2200017850@stu.pku.edu.cn

Tang Yu
The School of Electronics Engineering and Computer Science
Peking University
2200013149@stu.pku.edu.cn

Ye Mao
Yuanpei College
Peking University
2200017852@stu.pku.edu.cn

Wang Keran
Yuanpei College
Peking University
2100017727@stu.pku.edu.cn

Abstract

Human-object interaction (HOI) detection is a crucial task in visual perception and physical scene understanding that involves identifying and interpreting interactions between humans and objects from visual information. In 2018, Qi et al. [27] proposed Graph Parsing Neural Network (GPNN) for detecting and recognizing HOI in images and videos. For an given scene, GPNN iteratively infers a parse graph including the HOI graph structure's adjacent matrix and the node labels. In our project, we review the key principles and methods of GPNN, replicate its model architecture and training, and evaluate its performance on benchmark datasets used in [27]: HICO-DET, VCOCO, and CAD-120. By comprehensively comparing GPNN with other models, including previous approaches and the more updated methods, we reflect on the strengths and limitations of GPNN and other methods. Our code can be found at https://github.com/slavic-codeman/Project-of-GPNN-Replicate.

1 Introduction

Human-object interaction (HOI) task is targeted at interpreting and predicting relationships between humans and objects in various physical scenarios. Beyond visual recognition, it requires a deeper understanding of the semantic connections and relations in visual contents. Before GPNN was proposed, deep neural networks had already been widely applied in computer vision tasks including classification, detection, and segmentation, yet only few approaches [3, 2, 13, 31] had been proposed for HOI task, mainly because that HOI is a comprehensive task that requires not only accurate recognition of various visual entities, but also precise and reasonable understanding of their relationships.

Fig. 1 shows the structure of GPNN and the HOI detection in images. For videos, GPNN can also iteratively interpret the optimal structure and node labels, as it can detect HOI in each frame and then update its hidden state. This allows GPNN to capture sequence information in videos. In fact, GPNN is a general framework adopted from Message Passing Neural Network (MPNN) [11], instead of a fixed model, and the network architecture would vary to handle different representations of information, like images and videos.

The essence of GPNN is Graph Neural Network (GNNs). GNNs can model a wide variety of relationships. It is mainly because that graphs can explicitly represent spatial and temporal dependencies, integrate and broadcast the information from different elements in a given scene. Previous deep neural network models [8, 18, 20, 34] rely on pre-fixed graph structures for HOI detection. However, interactions and relationships are not certain in real scenarios, thus not always compatible with a stationary graph structure. Therefore, GPNN introduces a *link function* to make the graph structure learnable. Meanwhile, the combination with neural network makes the model fully differentiable, which means it can be optimized in an end-to-end manner, without the requirement for any prior knowledge, enhancing the model's generalization ability.

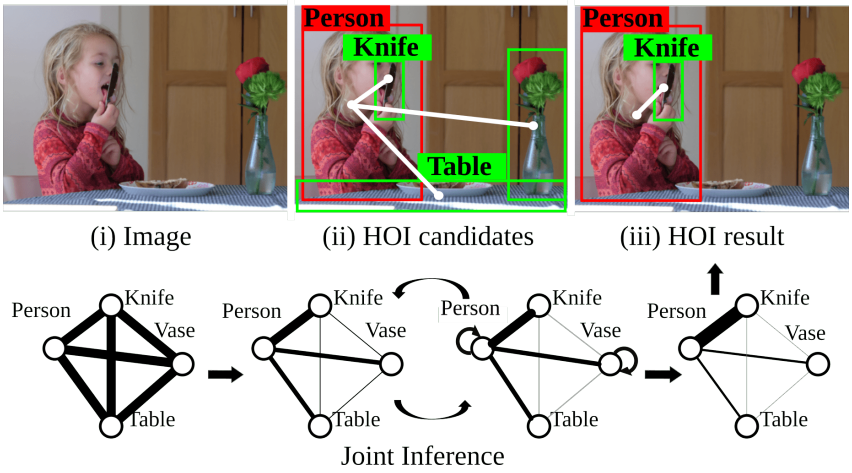


Figure 1: An illustration of GPNN in original paper [27]. It demonstrates the HOI detection in images. Using neural networks to represent graph models, GPNN can iteratively learn or infer the graph structure and message passing. The final parse graph explains the image with the graph structure and the node labels. For example, the graph structure shows the link between the person and the knife and the relationship and the relationship "lick". The intensity of information flow between nodes corresponds to the thickness of node edges.

GPNN demonstrates the effectiveness of graph representations and inference. After GPNN was proposed, other graph-based approaches have also emerged to tackle HOI detection and prediction, including human-object pairs with graph convolution [32], dual sub graphs learning [9], graph attention [24], etc. These models have all attempted to learn human-object relations through graph representations, where nodes typically represent humans and objects, and edges encode their interactions or relationships.

Three HOI datasets have been used to evaluate GPNN. HICO-DET [3] and VCOCO [16] are datasets for HOI detection in images, while CAD-120 [20] is for HOI recognition and anticipation in videos. Experimental results in [27] have been replicated and compared with other models, to demonstrate and verify GPNN's superiority to previous approaches. However, due to the fact that GPNN was completed many years ago, we will also compare it with some more updated methods in HOI task and discuss possible improvements.

2 Related Work

2.1 Graph Neural Network

GPNN is based on GNNs. GNN was firstly proposed in [30], it extended the neural network methods for processing the data represented in graph domains. In GNN, each node is defined by its own characteristic and characteristics of adjacent nodes, and is updated by propagating through the graph. Therefore, GNNs can intuitively represent relations of different entities in a given scene, and learn to

interpret relevant features or labels. Wu et al. [35] comprehensively proposed four categories for recent GNNs, including Recurrnet GNN, Convolutional GNN, Graph Autoencoder and Spatial-temporal GNN.

Recurrent Graph Neural Networks This framework is close to the prototype of GNN, in which the nodes constantly exchange information with adjacent nodes until the equilibrium. The graph structure (nodes and edges) is leveraged to iteratively update node representations through message-passing mechanisms. At each time step, information from neighboring nodes and edges is aggregated and combined with the node's previous state using recurrent mechanisms like LSTMs (Long Short-Term Memory units) [10] or Gated Recurrent Unit (GRU) [4]. This allows the network to capture both the static structure of the graph and the dynamic, temporal, or sequential dependencies within it.

Convolutional Graph Neural Networks In ConvGNN, different Gconvs are leveraged to represent the update of nodes. It can be further divided into two categories: Spectral-Based ConvGNN and Spatial-Based ConvGNN. In Spectral-Based ConvGNN, graph convolution is defined by introducing filters from the perspective of graph signal processing, where graph convolution operation is interpreted as removing noise from graph signals. While in Spatial-Based ConvGNN, graph convolution is defined by the spatial relationship between nodes.

Graph Autoencoders Graph Autoencoder is a deep neural network mapping the nodes into latent feature space and thus decoding the graphic information from it, and is widely used for learning the embedding and generating new graphs.

When Graph Autoencoder is used for learning network embedding, it utilizes an encoder to extract network embeddings and a decoder to enhance network embeddings to preserve the topological information of the graph. As for graph generation, Graph Autoencoders can learn the generative distribution of a graph by encoding it into a hidden representation and decoding the graph structure of the given hidden representation.

Spatial-Temporal Graph Neural Networks This framework can obtain the spatial and time reliability of graph simultaneously. The task of STGNN is to predict future node values or lables, or to predict spatiotemporal graph labels. For example, STGNN can be used for traffic speed prediction, driver maneuver prediction and human behavior recognition.

2.2 Human-Object Interaction

HOI before GPNN Early research on HOI includes Bayesian approach [14, 15], contextual relationship [36–38], structured representations incorporating spatial interaction and contextual understanding [6], compositional models [7], and HOI exemplars reference [17]. Later the advancement of neural network greatly boosted the exploration of HOI models. Well-established visual perception models been modified and applied to HOI, like Fast-RCNN [12] for [25] and Faster-RCNN [29] for [13]. Meanwhile, Shen et al. [31] utilized zero-shot learning to alleviate the long-tail issuse in HOI tasks. Chao et al. [3] proposed a multi-stream network to address the HOI detection challenge by processing human proposals, object regions, and their combinations. These neural models achieved remarkable performance compared to previous methods, but still faced the limitation of building a general representation for relations and interactions between human and object in both images and videos.

Interaction Points Since most HOI detection methods are based on objects and predict all the possible interactions using appearance characteristics, the efficiency and capability are limited. In recent works[33, 23], classification for intersection points is proposed. In [33], researchers built up a neural network including characteristic extraction, interaction generation and interaction grouping. By pairing the interaction vectors, they can be connected with HOI detection and outputs the prediction. In [23], the construction for parallel points detection and mapping (PPMD) simultaneously predict displacement both for objects and human, providing context and regularization for HOI implicitly to enhance the accuracy.

Prior Knowledge The giant gap from image to activity strongly decrease the performance of interaction prediction. Li et al. [22] found the fine-grained states of human parts, neglecting useless information. Based on this insight, researchers proposed to first infer the state of human body parts, and then infer activity based on semantic inference at the component level. As for scarce HOI categories, Kim et al. [19] modeled the interaction between these HOIs as co-occurrence matrix,

leveraging this prior to predict HOI. Compared to language prior, it can be obtained simply from statistics of training data, independent from external knowledge.

Graph-based Method Some works derive insights from graph-based models, which excel at capturing complex relationships between humans and objects. The iterative optimization nature of graph-based models further enhances their capacity of representing both spatial and semantic interactions. In VSGNet [32], humans and objects are treated as individual nodes, and human-object pairs are constructed. By using interaction proposal score as edge adjacency, which is generated from spatially refined features of human-object pair, the graph convolution branch of VSGNet effectively learns the relational structure between humans and objects. DRG [9] builds human-centric sub-graphs—connecting each human with all objects—along with object-centric sub-graphs to for learning dual relation. By inserting HOI nodes for each edge and training a attentional graph convolutional network, DRG computes action scores of each stream for final prediction. Expanding on these approaches, Lin et al. [24] proposes an innovative representation, treating human-object pairs as nodes and utilizing Graph Attention Networks (GATs) to filter out irrelevant pairs.

3 Methodology

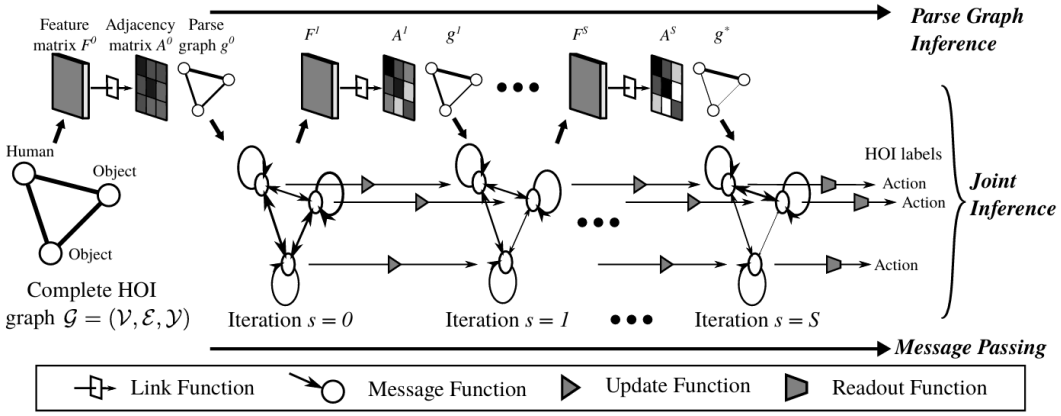


Figure 2: An illustration of the forward pass of GPNN in original paper [27]. It takes node and edge features as input, and computes a parsing graph in a message passing manner.

The method we use to replicate GPNN is basically consistent with the original paper. We regard human and objects as nodes and the relationships between them as edges, and try to compute a parsing graph where only the meaningful edges are kept and the nodes are labeled.

As a description of symbols, $\mathcal{V}, \mathcal{E}, \mathcal{Y}$ represent the node set, edge set, and label set, respectively. Nodes $v \in \mathcal{V}$ traverse the set $\{1, \ldots, |\mathcal{V}|\}$. Edges $e \in \mathcal{E}$ are represented as two-tuples $(v, w) \in \mathcal{V} \times \mathcal{V}$. Each node v has an output state $y_v \in \mathcal{Y}$ that takes its value from $\{1, \ldots, Y_v\}$ as actions or other labels. Then $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{Y})$ denotes the complete HOI graph. Our goal is to obtain a sub-graph of $\mathcal{G}$, namely a parsing graph $g^* = (\mathcal{V}_{g^*}, \mathcal{E}_{g^*}, \mathcal{Y}_{g^*})$ where $\mathcal{V}_{g^*} \subseteq \mathcal{V}$ and $\mathcal{E}_{g^*} \subseteq \mathcal{E}$, which can best explain the HOI relationship of the input. Given the node features $\Gamma^{\mathcal{V}}$ and edge features $\Gamma^{\mathcal{E}}$, and let $\Gamma = \{\Gamma^{\mathcal{V}}, \Gamma^{\mathcal{E}}\}$, then we have the formula for optimization:

$$
\begin{aligned}
g^* &= \arg\max_g p(g|\Gamma, \mathcal{G}) = \arg\max_g p(\mathcal{V}_g, \mathcal{E}_g, \mathcal{Y}_g|\Gamma, \mathcal{G}) \\
&= \arg\max_g p(\mathcal{Y}_g|\mathcal{V}_g, \mathcal{E}_g, \Gamma) p(\mathcal{V}_g, \mathcal{E}_g|\Gamma, \mathcal{G})
\end{aligned} \tag{1}
$$

The GPNN is designed to approximate the computation of the probability of the graph structure $p(\mathcal{V}_g, \mathcal{E}_g|\Gamma, \mathcal{G})$ and the probability of the label $p(\mathcal{Y}_g|\mathcal{V}_g, \mathcal{E}_g, \Gamma)$ for nodes in the parse graph. There are four types of functions as individual modules in the forward pass of a GPNN named *link functions, message functions, update functions, readout functions* (as illustrated in Fig. 2). Here are the details of those four types of functions and the forward pass of GPNN.

Link Function The link function takes the node features $\Gamma^{\mathcal{V}}$ and edge features $\Gamma^{\mathcal{E}}$ as input, and outputs an adjacency matrix $A \in [0, 1]^{|\mathcal{V}| \times |\mathcal{V}|}$. In implementation, at step s, the node features

(hidden states, mentioned later) $\{h_v^{s-1} \in \mathbb{R}^{d_V}\}_v$ and the edge features (messages, mentioned later) $\{m_{vw}^{s-1} \in \mathbb{R}^{d_E}\}_{v,w}$ are concatenated into a feature matrix $F^{s-1} \in \mathbb{R}^{|V|\times|V|\times(2d_V+d_E)}$, where d_V and d_E denote the dimensions of the node features and edge features, respectively. Then we can compute the adjacent matrix as:

$$A^s = \sigma(\mathbf{W}^L * F^{s-1}) \tag{2}$$

where $\mathbf{W}^L$ denotes the learnable parameters of the link function and $*$ denotes convolution operation with $1 \times 1 \times (2d_V + d_E)$ kernels. For spatial-temporal problems, the link function can be substituted with convolutional LSTM.

Message Function The message function is designed to summarize messages to the target nodes of other nodes. In the process of calculating messages, the adjacency matrix of current step, the hidden states of previous step and the edge features are needed. In implementation, the calculation of messages is:

$$m_{vw}^s = A_{vw}^s[\mathbf{W}_V^M h_v^{s-1}, \mathbf{W}_V^M h_w^{s-1}, \mathbf{W}_E^M \Gamma_{vw}] \tag{3}$$

$$m_v^s = \sum_w m_{vw}^s \tag{4}$$

where $[.,.]$ denotes concatenation. The concatenation is the output of the message function, which can be viewed as the concatenation of hidden states and edge features through several fully connected layers.

Update Function The update function is designed to update the hidden states according to the income messages. The calculation of hidden states takes the hidden states of previous step and the messages of current step as input. In implementation, the calculation of hidden states is:

$$h_v^s = GRU(h_v^{s-1}, m_v^s) \tag{5}$$

where GRU [4] is utilized as the update function. And in the beginning, the first hidden states h_v^0 are initialized by the node features Γ_v.

Readout Function The readout function is designed to classify the final hidden state into labels. And the calculation of labels is as simple as:

$$y_v = \varphi(\mathbf{W}^R h_S^v) \tag{6}$$

where the activate function $\varphi(\cdot)$ can be used as *softmax* or *sigmoid* in accordance with different HOI tasks.

Given the above four functions, we can pecerceive the process of inferring adjacency matrix, calculating messages, and updating hidden states as an iteration, and the network executes its forward pass iteratively. In this way, the implementation of GPNN makes it fully differentiable and trainable from end to end.

4 Replication Details

The source code can be found from `https://github.com/SiyuanQi-zz/gpnn`. Since the repository was built 6 years ago, the Python environment configuration is different. It was based on Python 2.7, yet most current Python versions are 3.7 or above, which creates significant incompatibility, even with tools like Anaconda. Therefore, we downloaded code files and revised them according to Python 3.9, correcting grammar and removing deprecated usage, while minimizing changes to the original code structure.

Another problem arrives when fetching datasets and pre-trained models. The Google Drive link on the original Github website is failed, which poses great challenges to our replication work. Fortunately, we contacted Professor Yixin Zhu, who directly asked the author of [27] for a new link. This helps us to verify the correctness of our modifications and reproduce experimental results.

We use a single NVIDIA GeForce RTX 4080 GPU for all training and inference tasks, as the models are relatively small and the computational demands are not particularly high. In our reproduction of the original work, there are certain deviations from the exact hyper-parameters settings described in [27]. For model trained on VCOCO dataset, the learning rate in its corresponding original Python

file is 1e-4, rather than 1e-3 mentioned in the original paper, while the default batch size is 1, and the model has only been trained for 10 epochs. For model trained on HICO-DET dataset, its default learning rate is 1e-5, and the batch size is still 1, also trained for 10 epochs. For CAD-120, Qi et al. [27] had not provided related hyper-parameters, so we tentatively use a learning rate of 5e-5, a batch size of 1, and train the model for 70 epochs. We compare the experimental results in the original paper with the performance of models that we practically trained, to verify reproducibility and discuss discrepancies. For the sake of simplicity, we use **GPNN** and **Our GPNN** to represent these two models,

Despite these adjustments, the results of our experiments closely align with those reported in the paper, demonstrating the general consistency and robustness of the proposed method. Nevertheless, we acknowledge that some discrepancies may arise from these differences in hyperparameter choices and further exploration is required to validate the findings under a wider range of settings.

5 Results

To prove the effectiveness of our implementations, we train and evaluate three models on HICO-DET [3], VCOCO [16], CAD-120 [20], respectively. We compare the performance of our models and the original models and show discrepancies. We also include results from other HOI models.

Using the same benchmark as HICO-DET[3] and V-COCO[16] datasets, we employed mean average precision (mAP) to evaluate HOI detection of the model. One HOI detection is correct if and only if the human detection, the object detection and the interaction class are all correct. When IoU(intersection over union) between the bounding box of a detected person or object and its ground truth is greater than 0.5, it is considered to be detected correctly.

For HICO-DET dataset, we performed the same processing as in the paper and conducted three testing experiments. The first is for all 600 HOI categories in the test dataset(Full), the second is for HOI categories with less than 10 instances in the training set(Rare), and the third is for other HOI categories in the test set except for those has been tested in the second experiment(Non-Rare). Results of these three experiments as well as results of other methods under same settings, are reported in Tab. 1. For V-COCO dataset, the original paper [27] explores two test sets of HOI categories: those with only one target object and those with two target objects. But until now we have not found a method to divide the V-COCO test set according to this rule, so we only tested mAP on the total test set, and results of the experiments are reported in Tab. 2 as well as results of other methods under same settings. Some examples of V-COCO pictures processed by our GPNN are shown in Fig. 3. It can be obeserved that the GPNN model we reproduced achieved better results on V-COCO dataset. It should be noted that we also retested pre-trained GPNN with provided weights and found that it also achieves a better results than reported in the original paper, which requires a further study in depth for explanations. On HICO-DET dataset, HOI categories that frequently appear in the training set were easier to be detected correctly than those that appear less frequently, which may be related to slight overfitting caused by multiple training epochs.

We also tested the model's performance on video on the CAD-120 dataset. The model is required to detect and predict the human sub-activity labels and object affordance labels through the video. Same as the paper, we also compared our model with anticipatory temporal CRF (ATCRF)[20] and structural RNN (S-RNN)[18], both of which are state-of-the-art methods for this problem. Tab. 3 shows F1-scores averaged on detection and activity anticipation tasks, and Fig. 4 are confusion matrices for detecting and predicting these two tasks. It can be seen that the GPNN model we reproduced has some slight fluctuations compared to the data presented in the paper [27]. However, the overall performance of GPNN in both tasks, especially in anticipation, is superior to ATCRF and S-RNN. This is due to the advantage of GPNN in deeply combining graph structure with neural networks, which does not rely on fixed graph structures, and is compatible with more complex relationships with stronger expressive power.

Table 1: HOI detection results on HICO-DET [3] test images on mAP (%). All results, except those generated from our model, can be found in paper[27, 33, 23, 22, 32, 24, 19, 9].

Methods	**Full (mAP %)** ↑	**Rare (mAP %)** ↑	**Non-rare (mAP %)** ↑
Random	1.35×10^{-3}	5.72×10^{-4}	1.62×10^{-3}
Fast-RCNN (union) [12]	1.75	0.58	2.10
Fast-RCNN (score) [12]	2.85	1.55	3.23
HO-RCNN [3]	5.73	3.21	6.48
HO-RCNN + IP [3]	7.30	4.68	8.08
HO-RCNN + IP + S[3]	7.81	5.37	8.54
Gupta and Malik [16]	9.09	7.02	9.71
Maraghi et al. [26]	6.46	4.24	7.12
InteractNet[13]	9.94	7.16	10.77
Wang et al. [33]	19.56	12.79	21.58
PPDM-Hourglass [23]	21.73	13.78	24.10
TIN [21]	17.03	13.42	18.11
TIN [21]+PaStaNet-Linear [22]	22.65	**21.17**	23.09
VSGNet [32]	19.80	16.05	20.91
AGRR [24]	16.63	11.30	18.22
ACP [19]	20.59	15.92	21,98
DRG [9]	**24.53**	19.47	**26.04**
GPNN [27]	13.11	**9.34**	14.23
Our GPNN	**15.87**	6.08	**16.67**

Table 2: HOI detection results (mAP) on V-COCO [16] dataset. Legend: *Set 1* indicates 18 HOI actions with one object, and *Set 2* corresponds to 3 HOI actions (*i.e.*, cut, eat, hit) with two objects (*instrument* and *object*). All results, except those generated from our model, can be found in paper[27, 33, 22, 32, 24, 19, 9].

Method	**Ave. (mAP %)** ↑
Gupta and Malik [16]	31.8
InteractNet [13]	40.0
Wang et al. [33]	51.0
TIN[21]+PaStaNet-Linear [22]	51.0
VSGNet [32]	51.76
AGRR [24]	48.1
ACP [19]	**52.98**
DRG [9]	51.0
GPNN (In paper) [27]	44.0
GPNN (Provided Weights, retest)	**64.36**
Our GPNN	64.03

Table 3: Human activity detection and future anticipation results on CAD-120 dataset, measured via F1-score.

Method	**Detection (F1-score)** ↑		**Anticipation (F1-score)** ↑	
	Sub-activity (%)	**Object Affordance (%)**	**Sub-activity (%)**	**Object Affordance (%)**
ATCRF [20]	80.4	81.5	37.9	36.7
S-RNN [18]	83.2	88.7	62.3	80.7
S-RNN (multi-task) [18]	82.4	**91.1**	65.6	80.9
GPNN	**88.9**	88.8	**75.6**	**81.9**
Our GPNN	85.7	85.7	74.9	79.3

6 Discussion and Reflection

6.1 Visual Perception Backbone

HOI task is a crucial aspect of physical scene understanding. From the perspective of cognitive science, to detect and predict HOI requires both physical and social commonsense to recognize various entities in a given scene and interpret their relations. Therefore, combining off-the-shelf visual detection model and reasoning model is a fundamental approach. Visual perception backbone can pre-process

(a) (b)

(c) (d)

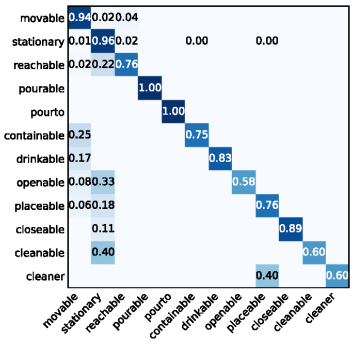
Figure 3: Examples of V-COCO processed by our model.

and compress inputs into a lower-dimensional feature space, which facilitates subsequent reasoning models to fully leverage the compact and salient features, while also reduce the computational cost for training and inference.

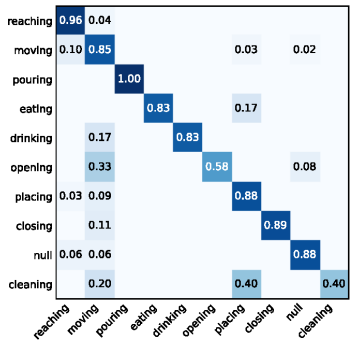
GPNN uses a pre-trained deformable convolution network [5] for object detection and feature extraction. To improve the performance of GPNN, more advanced object detection models can be utilized and incorporated into the framework, like DETR [1], YOLO [28]. On the other hand, to unify the entity detection and HOI task, a possible new direction is to train visual backbone and reasoning model together, to achieve a better alignment. This can train the visual backbone to extract features that are more relevant to HOI and ignoring irrelevant information. However, this requires a large amount of data that are annotated with HOI labels, while also increases the computational cost for training.

6.2 Graph Representation

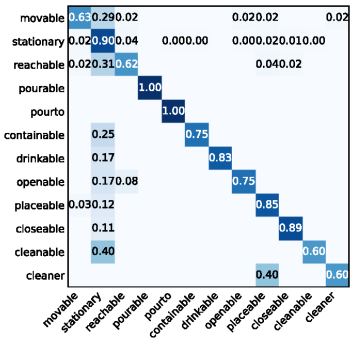
Theoretically, graph representation is particularly suitable for HOI task, since it provides an effective framework for capturing and modeling the complex relationships. In GPNN, learnable adjacent matrix is used to represent nodes and edges in a given graphs. This can explicitly and intuitively represent the structure of graphs, and is particularly compatible with linear algebra operations used in many graph algorithms. However, its memory usage requires $O(n^2)$ space, which makes adjacent matrix impractical for large graphs. Furthermore, dynamic graphs pose challenges, as adding or removing nodes necessitates resizing the matrix, which can be computationally expensive. Thus, adjacency matrices are best suited for small, dense graphs. In the three datasets [3, 16, 20] used to train GPNN, most scenarios are relatively simple and adjacent matrices are usually small.



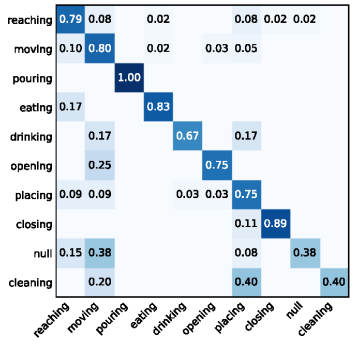
Affordance confusion matrices of HOI detection



Sub-activity confusion matrices of HOI detection



Affordance confusion matrices of HOI anticipation



Sub-activity confusion matrices of HOI anticipation

Figure 4: Confusion matrices on CAD-120.

There are other graph representation feasible for large and sparse graphs, including adjacency list and edge list, yet they are not directly compatible with matrix operations. An ideal model for parsing HOI should have the ability to automatically build graph structures and then analyze relations. Currently, introducing attention mechanism is a possible solution, since it can capture both local and global features and autoregressively generate sequential outputs. Works like [24] have already successfully incorporated attention map to guide models to achieve fine-grained HOI detection. Furthermore, specially-designed transformer blocks can be used to learn and generate representations of graphs with various structures and complexities, since the representation of a graph can also be sequentially constructed, starting from a small sub-graph and gradually expanding into the entire representation of a given scene.

7 Conclusion

We successfully review and replicate Graph Parsing Neural Network (GPNN) in this project. We have obtained relatively consistent results compared to the original paper and discussed discrepencies. We thoroughly analyze the design and structure of GPNN to understand its essence. Using link functions, message functions, update functions, and readout functions, the model effectively combines graph representations and end-to-end neural networks for HOI tasks in both images and videos. After comparing GPNN with other methods, including some more updated models, we propose several possible directions for future work. With the development of computer vision and cognitive science, more advanced models for HOI can be expected in the future.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 8

[2] Yu-Wei Chao, Zhan Wang, Yugeng He, Jiaxuan Wang, and Jia Deng. Hico: A benchmark for recognizing human-object interactions in images. In *Proceedings of the IEEE international conference on computer vision*, pages 1017–1025, 2015. 1

[3] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *2018 ieee winter conference on applications of computer vision (wacv)*, pages 381–389. IEEE, 2018. 1, 2, 3, 6, 7, 8

[4] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In Dekai Wu, Marine Carpuat, Xavier Carreras, and Eva Maria Vecchi, editors, *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-4012. URL https://aclanthology.org/W14-4012. 3, 5

[5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 8

[6] Vincent Delaitre, Josef Sivic, and Ivan Laptev. Learning person-object interactions for action recognition in still images. *Advances in neural information processing systems*, 24, 2011. 3

[7] Chaitanya Desai and Deva Ramanan. Detecting actions, poses, and objects with relational phraselets. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part IV 12*, pages 158–172. Springer, 2012. 3

[8] Hao-Shu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. Learning pose grammar to encode human body configuration for 3d pose estimation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. 2

[9] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. DRG: dual relation graph for human-object interaction detection. *CoRR*, abs/2008.11714, 2020. URL https://arxiv.org/abs/2008.11714. 2, 4, 7

[10] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. *Neural computation*, 12(10):2451–2471, 2000. 3

[11] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017. 2

[12] Ross Girshick. Fast r-cnn, 2015. URL https://arxiv.org/abs/1504.08083. 3, 7

[13] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8359–8367, 2018. 1, 3, 7

[14] Abhinav Gupta and Larry S Davis. Objects in action: An approach for combining action understanding and object perception. In *2007 IEEE Conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007. 3

[15] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(10):1775–1789, 2009. 3

[16] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling, 2015. URL https://arxiv.org/abs/1505.04474. 2, 6, 7, 8

[17] Jian-Fang Hu, Wei-Shi Zheng, Jianhuang Lai, Shaogang Gong, and Tao Xiang. Recognising human-object interaction via exemplar based modelling. In *Proceedings of the IEEE international conference on computer vision*, pages 3144–3151, 2013. 3

[18] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 5308–5317, 2016. 2, 6, 7

[19] Dong-Jin Kim, Xiao Sun, Jinsoo Choi, Stephen Lin, and In So Kweon. Detecting human-object interactions with action co-occurrence priors. *CoRR*, abs/2007.08728, 2020. URL https://arxiv.org/abs/2007.08728. 3, 7

[20] Hema S Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):14–29, 2015. 2, 6, 7, 8

[21] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 7

[22] Yong-Lu Li, Liang Xu, Xinpeng Liu, Xijie Huang, Yue Xu, Shiyi Wang, Haoshu Fang, Ze Ma, Mingyang Chen, and Cewu Lu. Pastanet: Toward human activity knowledge engine. *CoRR*, abs/2004.00945, 2020. URL https://arxiv.org/abs/2004.00945. 3, 7

[23] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, and Jiashi Feng. PPDM: parallel point detection and matching for real-time human-object interaction detection. *CoRR*, abs/1912.12898, 2019. URL http://arxiv.org/abs/1912.12898. 3, 7

[24] Xue Lin, Qi Zou, and Xixia Xu. Action-guided attention mining and relation reasoning network for human-object interaction detection. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 1104–1110. International Joint Conferences on Artificial Intelligence Organization, 7 2020. doi: 10.24963/ijcai.2020/154. URL https://doi.org/10.24963/ijcai.2020/154. Main track. 2, 4, 7, 9

[25] Arun Mallya and Svetlana Lazebnik. Learning models for actions and person-object interactions with transfer to question answering. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 414–428. Springer, 2016. 3

[26] Vali Ollah Maraghi, Karim Faez, and Miguel Cazorla. Scaling human-object interaction recognition in the video through zero-shot learning. *Intell. Neuroscience*, 2021, January 2021. ISSN 1687-5265. doi: 10.1155/2021/9922697. URL https://doi.org/10.1155/2021/9922697. 7

[27] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 401–417, 2018. 1, 2, 4, 5, 6, 7

[28] J Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 8

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016. 3

[30] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009. doi: 10.1109/TNN.2008.2005605. 2

[31] Liyue Shen, Serena Yeung, Judy Hoffman, Greg Mori, and Li Fei-Fei. Scaling human-object interaction recognition through zero-shot learning. 2018. 1, 3

[32] Oytun Ulutan, A. S. M. Iftekhar, and Bangalore S. Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. *CoRR*, abs/2003.05541, 2020. URL https://arxiv.org/abs/2003.05541. 2, 4, 7

[33] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. *CoRR*, abs/2003.14023, 2020. URL https://arxiv.org/abs/2003.14023. 3, 7

[34] Wenguan Wang, Yuanlu Xu, Jianbing Shen, and Song-Chun Zhu. Attentive fashion grammar network for fashion landmark detection and clothing category classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4271–4280, 2018. 2

[35] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2021. doi: 10.1109/TNNLS.2020.2978386. 3

[36] Bangpeng Yao and Li Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 9–16. IEEE, 2010. 3

[37] Bangpeng Yao and Li Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 17–24. IEEE, 2010.

[38] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *2011 International conference on computer vision*, pages 1331–1338. IEEE, 2011. 3