Supplementary Material for
Graph Matching with Bi-level Noisy Correspondence

Yijie Lin[1], Mouxing Yang[1], Jun Yu[2], Peng Hu[1], Changqing Zhang[3*], Xi Peng[1*]
[1] Sichuan University, [2] Hangzhou Dianzi University, [3] Tianjin University
{linyijie.gm,yangmouxing,penghu.ml,pengx.gm}@gmail.com,
yujun@hdu.edu.cn, zhangchangqing@tju.edu.cn

1. Introduction

In this supplementary material, we first present the proof of Theory 1. After that, we present experiment details about the network architectures and more experiment results to further investigate the effectiveness of our method. Finally, we discuss the broader impact of our work.

2. Proof of Theorem 1

This theorem is based on the Proposition 2 of [12]. We refer the readers to [12] for more explanations.

Theorem 1 *The log-sum-exp [1] smoothed structured linear assignment loss L with row-stochastic relaxation is equivalent to the InfoNCE contrastive loss [7, 3].*

Proof 1 *We first relax the constraint $\mathbf{Y} \in \Pi$ to $\mathbf{Y} \in \mathcal{R}$ where $\mathcal{R}$ is a set of row-stochastic binary matrix, i.e., $[\mathbf{Y}]_{ij} \in \{0,1\}$ and $\sum_j \mathbf{Y}_{ij} = 1\ \forall i$. Based on it, we reformulate the structured linear assignment loss as,*

$$\begin{aligned}
L &= -\operatorname{tr}\left(\mathbf{S}\mathbf{Y}_{gt}^{\top}\right) + \max_{\mathbf{Y}\in\mathcal{R}} \operatorname{tr}(\mathbf{S}\mathbf{Y}^{\top}) \\
&= -\operatorname{tr}\left(\mathbf{S}\mathbf{Y}_{gt}^{\top}\right) + \max_{y_1\cdots y_n} \sum_i (\sum_j [\mathbf{S}]_{ij}\,[y_i]_j) \\
&= -\operatorname{tr}\left(\mathbf{S}\mathbf{Y}_{gt}^{\top}\right) + \sum_i \max_{y_i} (\sum_j [\mathbf{S}]_{ij}\,[y_i]_j) \\
&= -\operatorname{tr}\left(\mathbf{S}\mathbf{Y}_{gt}^{\top}\right) + \sum_i \max_j\, [\mathbf{S}_{ij}]\,,
\end{aligned} \tag{1}$$

where y_i is the i-th row of $\mathbf{Y}$. The third identity is based on the independence of the rows $y_1, \ldots, y_n$ and the last identity follows the fact that y_i is a one-hot vector containing the maximum index. As the structured linear loss is non-smoothness and difficult to optimize, we utilize the common log-sum-exp approximation [1] on the max function which leads to,

$$L = -\sum_{(i,j)\in\mathbf{Y}_{gt}} [\mathbf{S}]_{ij} + \tau \sum_i \log(\sum_j \exp(\frac{1}{\tau}[\mathbf{S}]_{ij})), \tag{2}$$

where τ controls the degree of smoothness. In fact, Eq. (2) is the so-called InfoNCE contrastive loss where the first and second term refer to the alignment and uniformity property [19, 6], respectively. □

3. Details of Network Architectures

Given two graphs $\mathcal{G}_A = \{\mathbf{U}_A, \mathbf{E}_A\}$ and $\mathcal{G}_B = \{\mathbf{U}_B, \mathbf{E}_B\}$ with n and m keypoints each ($n \le m$). $\mathbf{U}$ indicates the set of nodes and $\mathbf{E}$ denotes the set of edges. The node and edge features are learned through the base encoder whose structure is nearly the same as that of BBGM [14] and NGM-v2 [18]. Concretely, the base encoder consists of an image encoder, a graph neural network, and a projection head. The proposed momentum encoder is with the same structure as the base encoder.

Image encoder. Following [18, 5, 14, 13, 10, 16, 17], we employ VGG16 [15] as the image encoder to extract the node features. Specifically, we extract the node features from relu4_2 and relu5_1 of VGG16, and concatenate them to form the initial node feature matrices $\bar{\mathbf{U}}_A \in \mathbb{R}^{n\times d_1}, \bar{\mathbf{U}}_B \in \mathbb{R}^{m\times d_1}$ where $d_1 = 1024$.

Graph neural network. Following [18, 14, 10, 13], we initial the edge structure $\mathbf{E}_A \in \mathbb{R}^{n\times n}$ and $\mathbf{E}_B \in \mathbb{R}^{m\times m}$ with Delaunay triangulation and $[\mathbf{E}]_{ij}$ is weighted as the difference between the coordinate positions of keypoint i and j. We pass the initial node features $\bar{\mathbf{U}}$ and the edge structure $\mathbf{E}$ through graph network SplineCNN [4], which is a powerful graph convolution network that encodes geometric features into node features by updating the node representation via a weighted summation of its neighbors. Formally, the update rule at keypoint i is,

$$\operatorname{SplineCNN}\left([\bar{\mathbf{U}}]_i\right) = \frac{1}{|\mathcal{N}(i)|} \sum_{j\in\mathcal{N}(i)} [\bar{\mathbf{U}}]_j \cdot g([\mathbf{E}]_{ij}), \tag{3}$$

where $\mathcal{N}(i)$ indicates the neighbors of node i, g is the B-Spline kernel, and $\cdot$ is the dot product. Separately feeding

*Corresponding author

the graphs $\mathcal{G}_A$ and $\mathcal{G}_B$ into SplineCNN, we obtain the refined node features $\hat{\mathbf{U}}_A \in \mathbb{R}^{n\times d_2}, \hat{\mathbf{U}}_B \in \mathbb{R}^{m\times d_2}$ where $d_2 = 1024$, respectively.

Projection Head. Following classical contrastive learning paradigms [2, 3, 8, 9, 11], we obtain the final node feature $\mathbf{V}_A \in \mathbb{R}^{n\times d_3}$ and $\mathbf{V}_B \in \mathbb{R}^{m\times d_3}$ where $d_3 = 256$ through two fully-connected layers (FCN). Formally,

$$\mathbf{V} = \mathrm{norm}\left(f_2\left(f_1(\hat{\mathbf{U}})\right)\right), \tag{4}$$

where FCN f_1 and f_2 are with the batch normalization layer and ReLU activation. norm operation denotes ℓ_2 normalization and the dimensionality of f_1 and f_2 is set to 1024 and 256, respectively.

Finally, we obtain the node similarity matrix $\mathbf{S}$ and the edge adjacency matrices $\mathbf{F}_A, \mathbf{F}_B$ through $\mathbf{S} = \mathbf{V}_A\mathbf{V}_B^\top$, $\mathbf{F}_A = \mathbf{V}_A\mathbf{V}_A^\top$, and $\mathbf{F}_B = \mathbf{V}_B\mathbf{V}_B^\top$.

4. Visualization on Graph Matching

We present the visual matching results of our method and the most comparable baselines BBGM [14] and ASAR [13] on the Pascal VOC and Spair-71k datasets. For better visualization, we crop the object according to its bounding box. As shown in Figs. 5 and 6, our method achieves superior matching performance, especially for the image pairs with high viewpoint difficulty and low recognizability.

5. Broader Impact

This work could be the first work that reveals the importance of the noisy correspondence problem in graph matching. Solving this problem could improve the tolerance for the errors of annotations, which might benefit the practitioners in the industry. Although the proposed COMMON achieves remarkable improvement, the complexity of training the model is slightly larger due to the additional momentum network. In practice, we find the time cost is approximately $\times$1.4 times that of training a base encoder only. Fortunately, the inference speed is exactly the same as we only keep the base encoder during testing.

References

[1] Amir Beck and Marc Teboulle. Smoothing and first order methods: A unified framework. *SIAM Journal on Optimization*, 22(2):557–580, 2012. 1

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv:2002.05709*, 2020. 2

[3] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020. 1, 2

[4] Matthias Fey, Jan Eric Lenssen, Frank Weichert, and Heinrich Müller. Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In *CVPR*, pages 869–877, 2018. 1

[5] Quankai Gao, Fudong Wang, Nan Xue, Jin-Gang Yu, and Gui-Song Xia. Deep graph matching under quadratic constraint. In *CVPR*, pages 5069–5078, 2021. 1

[6] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *EMNLP*, 2021. 1

[7] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 1

[8] Yijie Lin, Yuanbiao Gou, Xiaotian Liu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. Dual contrastive prediction for incomplete multi-view representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–14, 2022. 2

[9] Yijie Lin, Yuanbiao Gou, Zitao Liu, Boyun Li, Jiancheng Lv, and Xi Peng. Completer: Incomplete multi-view clustering via contrastive prediction. In *CVPR*, pages 11174–11183, 2021. 2

[10] Chang Liu, Shaofeng Zhang, Xiaokang Yang, and Junchi Yan. Self-supervised learning of visual graph matching. In *ECCV*, 2022. 1

[11] Junhong Liu, Yijie Lin, Liang Jiang, Jia Liu, Zujie Wen, and Xi Peng. Improve interpretability of neural networks via sparse contrastive coding. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 2022. 2

[12] Artem Moskalev, Ivan Sosnovik, Volker Fischer, and Arnold Smeulders. Contrasting quadratic assignments for set-based representation learning. In *ECCV*, pages 88–104, 2022. 1

[13] Qibing Ren, Qingquan Bao, Runzhong Wang, and Junchi Yan. Appearance and structure aware robust deep visual graph matching: Attack, defense and beyond. In *CVPR*, pages 15263–15272, 2022. 1, 2

[14] Michal Rolínek, Paul Swoboda, Dominik Zietlow, Anselm Paulus, Vít Musil, and Georg Martius. Deep graph matching via blackbox differentiation of combinatorial solvers. In *ECCV*, pages 407–424, 2020. 1, 2

[15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2014. 1

[16] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Learning combinatorial embedding networks for deep graph matching. In *ICCV*, pages 3056–3065, 2019. 1

[17] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Combinatorial learning of robust deep graph matching: an embedding based approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1

[18] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Neural graph matching network: Learning lawler's quadratic assignment problem with extension to hypergraph and multiple-graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1

[19] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, pages 9929–9939, 2020. 1

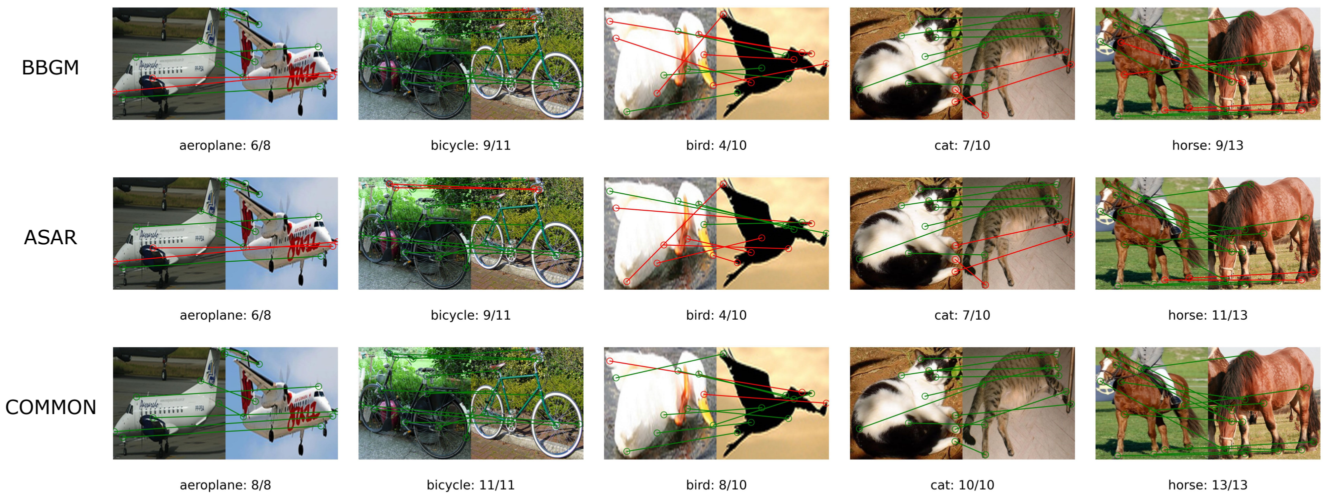

Figure 5. **Visualization of the matching results** on Pascal VOC. Green and red lines denote correct and false matching results, respectively.

Figure 6. **Visualization of the matching results** on SPair-71k.