Supplementary: Deep Graphical Feature Learning for the Feature Matching Problem

Zhen Zhang[¶]
Australian Institute for Machine Learning
School of Computer Science
The University of Adelaide
zhen@zzhang.org

Wee Sun Lee
Department of Computer Science
National University of Singapore
leews@comp.nus.edu.sg

A. Additional Experimental Results

In this section, we provide additional experimental result.

A.1. Generalization of the Model

First we show the generalization of the our model with different graph structure. In our training, we use k-nearest neighbor to construct the graph for our graph neural network with $k = 8$. In this section, we test the trained model In this section, we will test the model with different k. The experimental result on synthetic data generated in Section 4.1 is shown in Figure 9. Through our model is trained with $k = 8$, from Figure 9 we can see that the proposed can easily generalize to $k = 4$ or $k = 12$. For $k = 2$, obviously it is hard to gather information for local neighbor with such parameter, and thus there is a significant gap between $k = 2$ and other cases.

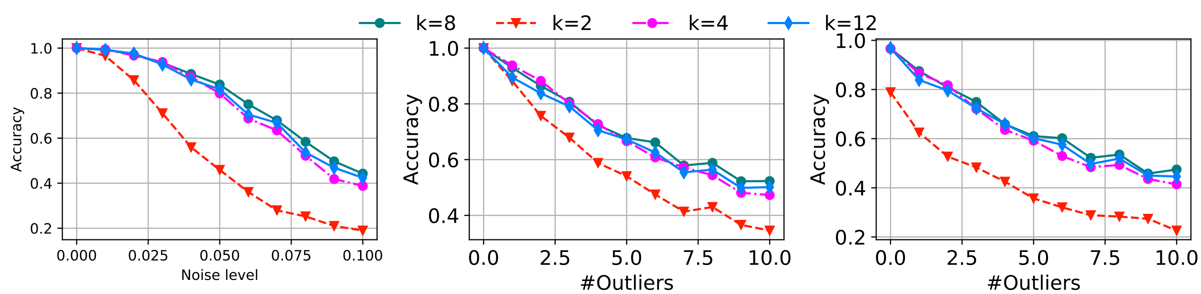

Figure 9: Performance evaluation with different parameter k for our model on synthetic dataset.

We also tested the CMU House dataset with different k for constructing the graph. The result is shown in Figure 10. On the CMU house dataset, for $k = 8$ and $k = 12$, our model gives perfect matching, while for $k = 4$ our model can not handle feature points pairs with large view angle changes perfectly. The parameter $k = 2$ fails as it results in a poor local information gather scheme. In terms of running time, different k results almost the same running time. The slight difference might be caused by the different running time of Hungarian or data-preprocess.

B. Detailed Running Time Analysis

The whole pipeline of the proposed method including three parts: (1) prepare the data, including normalize the data, constructing the graph and copy all data to GPU; (2) running inference on GPU, which outputs the feature similarity matrix; (3) running inference over a Linear Assignment Problem (LAP) [2] using Hungarian algorithm. We tested the running time of each part on the CMU house dataset (shown in Figure 11) with different k. The results suggests that for all k and different

[¶]This work is done at the Department of Computer Science, National University of Singapore.

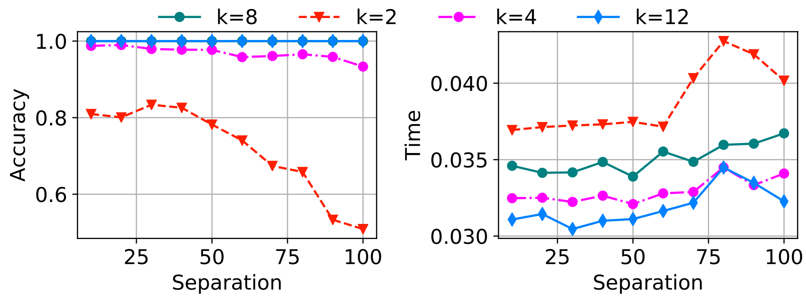

Figure 10: Performance evaluation with different k for our model on CMU house dataset.

separation, the inference time of the neural network (NN-Infer) and the data preparation (PreProcess) is almost the same. For the inference of the LAP (LAP-Infer), the running time increases as the separation increases. It is also notable that for $k = 2$, it takes significantly longer LAP-Infer time. From previous section, we can see that the feature provided by our model $k = 2$ will results lower accuracy due to its poor quality. In this section, we also note that the low quality feature make it hard to distinguish the correct matching from incorrect ones, which will results in harder inference problem and longer inference time.

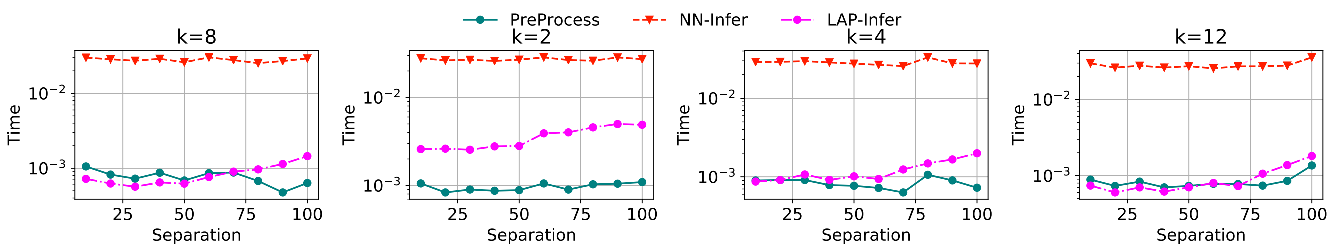

Figure 11: Detailed running time for different k.