Multi-Robot Active Mapping via Neural Bipartite Graph Matching

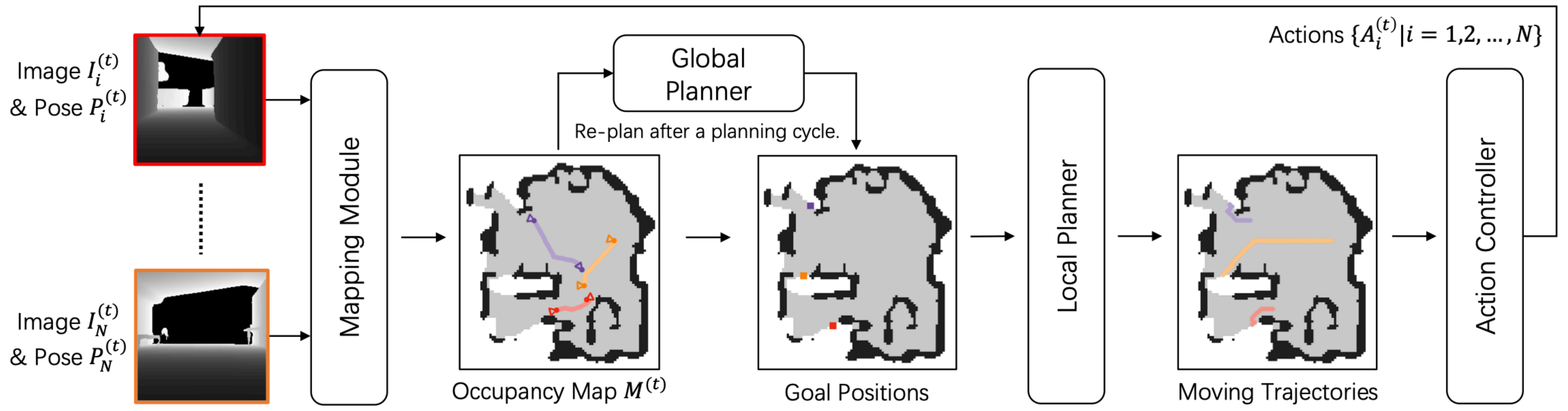
Kai Ye[1]* Siyan Dong[2,1]* Qingnan Fan[3]† He Wang[1] Li Yi[4] Fei Xia[5] Jue Wang[3] Baoquan Chen[1]†

*Joint first authors

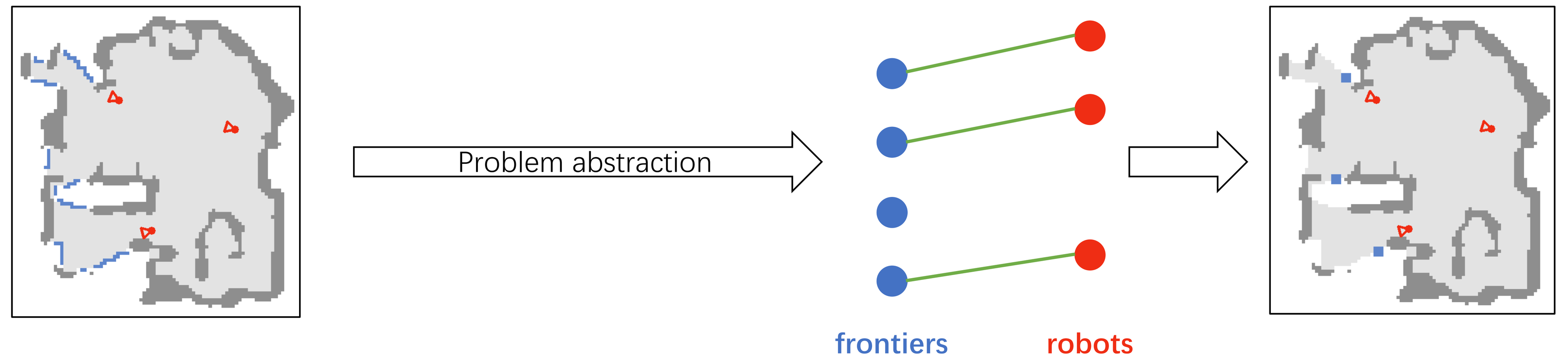
†Corresponding authors

• Problem Definition of Multi-Robot Active Mapping

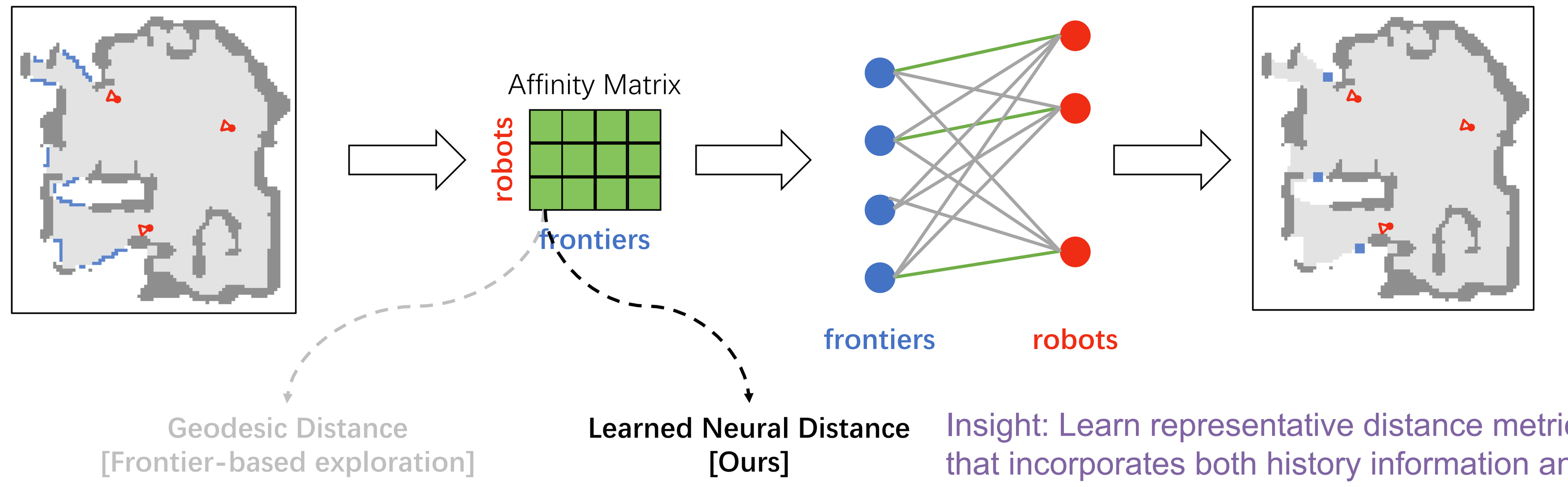
- Input: depth images, camera poses
- Output: scene map, actions



• Challenge: Goal positions estimation for robots to achieve efficient collaboration

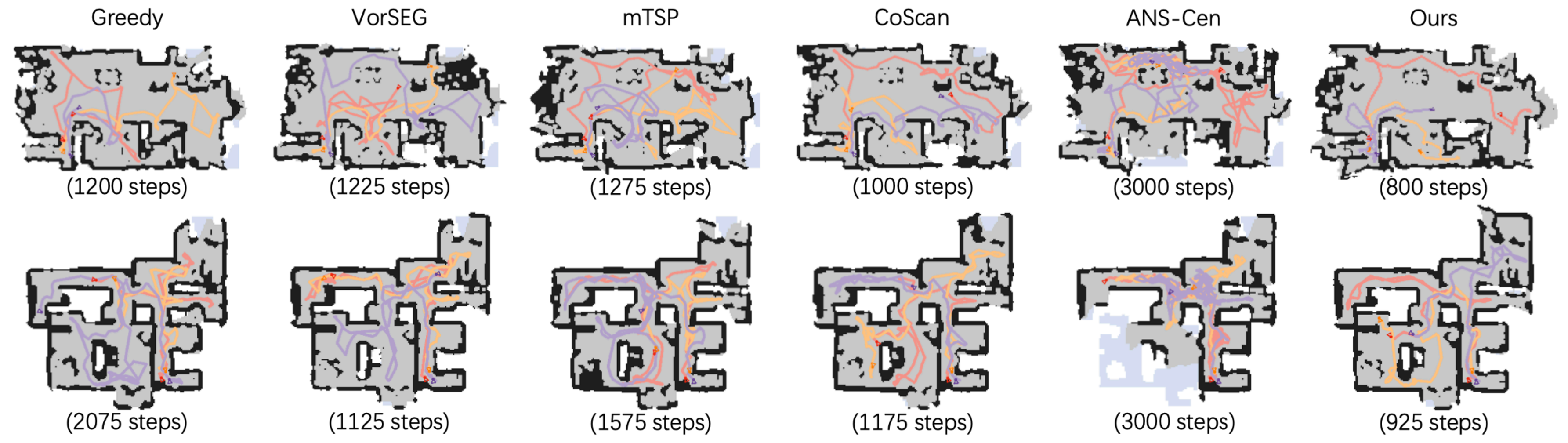


• Idea: Learning neural bipartite graph matching



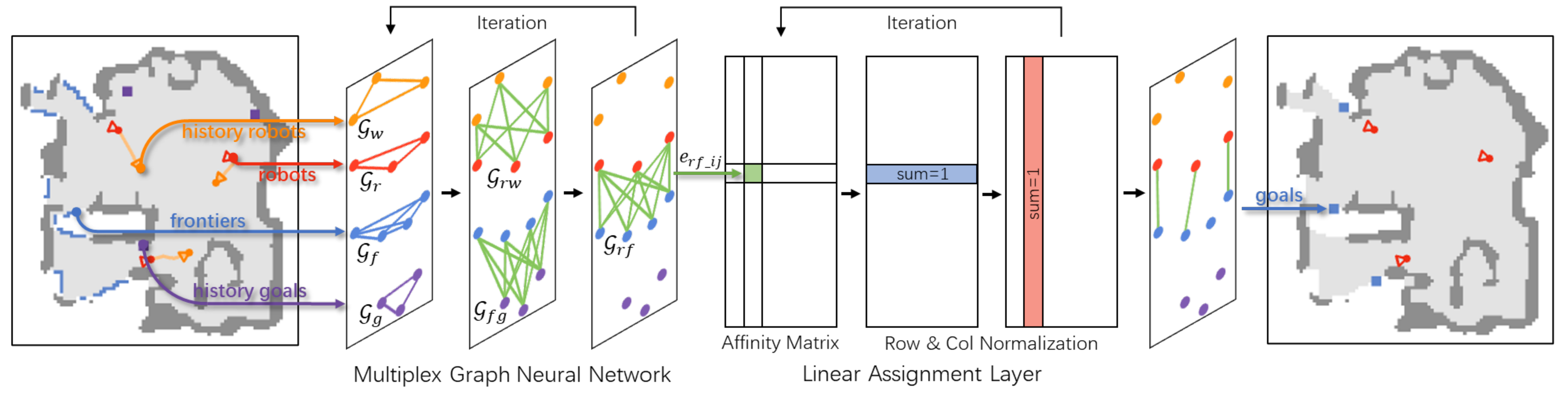
Insight: Learn representative distance metric that incorporates both history information and future prediction.

• Visual Results



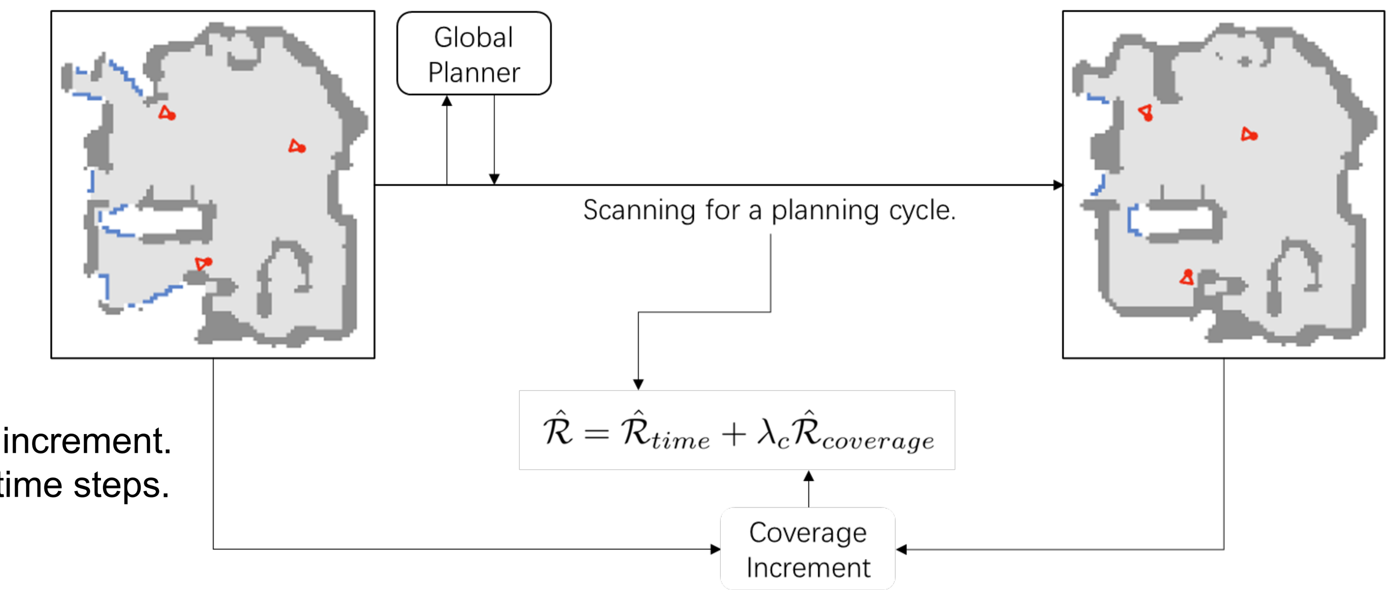
• NeuralCoMapping

- Global Planner: Multiplex Graph Neural Network (mGNN) + Linear Assignment Layer



- Training with Reinforcement Learning

Make use of long-term rewards to incorporate future predictions.



- Encourage coverage increment.
- Punish unnecessary time steps.

• Numerical Results: Our algorithm achieves more superiority in larger scenes.

	Small Scenes ($< 35m^2$)		Middle Scenes ($35-70m^2$)		Large Scenes ($> 70m^2$)	
Method	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)
Greedy [41]	98.7	420.6	98.8	669.4	99.4	1057.0
VorSEG [2]	98.5	350.8	98.8	570.3	99.6	1080.5
mTSP [19]	98.8	351.4	98.6	564.5	98.9	1029.0
CoScan [16]	98.6	304.2	99.0	496.1	99.5	985.0
NeuralCoMapping (Ours)	98.6	302.5 (-0.6%)	98.8	471.7 (-4.9%)	98.9	882.0 (-10.5%)

Table 1. Numerical results on the Gibson dataset [45]. Parentheses: %steps reduced against the best competitor (blue) for our algorithm (red).

	Small Scenes ($< 100m^2$)		Middle Scenes ($100-300m^2$)		Large Scenes ($> 300m^2$)	
Method	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)
Greedy [41]	95.9	801.2	95.1	2043.1	91.3	3345.4
VorSEG [2]	95.5	652.7	94.7	1693.2	91.0	2852.0
mTSP [19]	96.6	712.5	95.6	1742.6	91.4	2963.6
CoScan [16]	97.1	581.7	96.1	1505.5	92.1	2781.0
ANS-DeCen	85.1	1860.4	59.9	3229.8	48.5	5639.6
ANS-Cen	89.7	1536.8	64.3	2781.3	52.2	4961.3
NeuralCoMapping (Ours)	96.7	506.1 (-13.0%)	96.0	1217.3 (-19.1%)	92.4	1874.8 (-32.6%)

Table 2. Generalization to the unseen Matterport3D dataset [7], which is consistently larger than the Gibson dataset. Note our algorithm is trained only on 9 scenes in the Gibson dataset, while the ANS variants are trained on the entire Gibson dataset.

• Generalization Ability

	Train with 2 robots		Train with 3 robots		Train with 4 robots	
Test number of robots	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)
2 robots	96.7	1002.6	96.4	1005.0	97.1	1019.7
3 robots	96.7	680.0	96.3	661.7	96.1	683.7
4 robots	97.1	617.2	96.7	626.3	96.7	614.6

Generalize to different #robots

- Only on learnable module
- Robots only partially involved
- Node features updated as weighted sum of neighbors

	Train with 3 robots		Upper bound	
Test number of robots	Cov. (%)	Time (#steps)	Cov. (%)	Time (#steps)
2 robots	97.1	1293.8	96.6	1276.5
4 robots	98.4	798.7	98.4	776.3
5 robots	98.1	728.7	98.2	693.5
7 robots	96.9	694.0	98.4	662.3
9 robots	98.6	589.7	98.5	580.8

When only trained on a single scene, our method can still performs well. It further demonstrates the generalization ability of our method.

NeuralCoMapping (Ours)	Cov. (%)	Time (#steps)
single training scene	97.1	691.3
nine training scenes	96.3	661.7

• More Visual Results