香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

HQ OWL

Leveraging 360° Cameras and Coordinated Multi-Agent Systems for Scalable Indoor 3D Reconstruction

Hoi Chuen Cheng

Thesis Supervisor: Prof. Chik Patrick Yue

28th February 2024

Optical Wireless Lab (OWL) & Integrated Circuit Design Center (ICDC)
Department of Electronic and Computer Engineering (ECE)
The Hong Kong University of Science and Technology (HKUST)

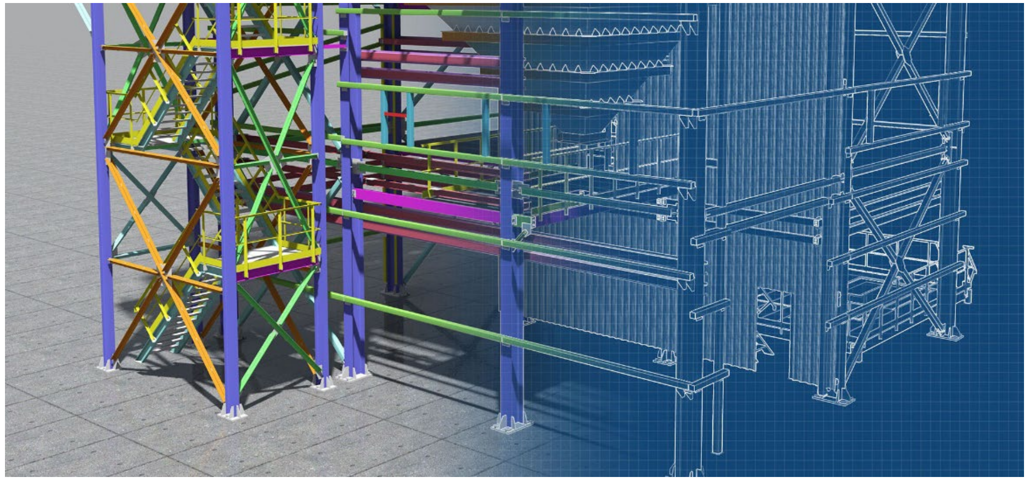
Outline

- Background and Motivations
- Leveraging 360° Cameras in 3D Reconstruction
- Optimizing Communication in Multi-Agent Path Finding
- Summary and Conclusion

Applications of 3D Reconstruction

Virtual Reality (VR)

Autonomous Driving

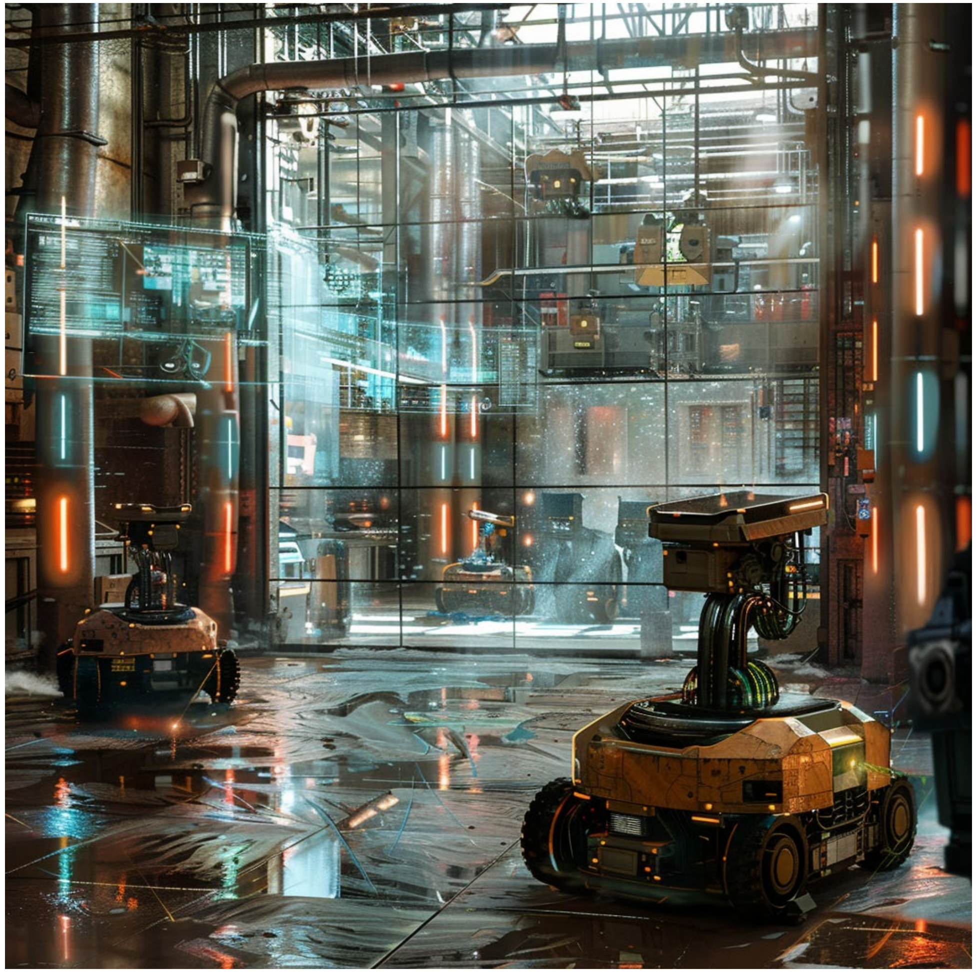


Building Information Modeling (BIM)

Motivations for Multi-Agent 3D Reconstruction

› Benefits of applying multi-agent system

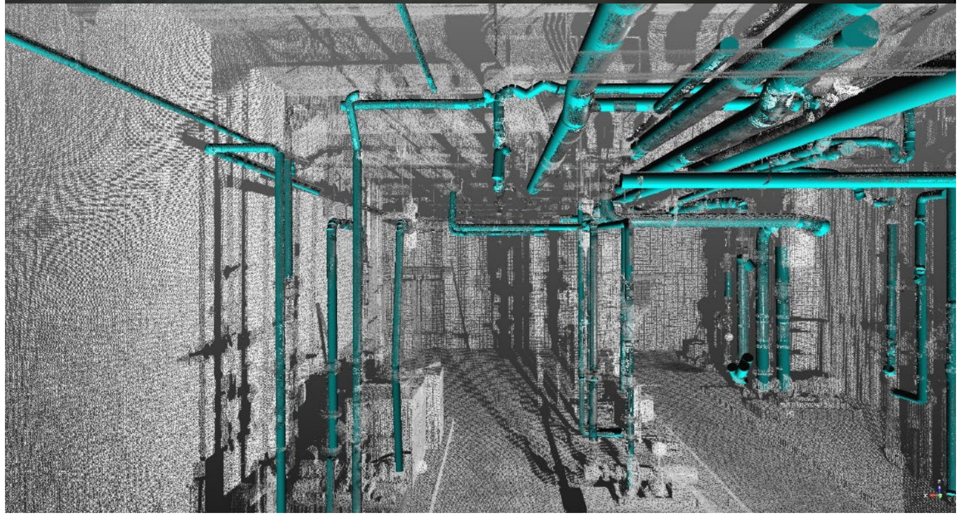
1. Increased coverage and completeness
   – Combination of multiple viewpoints
2. Efficient data acquisition
   – Saving time by parallel data collection
3. Real-time reconstructed scene update
   – Continuous processing through fleets of agents



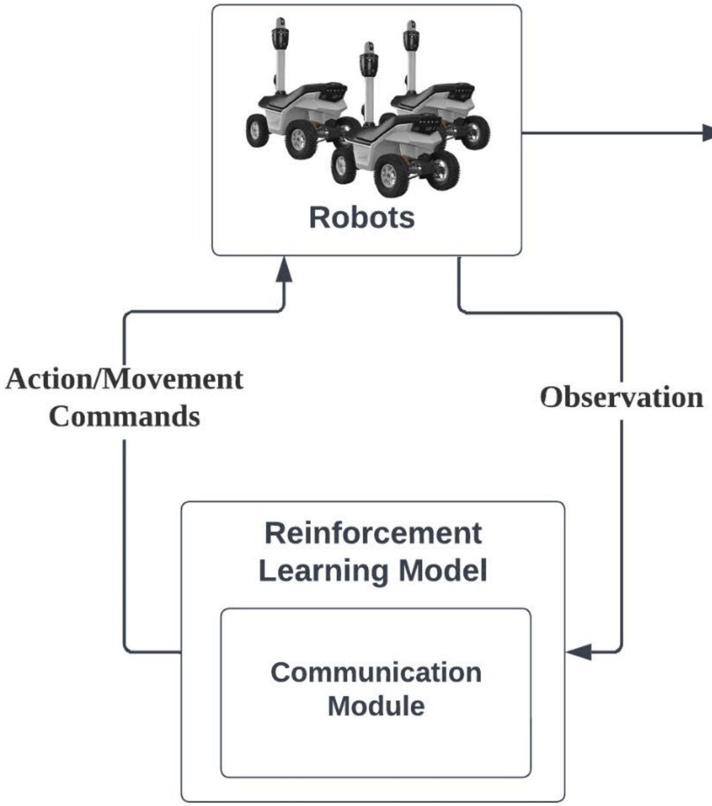
Motivations for Multi-Agent 3D Reconstruction

- In Building Information Modeling (BIM) & construction site monitoring

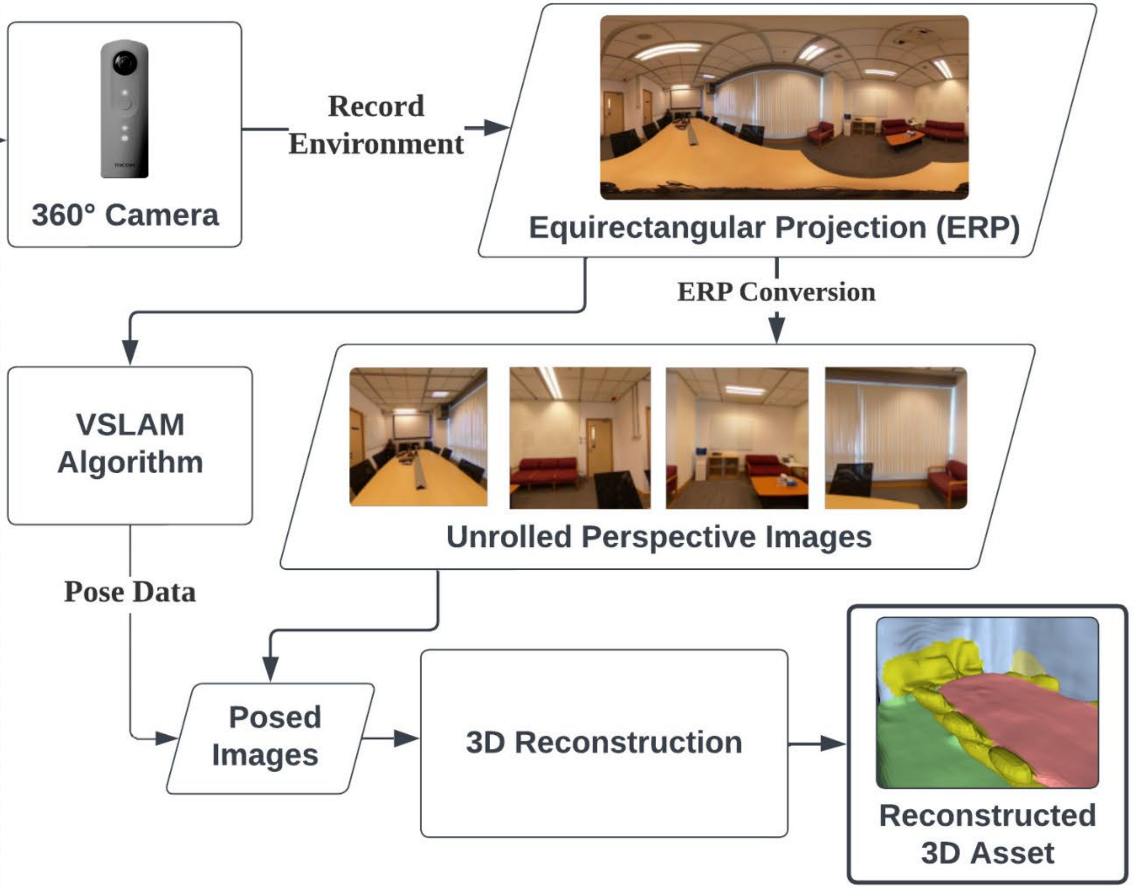
1. Increased coverage and completeness
   → Accurate documentation and design planning
2. Efficient data acquisition
   → Save significant time and cost
3. Real-time reconstructed scene update
   → Correct building errors in time



Thesis Organization and Contribution



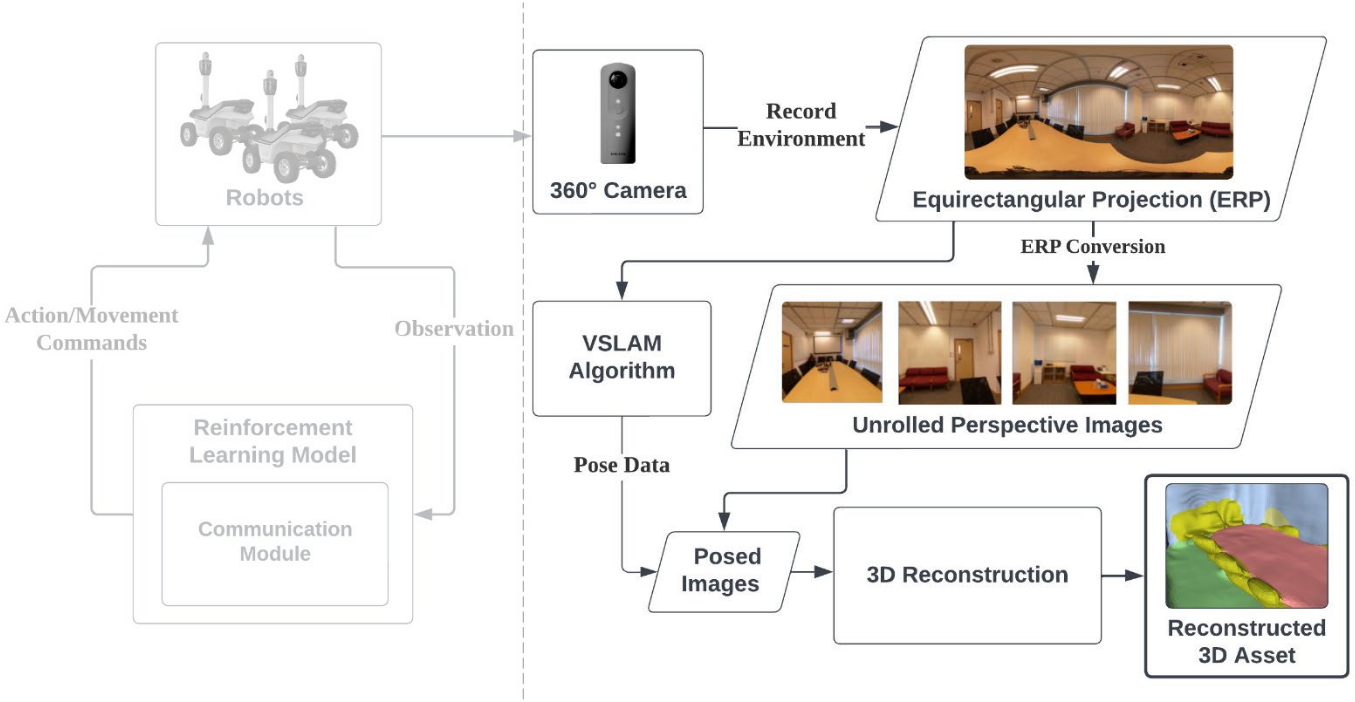
Optimizing Communication in Multi-Agent Path Finding



Leveraging 360° Cameras in 3D Reconstruction

Outline

- Background and Motivations
- Leveraging 360° Cameras in 3D Reconstruction
- Optimizing Communication in Multi-Agent Path Finding
- Summary and Conclusion



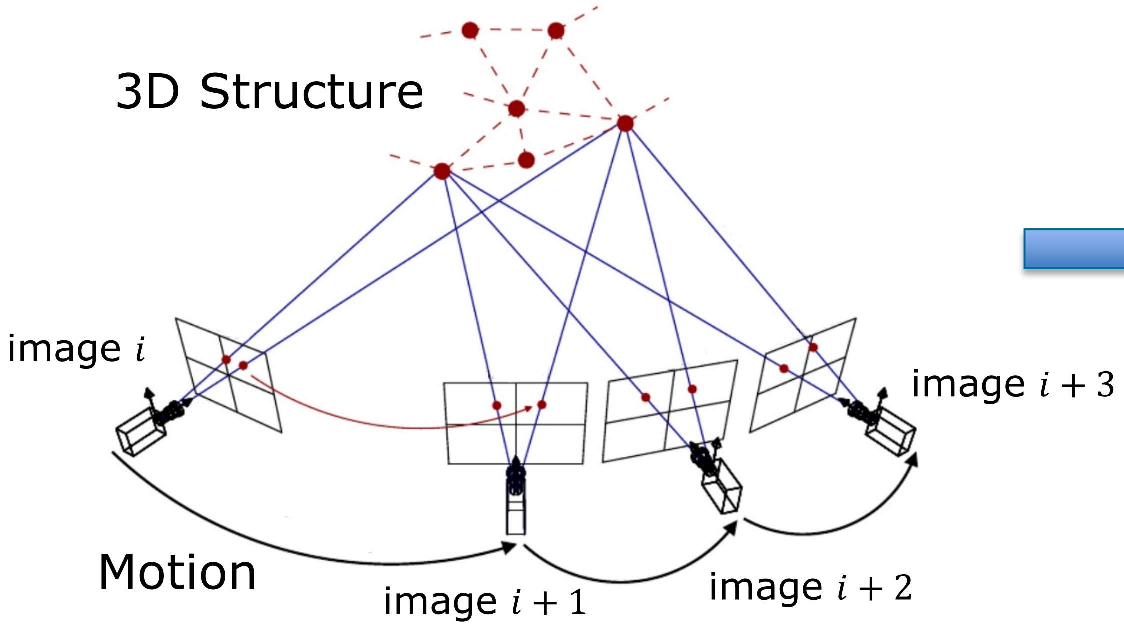
Conventional 3D Reconstruction

LiDAR Sensors

Point Cloud

- Range sensors such as LiDAR can generate 3D point clouds
  - More expensive
  - Struggle in low albedo/dark surface
  - Point clouds can be noisy

Conventional 3D Reconstruction

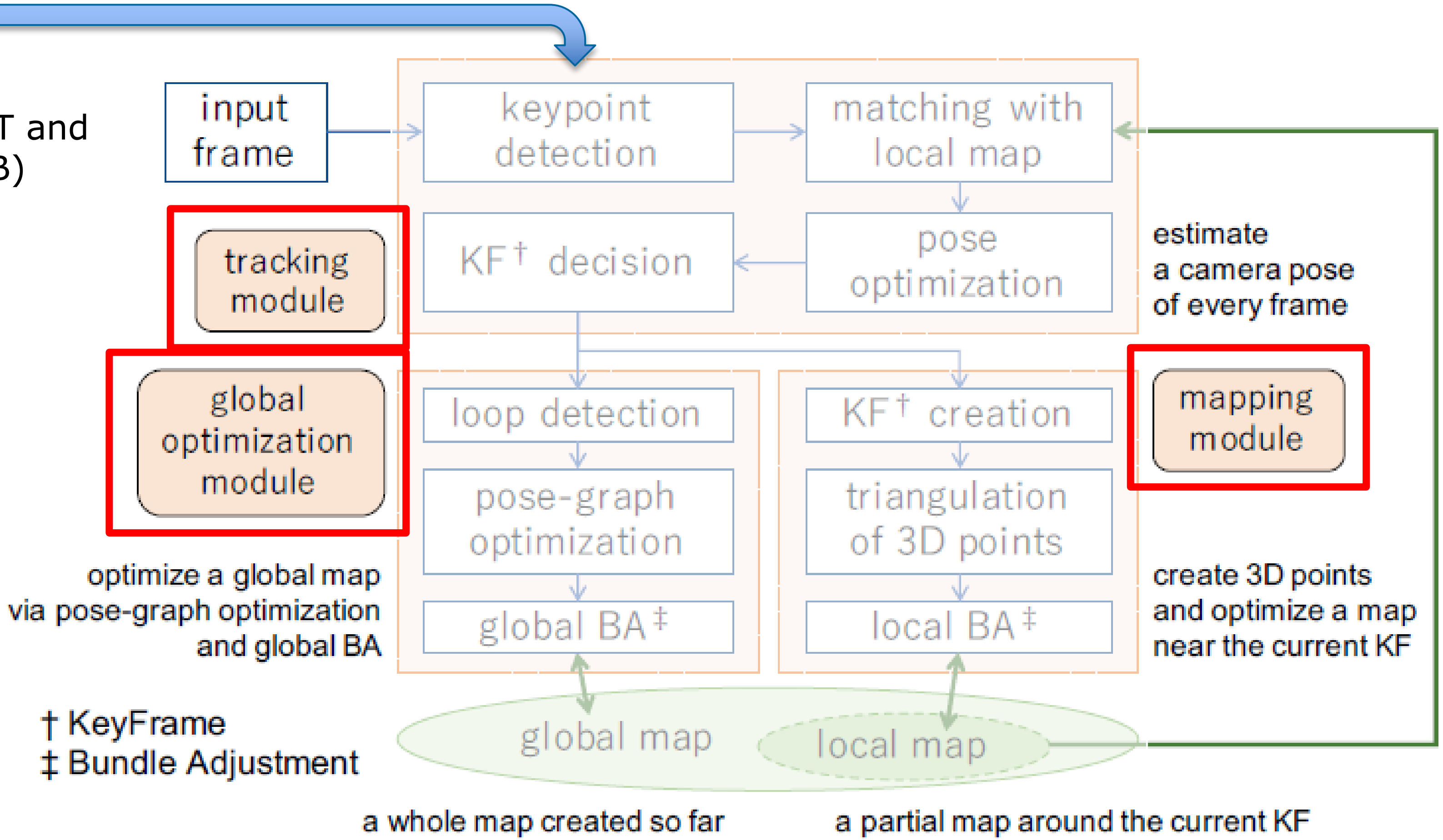


Point Cloud

- Structure from Motion
  - Tracks corresponding features across different images
  - Estimates 3D structure and camera poses from a set of 2D images
  - Sensitive to noise, sparse results, and high processing demands

Prior Works on Camera Pose Estimation

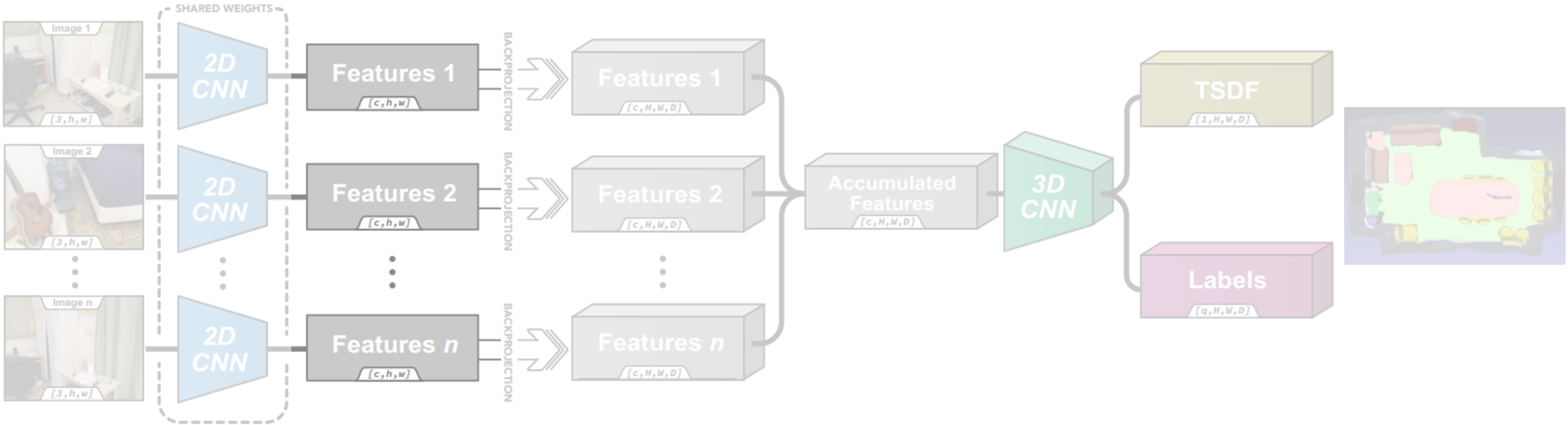
Using Oriented FAST and Rotated BRIEF (ORB) feature extractor



OpenVSLAM : A versatile visual SLAM framework
Sumikura et al., ACM MM'19

Prior Works on Learning-based 3D Reconstruction

Posed Images



Atlas: End-to-end 3D scene reconstruction from posed images
Murez et al., ECCV'20

- Extract 2D features with 2D convolutional neural network (CNN)
- Backproject 2D features into 3D features using pose data
- Refine accumulated 3D features into a 3D model

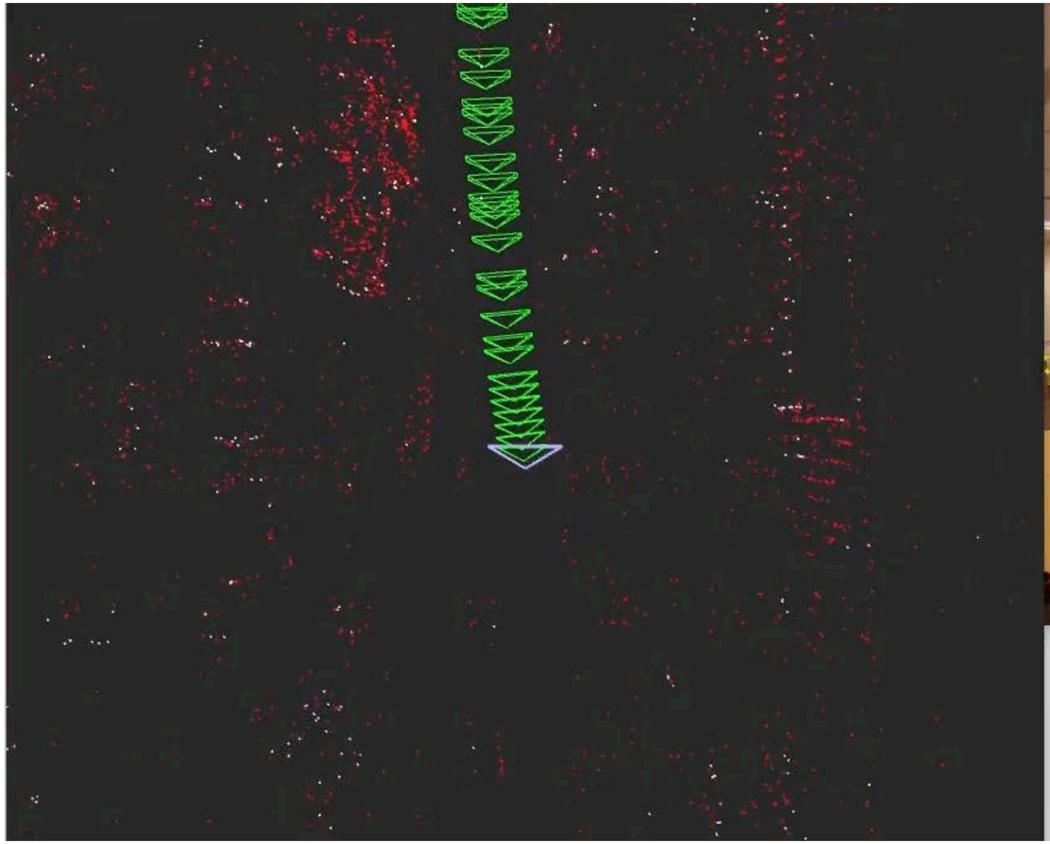
Motivations for Utilizing 360° cameras

- Commonly used for surveillance purposes
- Advantages of using 360° cameras
  - Enhanced Coverage
    - Captures full spherical view of the environment
  - Simplified Data Collection
    - No need to capture from multiple angles
- Challenges and limitations
  - Complex 360° camera calibration
  - Challenging to integrate with existing deep learning pipelines for 3D reconstruction

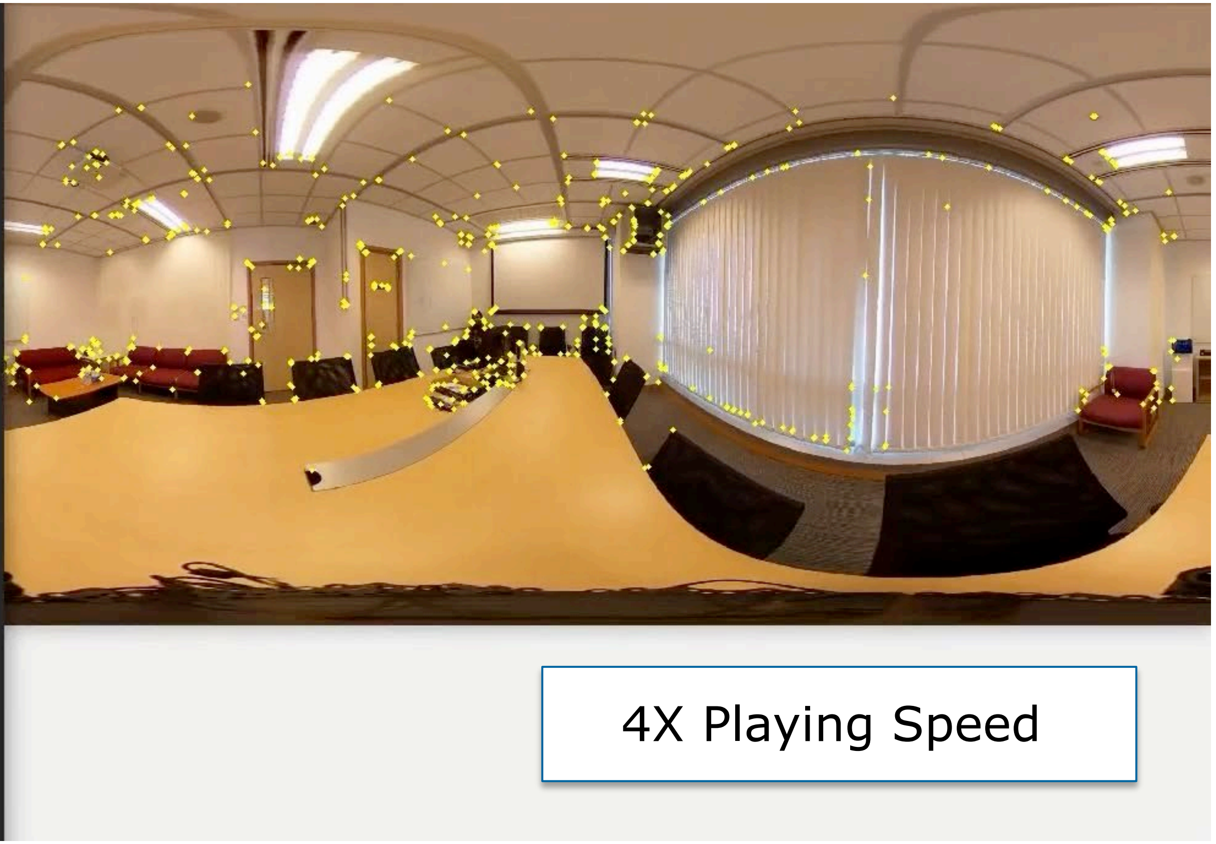
360° camera tested: Ricoh Theta V

Equirectangular Projection (ERP)

Pose Estimation



3D Point Cloud & Poses



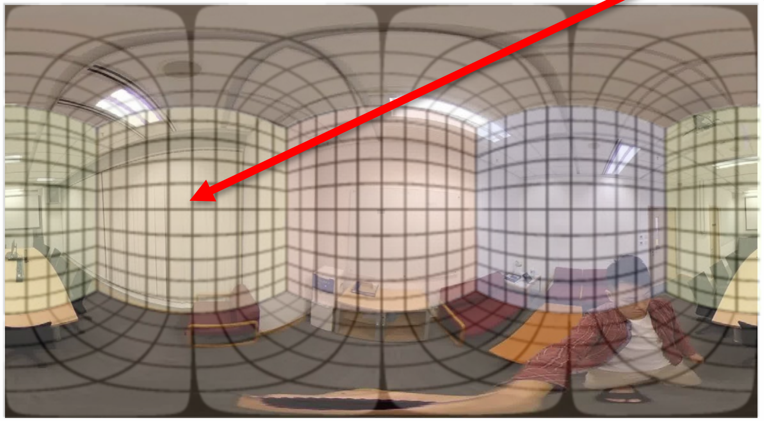
Video from 360° Camera

- Extracting poses from a 360° video using OpenVSLAM

S. Sumikura et al., “OpenVSLAM: A versatile visual SLAM framework,” in *Proceedings of the 27th ACM International Conference on Multimedia* (pp. 2292-2295), 2019.

Pose Estimation

Original pose's direction

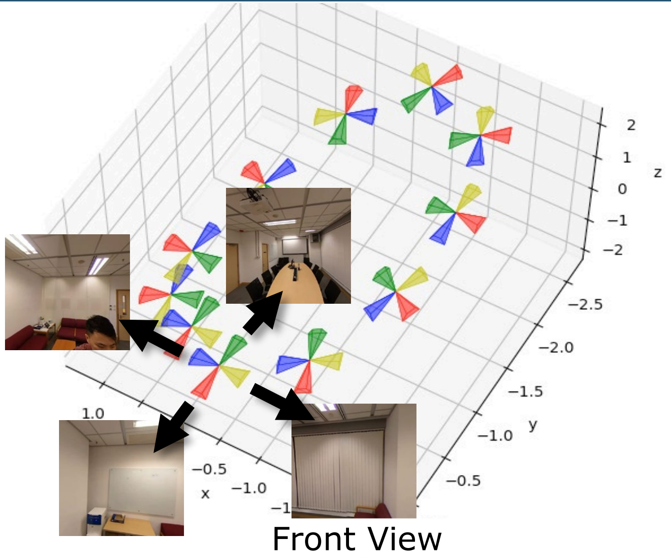


\+ ...

OpenVSLAM

Rigid Body Rotation

ERP frames

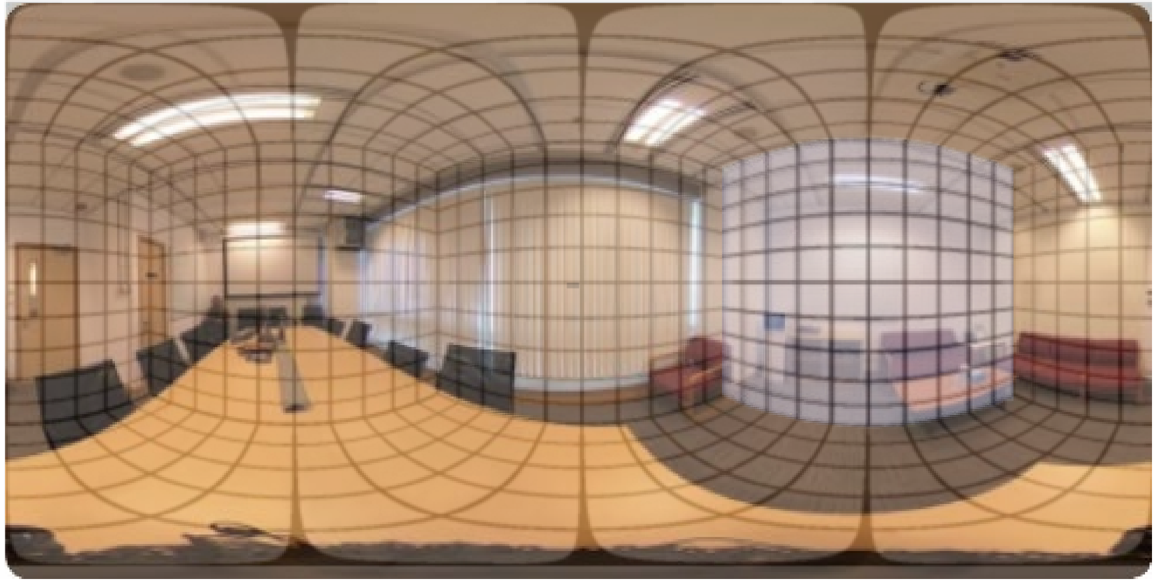


Visualized Poses

- Obtain poses of every ERP frames from OpenVSLAM
- Transform ERP's poses to 3 extra poses with 90° difference using rigid body rotation
- Each of the transformed pose correspond to a perspective image

Equirectangular Projection Conversion

- To convert an ERP to a perspective image

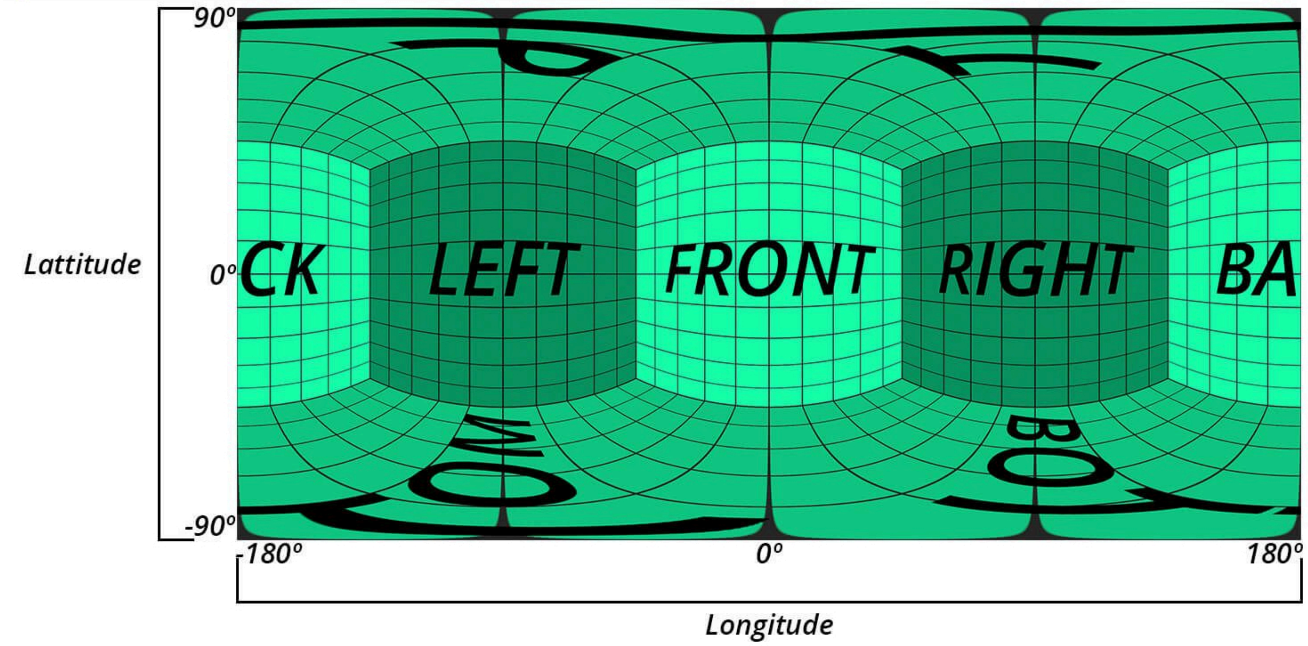


Equirectangular Projection (ERP)

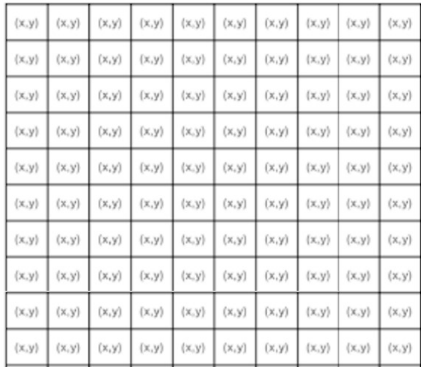
Perspective Image

Equirectangular Projection (ERP) Conversion

- To convert an ERP to a perspective image



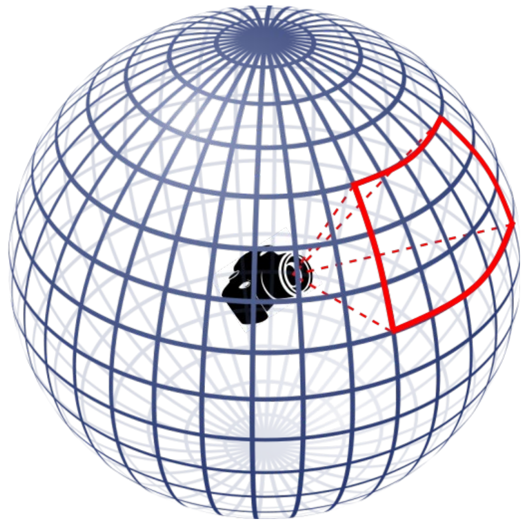
Equirectangular Projection (ERP)



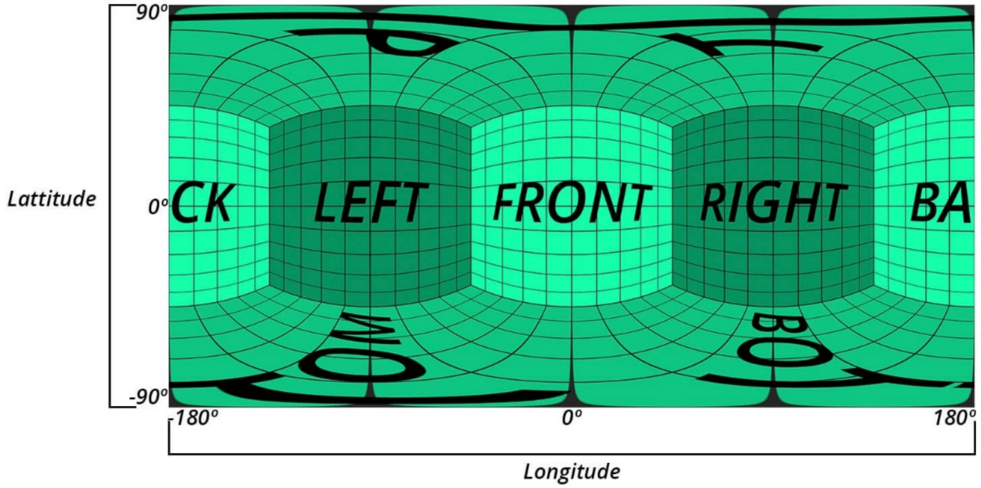
Perspective Image

Equirectangular Projection (ERP) Conversion

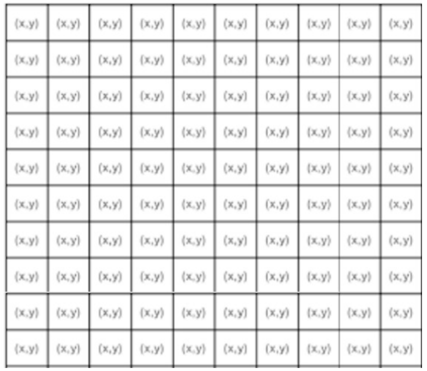
- To convert an ERP to a perspective image



3D Coordinates of Region of Interest



Equirectangular Projection (ERP)



Perspective Image

Equirectangular Projection (ERP) Conversion

1. Define viewing angle and output perspective image size (H, W)

2. Create 3D coordinates for n ERP pixels

   – $n = H \cdot W$

3. Rotate the 3D coordinates based on offset angles (θ, φ)

   – $v_r = v + \sin\alpha\,(k \times v) + (1 - \cos\alpha)k \times (k \times v)$

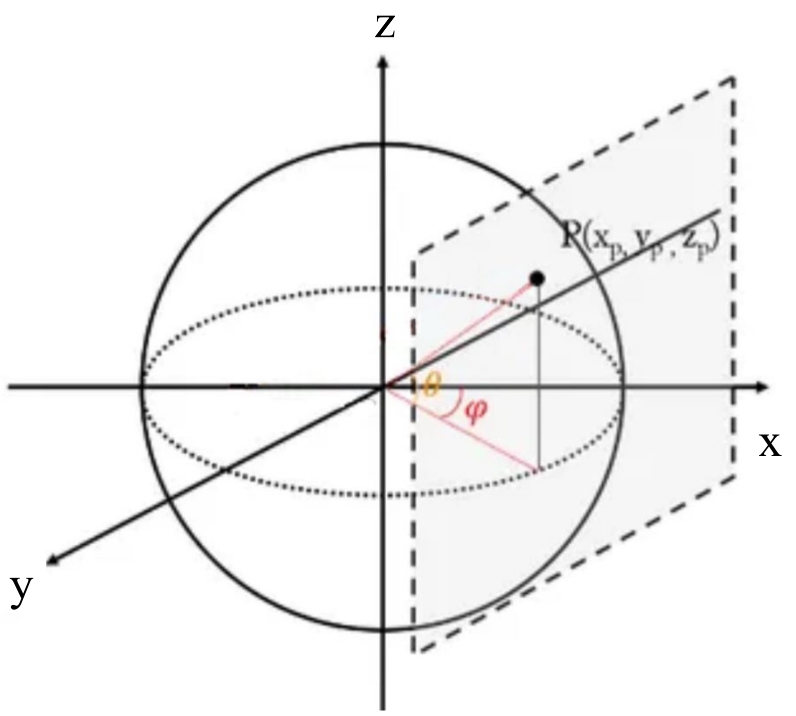
   - $v_r = Rv$

   – $R = I + (\sin\alpha)K + (1 - \cos\alpha)K^2$

   – $K = \begin{bmatrix} 0 & -k_z & k_y \\ k_z & 0 & -k_x \\ -k_y & k_x & 0 \end{bmatrix}$

   - Representation of rotation axis

   – Example: Unit vector's representation along z-axis $= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$



Equirectangular Projection (ERP) Conversion

4. Convert 3D coordinates to latitude and longitude

$$latitude = \sin^{-1}\frac{z}{r}$$

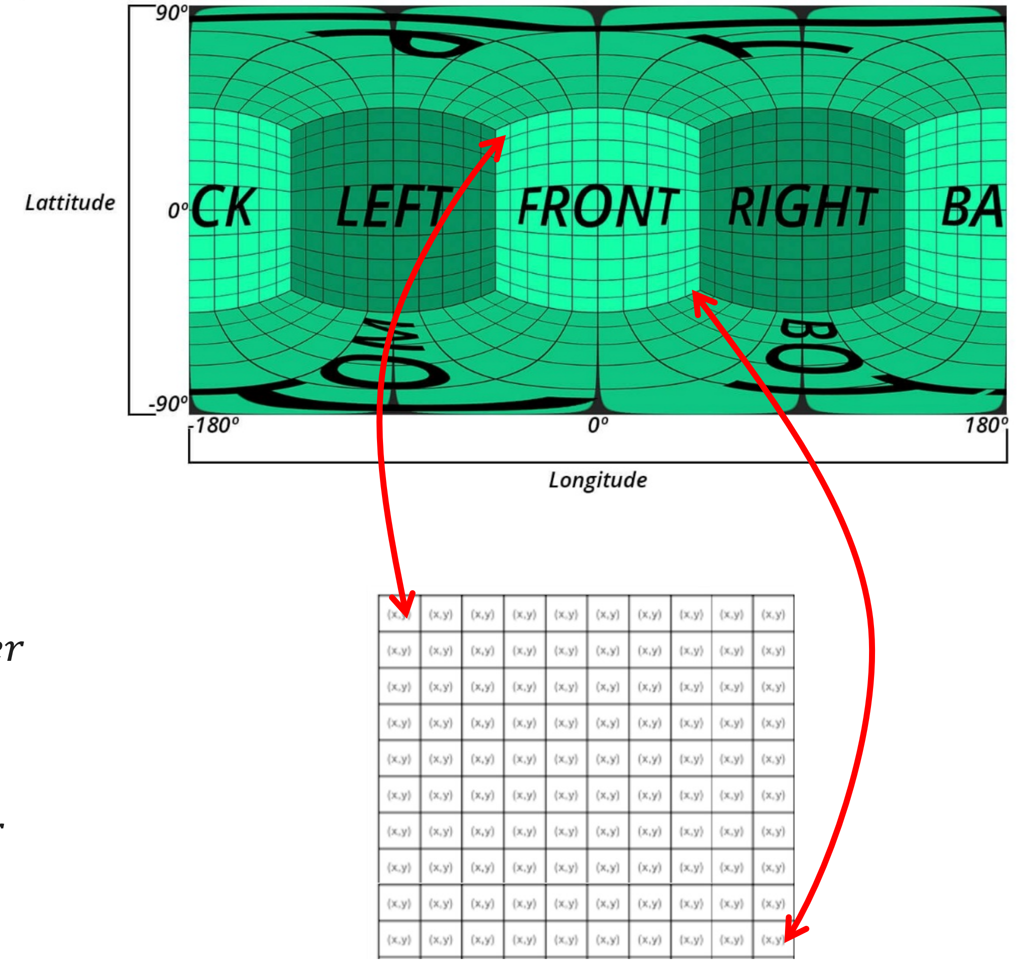
$$longitude = \tan\frac{y}{x}$$

5. Obtain ERP pixels based on latitude and longitude

$$x_{ERP} = \frac{longitude}{180} \cdot x_{ERP\_center} + x_{ERP\_center}$$

$$y_{ERP} = \frac{latitude}{90} \cdot y_{ERP\_center} + y_{ERP\_center}$$

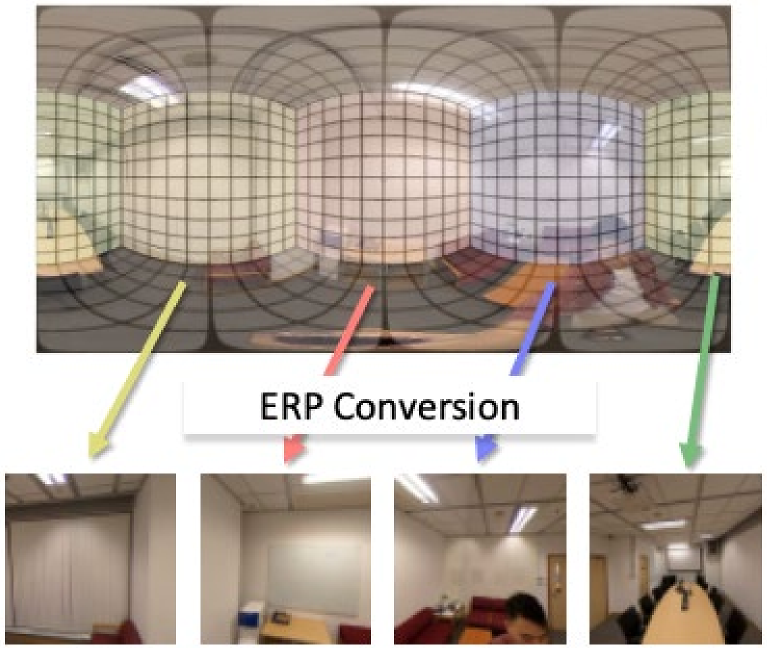
6. Remap the ERP pixels to the perspective image



Perspective Image

Equirectangular Projection (ERP) Conversion

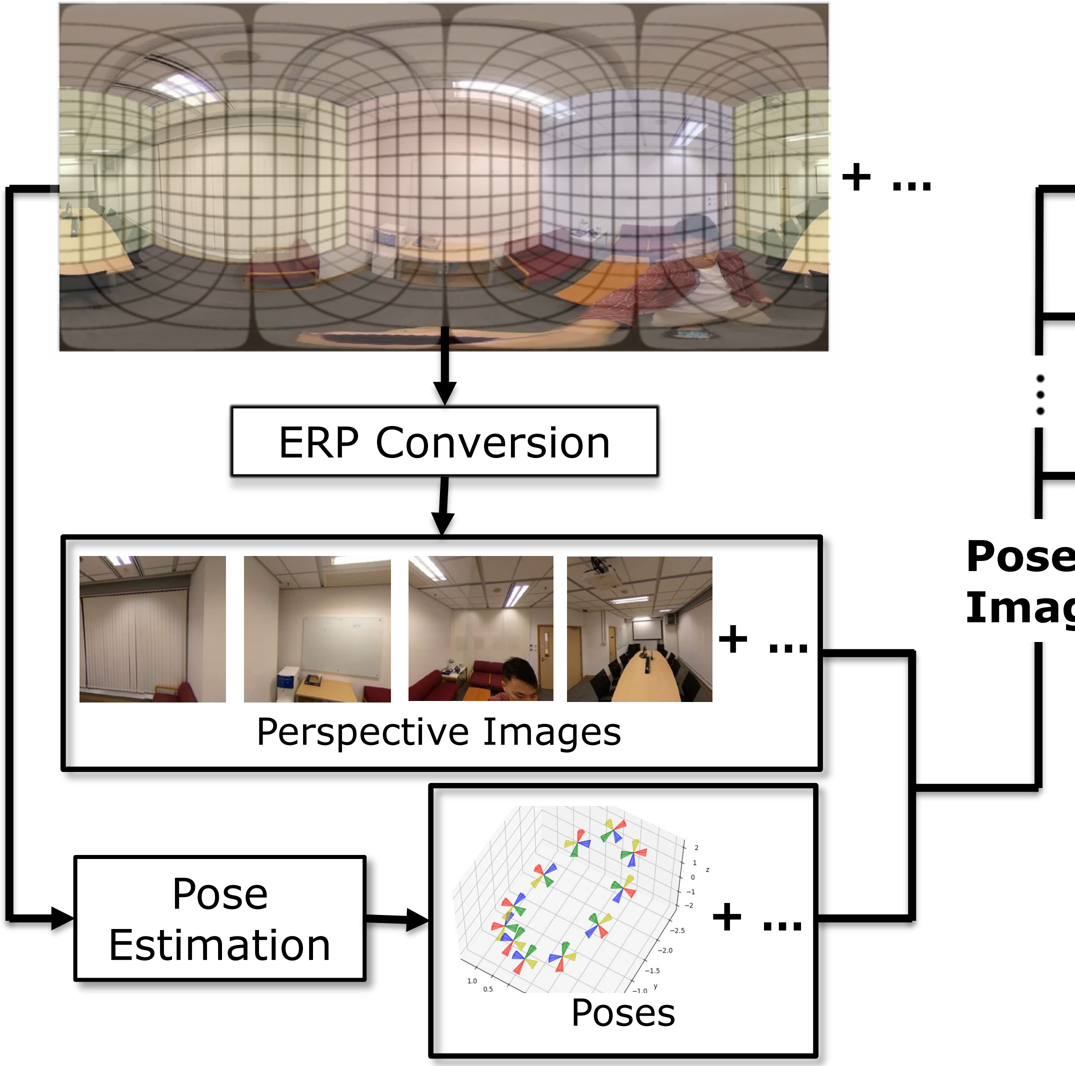
Equirectangular Projection (ERP)

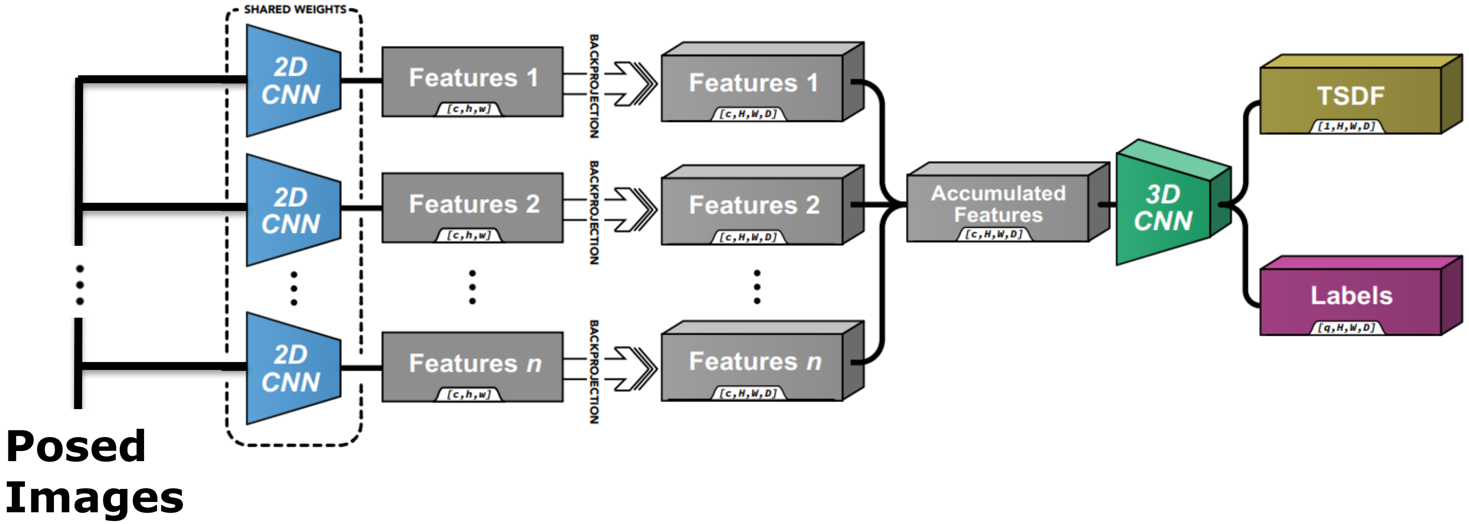


4 Perspective Images

- Convert an ERP into 4 perspective images
- Treat the 4 views as part of cube-maps, resembling four virtual cameras of field of view of 90°, pointing in 4 directions
- The final perspective images are compatible with established deep learning pipelines

3D Reconstruction Pipeline





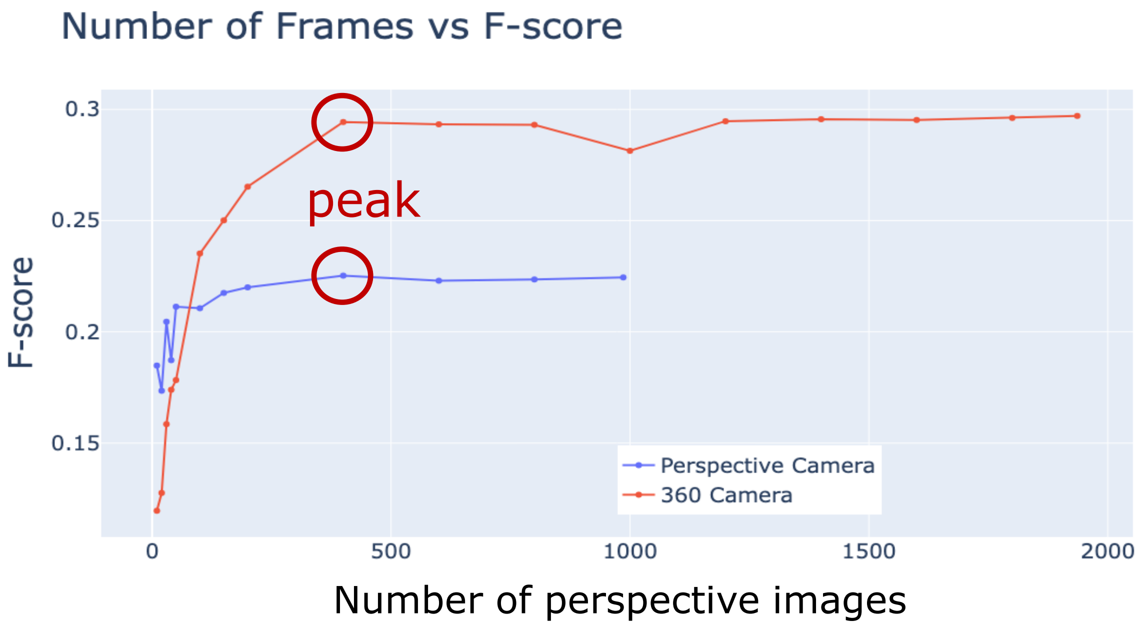
- Input of 3D reconstruction framework (Atlas)
  1. Perspective images
  2. Perspective images' poses

Qualitative 3D Reconstruction Results

Ground Truth (with texture) Perspective Camera 360° Camera

- 3D semantics comparison

Data Collection Efficiency



- Finding the least amount of posed images needed for best performance
  1. 360° camera: 400 perspective images → **100 ERP**
  2. Perspective Camera: 400 perspective images
- Total images captured
  1. 360° camera: 500 ERP
  2. Perspective Camera: 1000 perspective images

Quantitative 3D Reconstruction Results

Input for 3D Reconstruction		F-score (%)	Estimated frames per m^2 needed to attain best performance
Image Data Source	**Pose Data Source**		
360° camera	OpenVSLAM	29.7	3.34
Perspective camera	OpenVSLAM	22.5	13.36

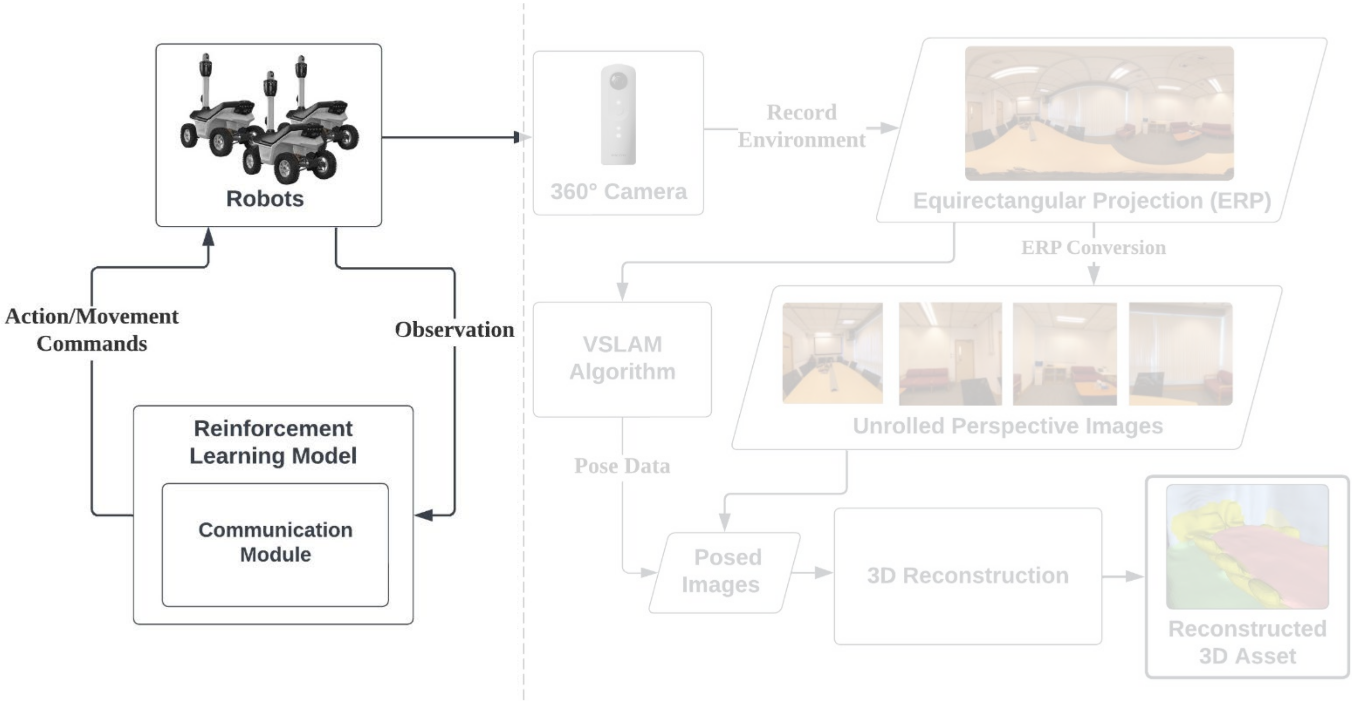
More Accurate

More Efficient

- 360° camera vs Perspective camera
  - 32% more accurate than perspective camera
  - 4 times more efficient in data collection
- Test environment size: $\sim 30m^2$

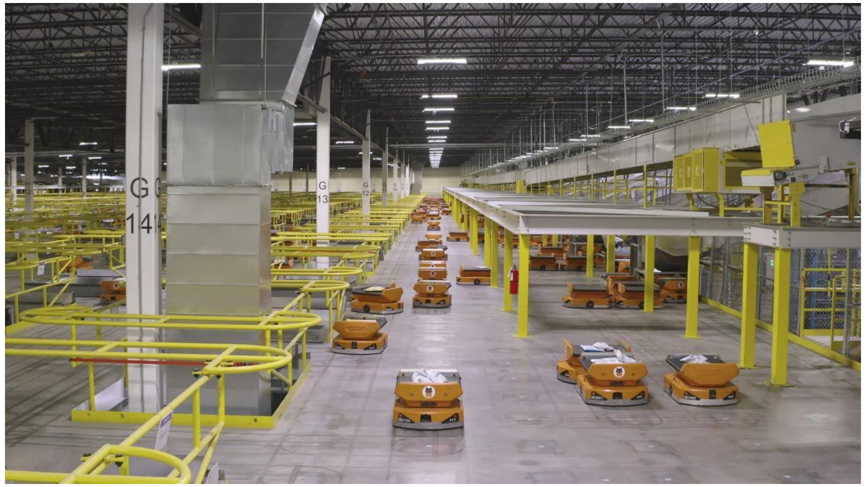
Outline

- Background and Motivations
- Leveraging 360° Cameras in 3D Reconstruction
- Optimizing Communication in Multi-Agent Path Finding
- Summary and Conclusion



Optimizing Communication in Multi-Agent Path Finding

- The Rise of Autonomous Systems:
  - Surveillance or monitoring robots
  - Autonomous vehicles
  - Warehouse robots



- The robots will soon operate in large numbers
- Scalable Multi-agent Path Finding (MAPF) relies on decentralized decision-making
- In a partially observable environment, agents need communication to
  - Coordinate actions
  - Share information

Problem Formulation of Multi-Agent Path Finding

- Given a graph $G = (V, E)$, with start and destination vertices as $\{s_1, \dots, s_n\} \in V$ and $\{d_1, \dots, d_n\} \in V$
  - The location change $v \to v'$ caused by an agent's movement corresponds to an edge in the graph (i.e., $v \to v' \in E$)
- Denote a sequence of actions taken by agent i from the beginning to time t as $\pi_i = \{a_1, \dots, a_t\}$
- The MAPF solution $\pi = \{\pi_1, \dots, \pi_n\}$ comprises all actions from n agents
- Forbidden conflicts
  - Vertex collision: agent i and j stays on the same location simultaneously $l_i(t) = l_j(t)$
  - Edge collision: agent i and j try to move on the same edge $(v \to v' \in E)$

R. Stern et al., "Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks," in Symposium on Combinatorial Search, pp. 151–159, 2019.

Prior Works on MAPF via Reinforcement Learning

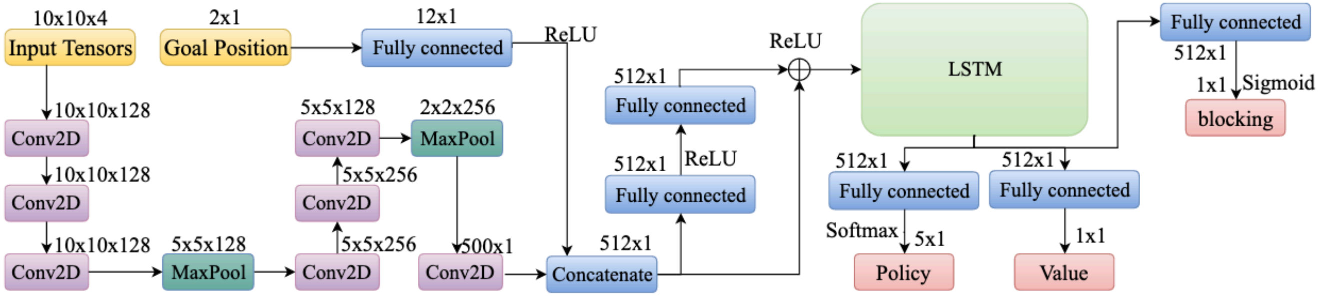
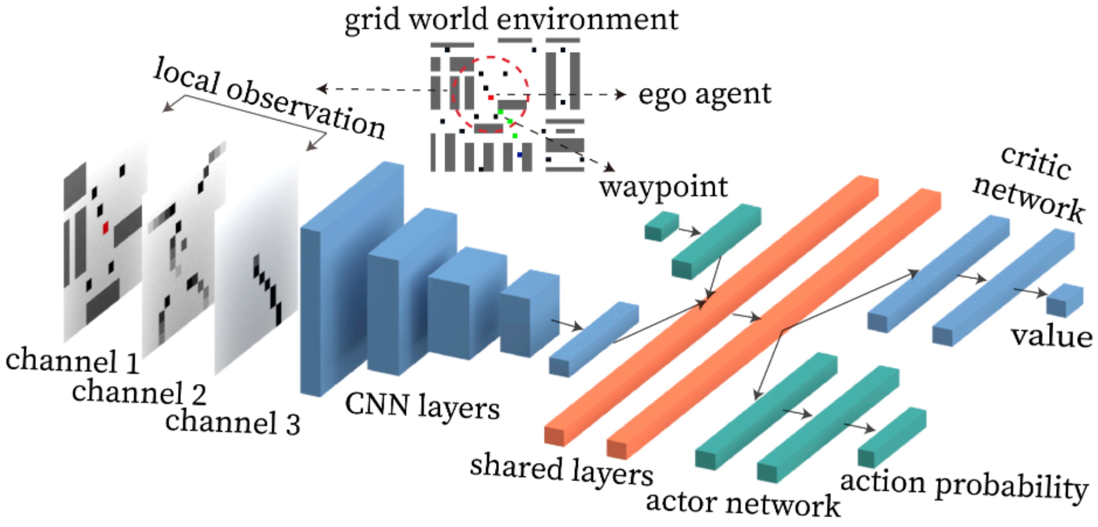


Fig. 3. The neural network consists of 7 convolutional layers interleaved with maxpooling layers, followed by an LSTM.

PRIMAL, Sartoretti et al., RA-L'19

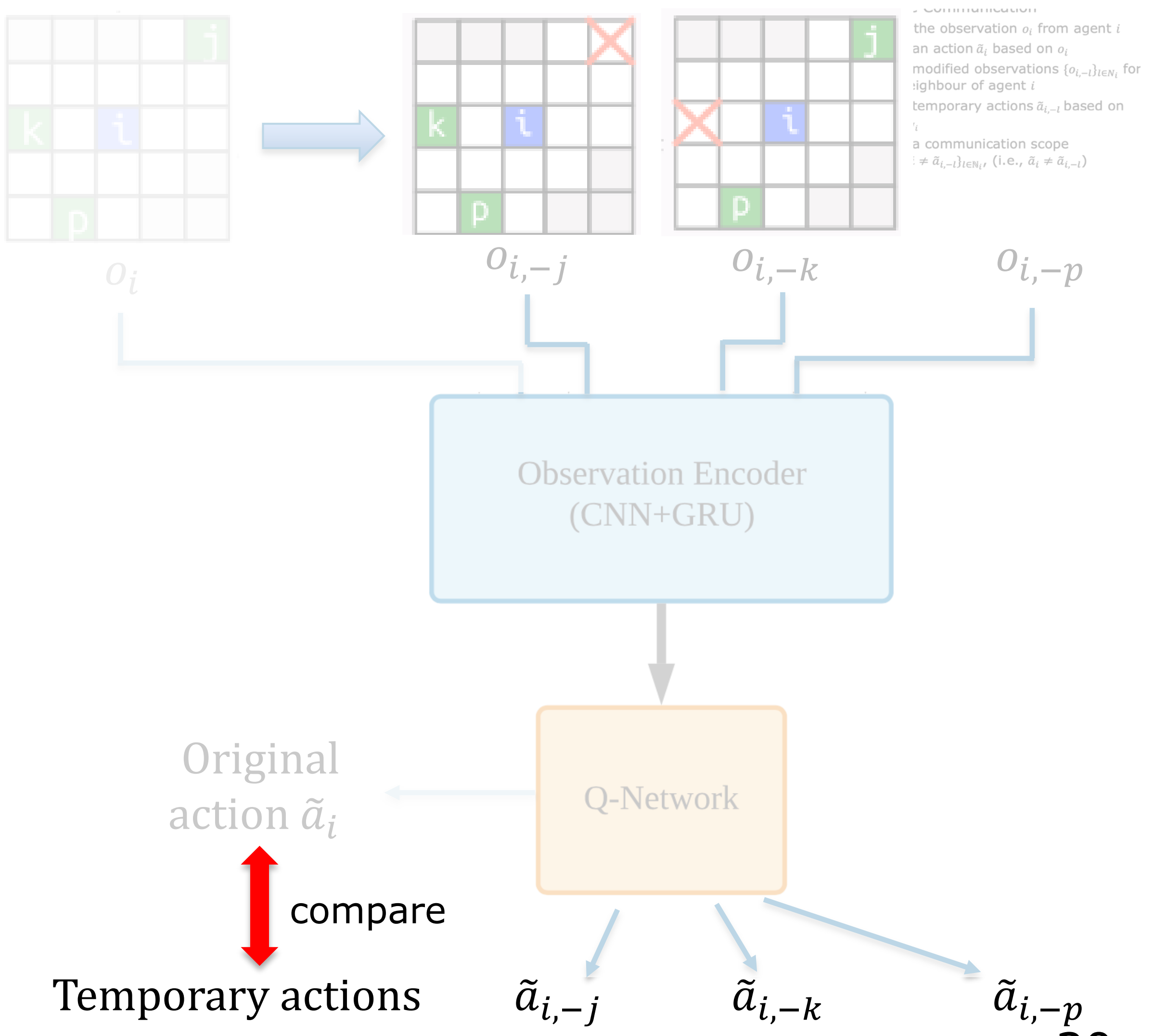


MAPPER, Liu et al., IROS'20

- Limitations
  - Heavily rely on expert algorithms
  - No **communication** involved

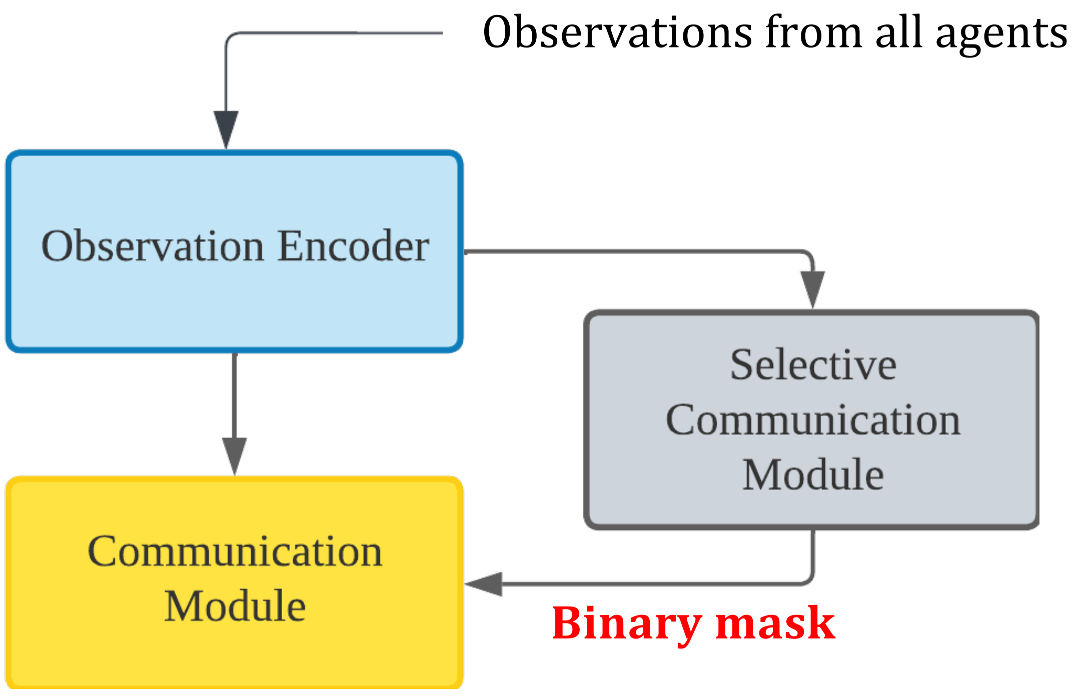
Selective Communication

- Selective Communication
  1. Gather the observation o_i from agent i
  2. Create an action $\tilde{a}_i$ based on o_i
  3. Create modified observations $\{o_{i,-l}\}_{l \in N_i}$ for each neighbour of agent i
  4. Create temporary actions $\tilde{a}_{i,-l}$ based on $\{o_{i,-l}\}_{l \in N_i}$
  5. Create a communication scope $\mathbb{C}_i = \{l | \tilde{a} \neq \tilde{a}_{i,-l}\}_{l \in \mathbb{N}_i}$, (i.e., $\tilde{a}_i \neq \tilde{a}_{i,-l}$)



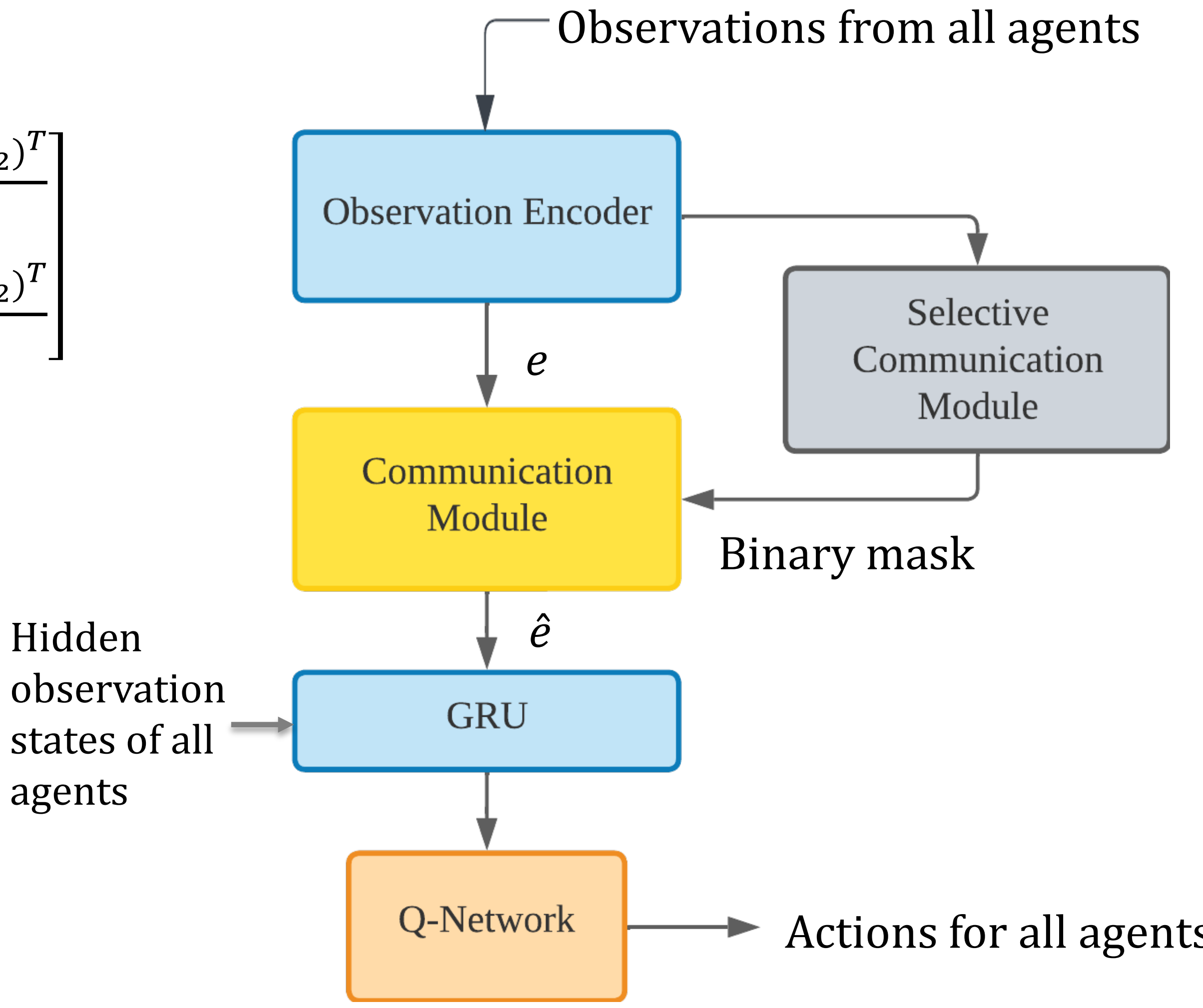
Binary Mask of Communication Scope

- i affects j & j does not affect i
  - binary mask = $\begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{1} & 1 \end{bmatrix}$
- i & j does not affect each other
  - binary mask = $\begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}$
- i & j affect each other
  - binary mask = $\begin{bmatrix} 1 & \mathbf{1} \\ \mathbf{1} & 1 \end{bmatrix}$
- $Masked\ information = information \cdot binary\ mask$

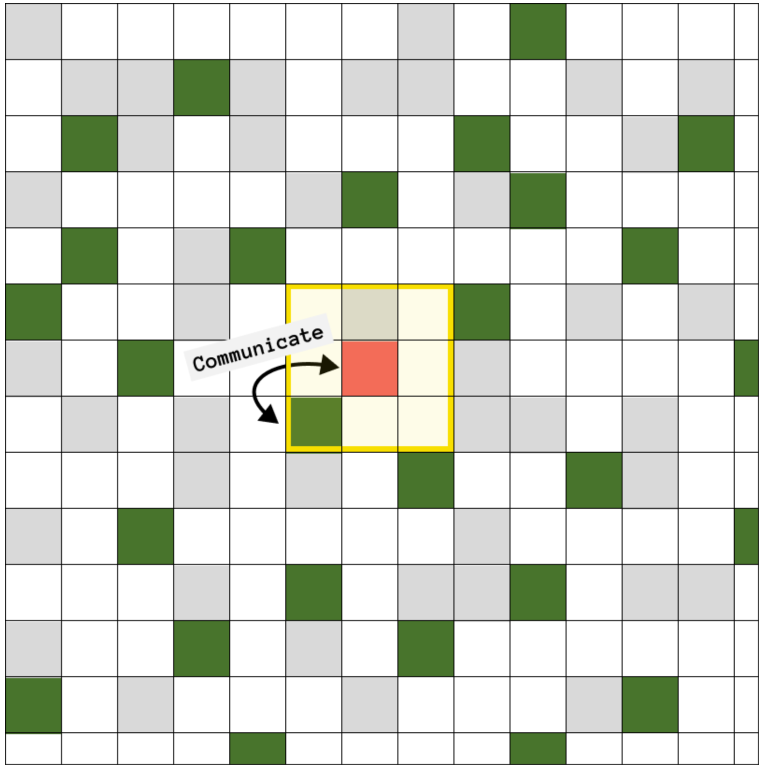


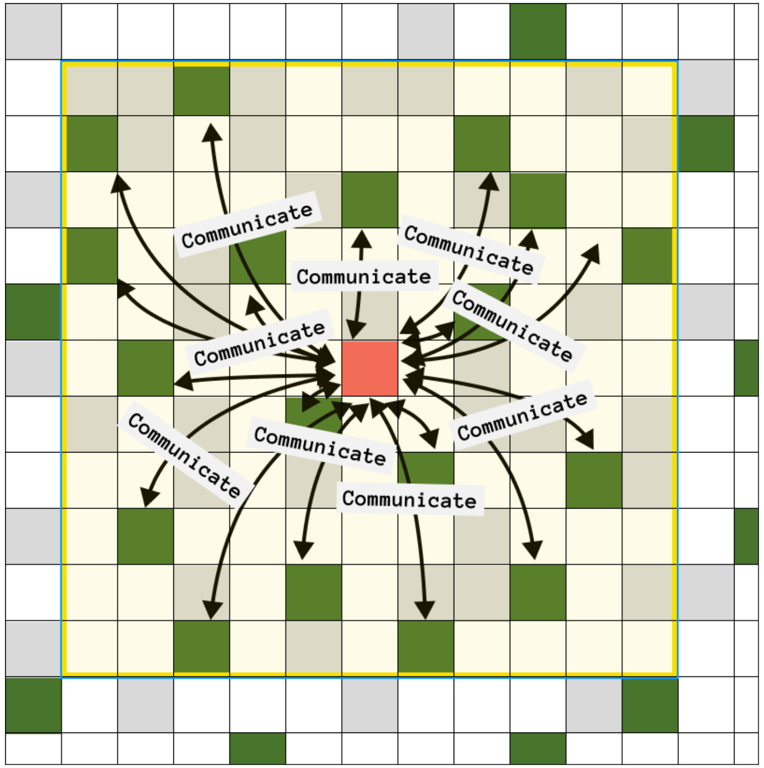
Communication Module

- Calculating attention scores across all agents
  - Attention score $\mu^h = softmax\begin{bmatrix} \frac{W_Q^h e_1 (W_K^h e_1)^T}{\sqrt{d_K}} & \frac{W_Q^h e_1 (W_K^h e_2)^T}{\sqrt{d_K}} \\ \frac{W_Q^h e_2 (W_K^h e_1)^T}{\sqrt{d_K}} & \frac{W_Q^h e_2 (W_K^h e_2)^T}{\sqrt{d_K}} \end{bmatrix}$
  - e : observation embeddings matrix of all agents
- Masked attention score
  $= attention\ scores \cdot binary\ mask$
- Embeddings from communication module
  - Attention weight $w^h = \mu^h W_V^h e$
  - $\hat{e} = f_o\left[concat\left[\sum w^h, \forall h \in H\right]\right]$
  - f_o: fully connected layer



Optimizing Field-of-View





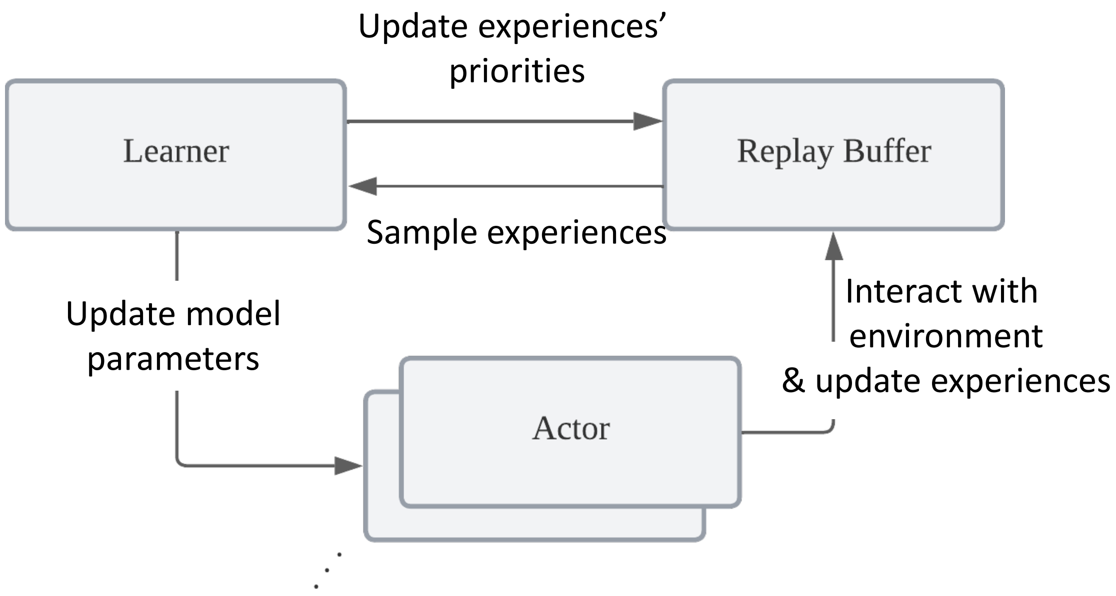
- Growing trend to use attention mechanism to explain relationship and share information between agents
- A larger Field-of-View (FOV) might have more redundant information

Motivation of Studying Field-of-View

- Field-of-View (FOV) represents
  - Perception range
    - Example: LiDAR sensors in autonomous robots
  - Communication scope
- Importance of studying FOV
  - Balancing performance and computational efficiency
  - Enhancing real-world deployment of multi-agent systems under resource constraints
- Current research landscape
  - Often neglected in reinforcement learning based MAPF studies
  - Many research studies use the same FOV (9×9) without extensive exploration

Experiment

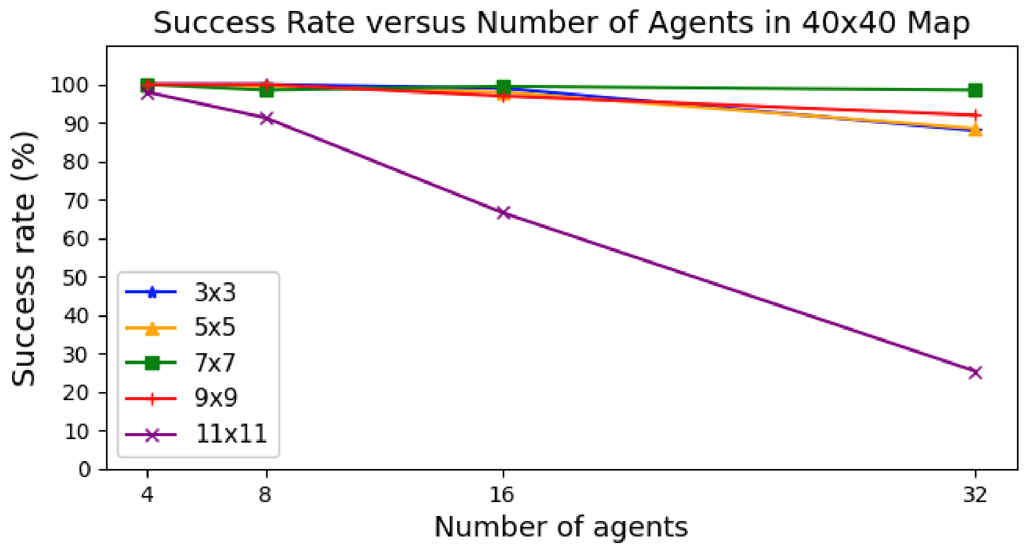
- Curriculum training
  - 5 models with FOV: $\{3\times3, 5\times5, 7\times7, 9\times9, 11\times11\}$
  - Starts with 2 agents and map size of 10×10
  - Up to 20 agents and max map size of 40×40
- Testing
  - FOV: $\{3\times3, 5\times5, 7\times7, \mathbf{9\times9}, 11\times11\}$
  - Number of agents: $\{4, 8, 16, 32\}$
  - Maps: $\{40\times40, 80\times80\}$
  - Max. steps allowed: 256
- Trained on HKUST HPC3 with Ape-X framework
  - 2 RTX 6000 GPUs and 16 Intel Xeon Gold 6230 CPUs

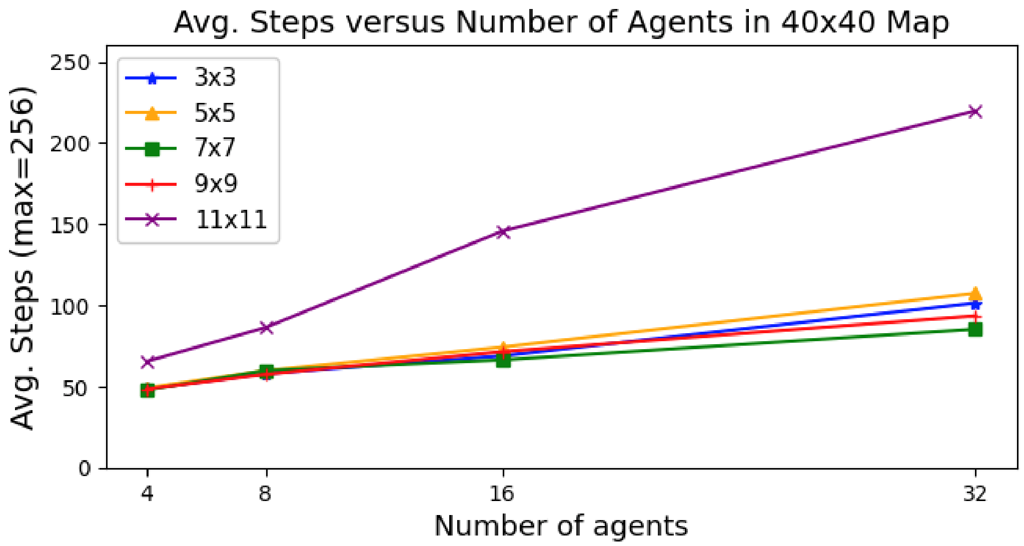


Ape-X framework
Horgan et al., ICLR'18

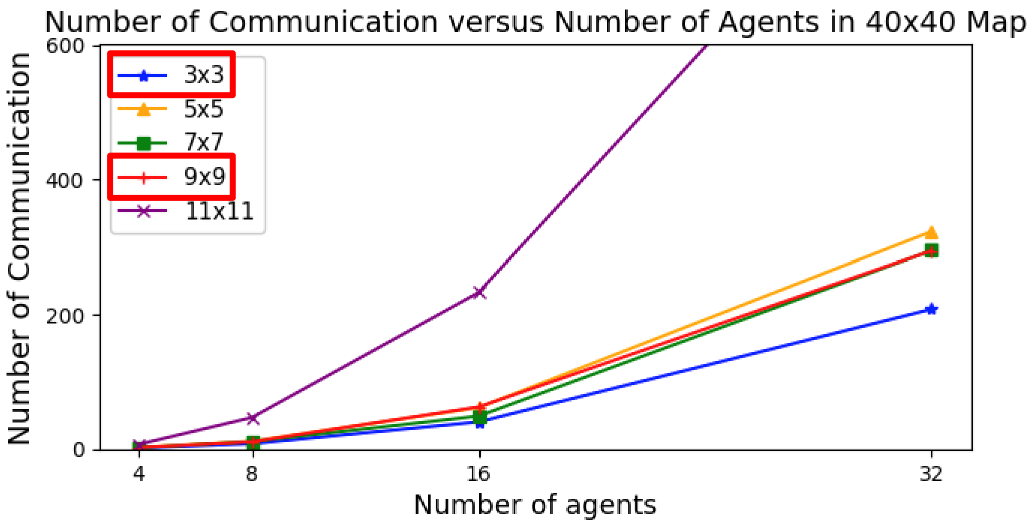
MAPF Performance

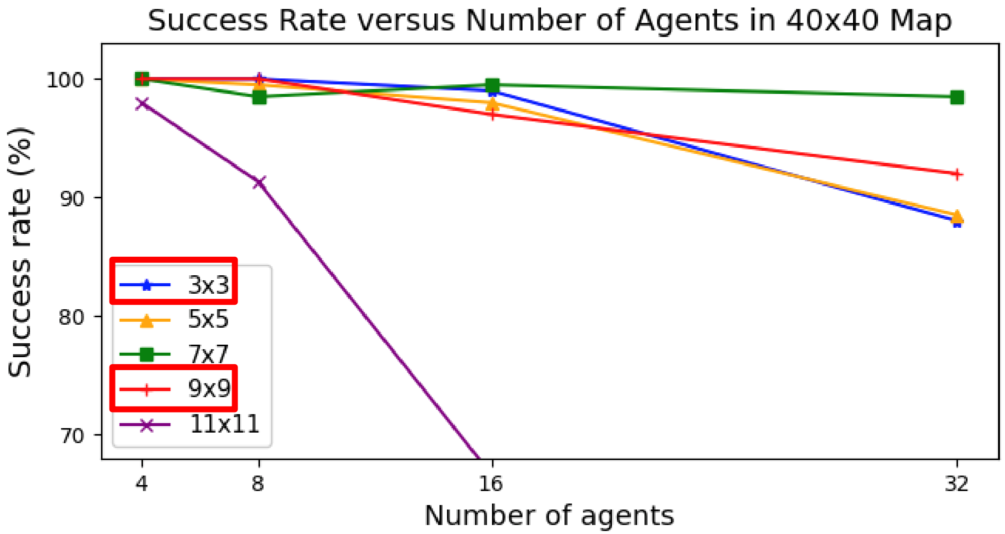
- Metrics
  - Success rate
    - Percentage of agents reaching the destination within the maximum number of steps
  - Average steps
    - Steps number needed to finish MAPF tasks across all agents
- The 7×7 FOV
  - Outperforms the baseline in both success rate and average steps
- The 3×3 FOV
  - With the least amount of information
  - Comparable performance with the baseline





Communication Overhead

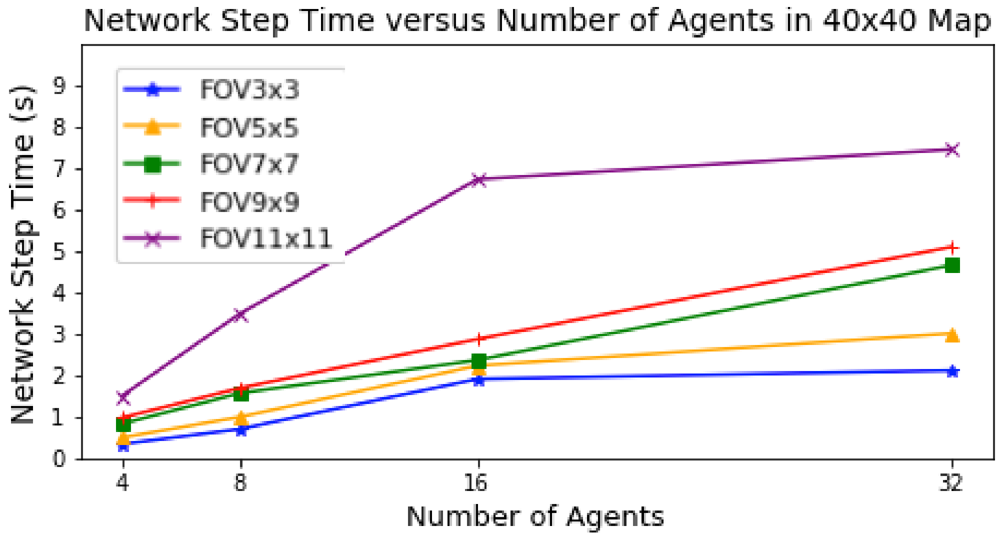


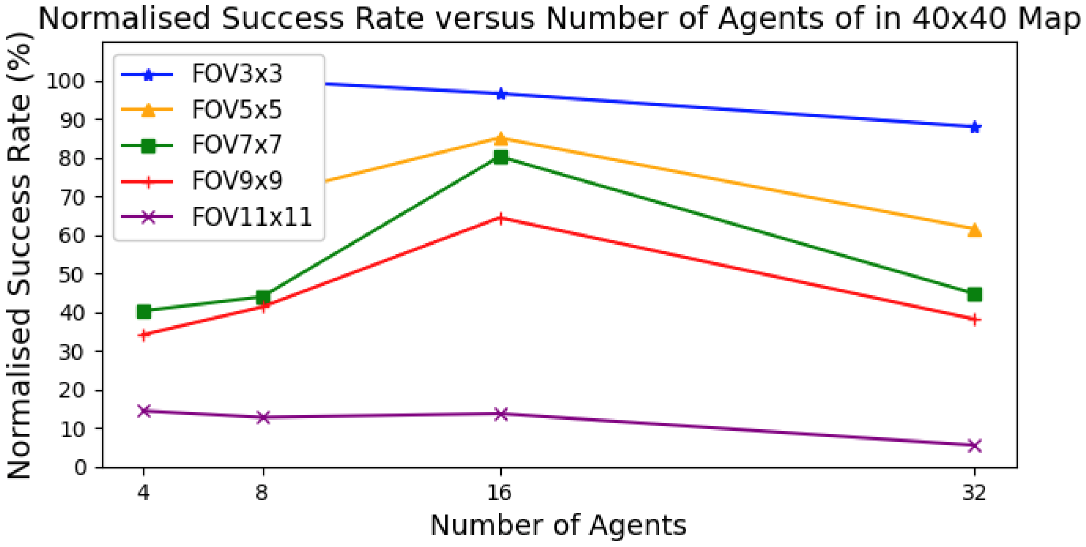


- Smallest FOV (3×3) reduced communication by 28.9% with only 1.7% decrease in average success rate compared to baseline (9×9)
- The 3×3 FOV is restricted in communication capacity
  - Reducing communication overheads and redundant information received

Computation Efficiency and Performance

- Analyzing computation time
  - Proportional to FOV size
  - Affected by communication scope & observation size
- All computation time is normalized by the minimum time
  - Eliminate hardware differences
  - i.e., 3×3 always have normalized time of 1
- Normalized success rate
  - Take computation time into consideration
  - $r' = \frac{r}{\bar{t}}, r$ = success rate, $\bar{t}$ = normalized time
  - 3×3 is the most efficient

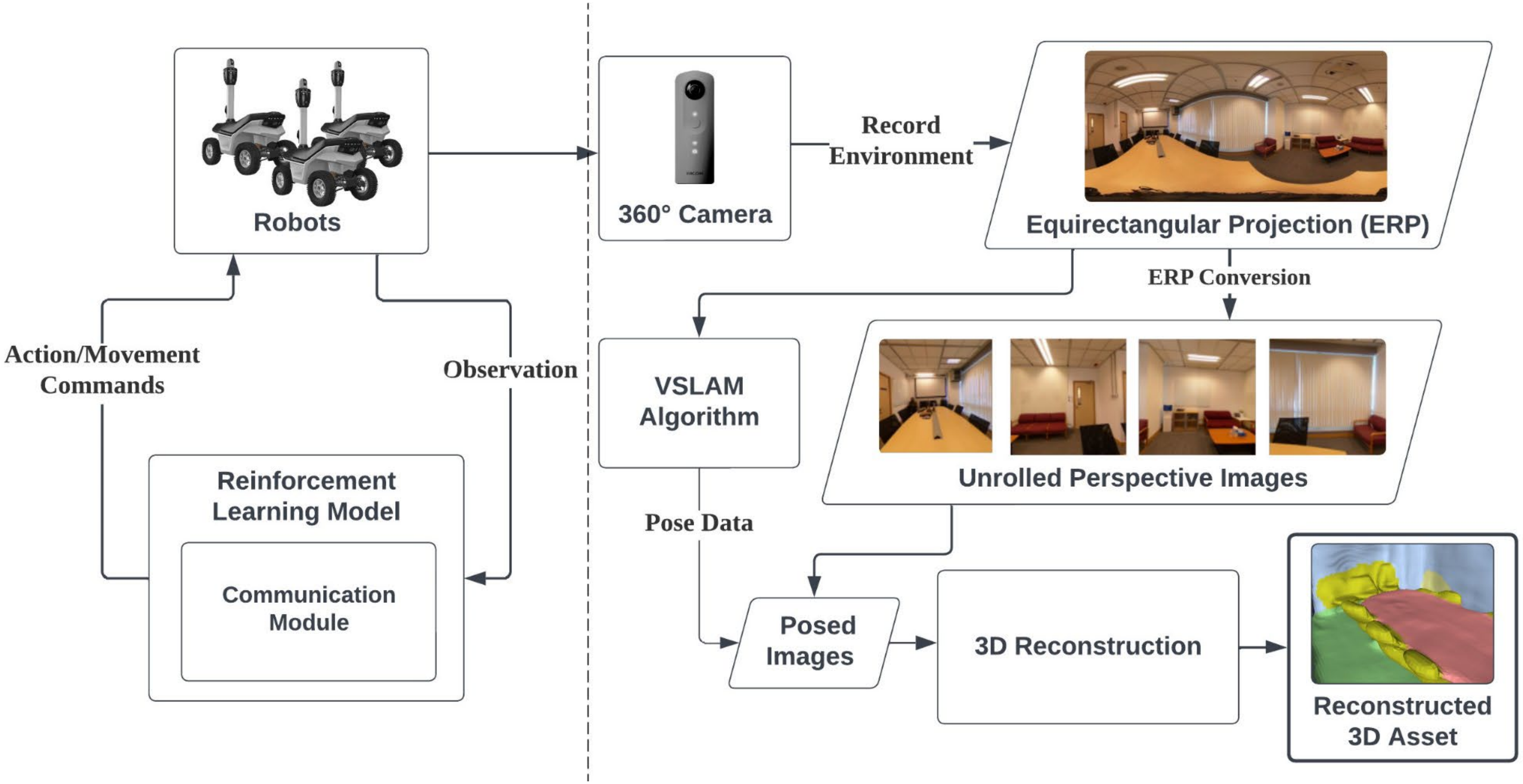




Outline

- Background and Motivations
- Leveraging 360° Cameras in 3D Reconstruction
- Optimizing Communication in Multi-Agent Path Finding
- Summary and Conclusion

Summary and Conclusion



1. A 3D reconstruction pipeline for 360° cameras
2. A communication-based MAPF framework and a FOV study

Future Works

- Explore the following approaches
  - Data fusion for 3D reconstruction
    - Combining measurements from LiDAR and Inertial sensors
  - 3D reconstruction algorithms
    - 3D Gaussian Splatting, 3D-aware diffusion model, NeRF, etc
  - Evaluation on simulation platform
    - Test our MAPF algorithm on simulation platform such as Gazebo and Issac Sim

Contribution List

- **H. C. Cheng**, Z. Hong, B. Hussain, Y. Wang, and C. P. Yue, “Development of Multi-Agent-based Indoor 3D Reconstruction, “ (Under Review), Robotics, 2024.
- **H. C. Cheng**, B. Hussain, Z. Hong, and C. P. Yue, “Leveraging 360° camera in 3d reconstruction: A vision-based approach,” in International Journal of Signal Processing Systems, vol. 12, pp. 1–6, 2024.
- **H. C. Cheng**, L. Shi, and C. P. Yue, "Optimizing Field-of-View for Multi-Agent Path Finding via Reinforcement Learning: A Performance and Communication Overhead Study," 2023 62nd IEEE Conference on Decision and Control (CDC), pp. 2141-2146, 2023.
- C. P. Yue, **H. C. Cheng**, and Z. Hong, “Enabling Technologies for Multi-Robot Human Collaboration,” *International Workshop on Signal Processing and Machine Learning*, 12943, pp. 286-293, 2023.
- B. Hussain, Y. Wang, R. Chen, **H. C. Cheng**, and C. P. Yue, “LiDR: Visible Light Communication-Assisted Dead Reckoning for Accurate Indoor Localization,” IEEE Internet of Things Journal, 9(17), pp. 15742-15755, 2022.
- B. Xu, B. Hussain, Y. Wang, **H. C. Cheng**, and C. P. Yue, “Smart Home Control System Using VLC and Bluetooth Enabled AC Light Bulb for 3D Indoor Localization with Centimeter-Level Precision,” Sensors, 22(21), pp. 8181, 2022.
- B. Xu, B. Hussain, X. Liu, T. Min, **H. C. Cheng,** and C. P. Yue, "Smart Home Control System Using VLC-Enabled High-Power LED Lightbulb," 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE), pp. 744-745, 2021.

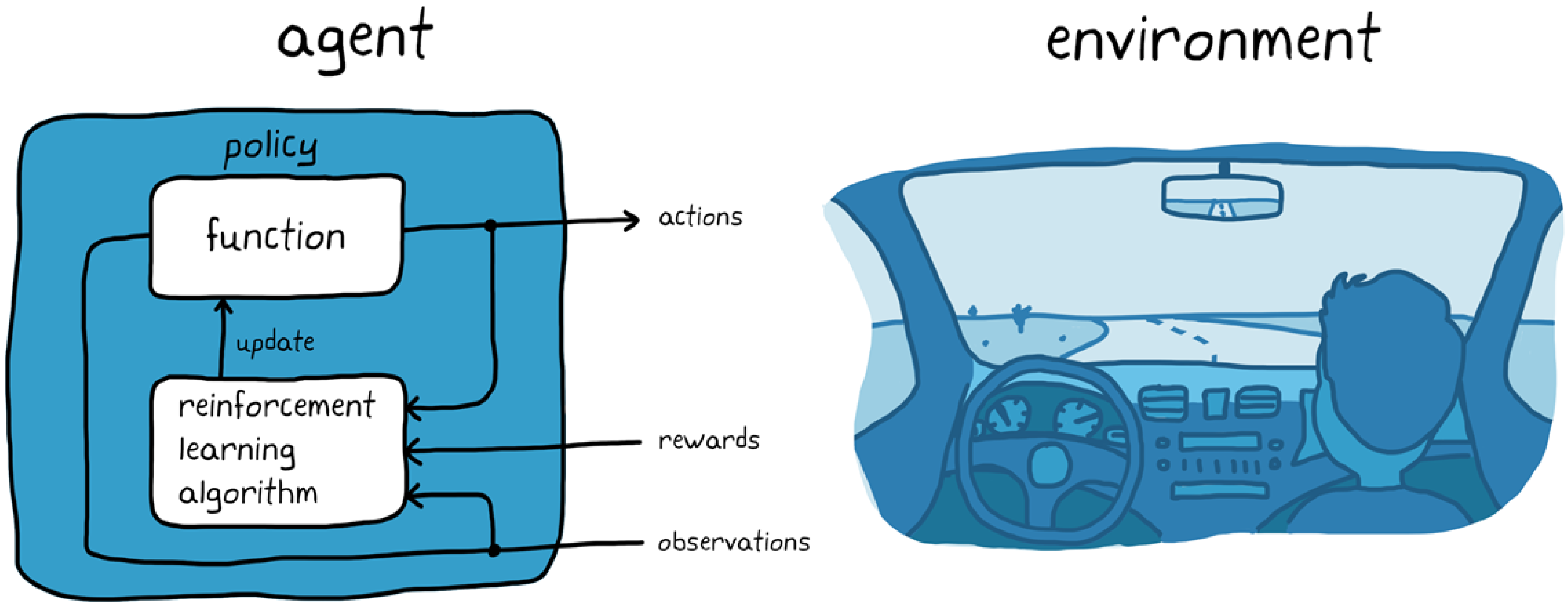
Acknowledgements

- Supervisor: Prof. Chik Patrick Yue
- Committee Membe
  - Prof. Ling Shi
  - Prof. Shenghui So
- Optical Wireless La
- Fa

Thank You!

Optical Wireless Lab (OWL) & Integrated Circuit Design Center (ICDC)
Department of Electronic and Computer Engineering (ECE)
The Hong Kong University of Science and Technology (HKUST)

Reinforcement Learning

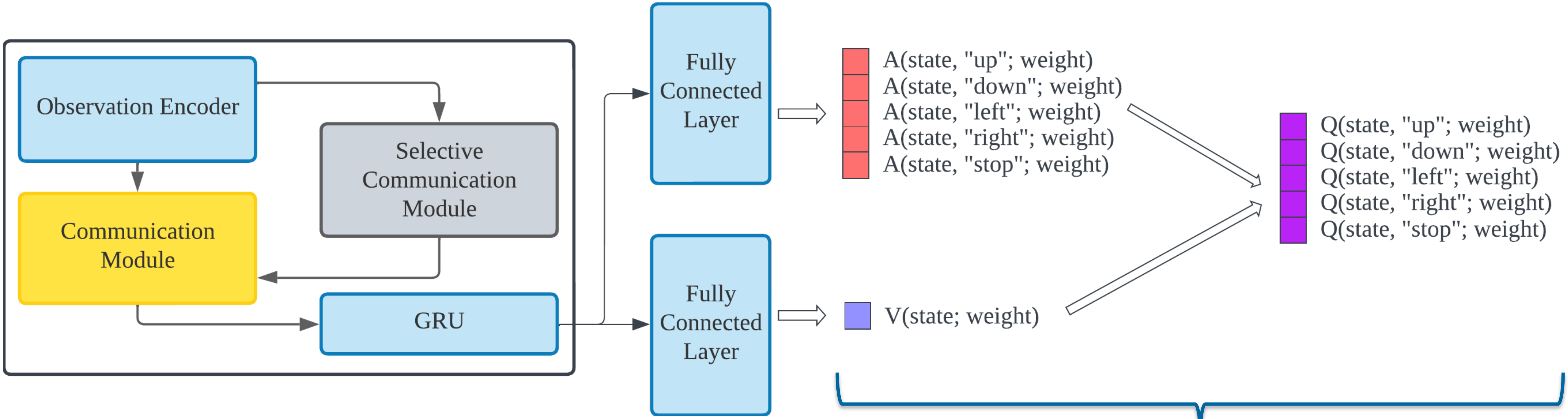


1. Initialize environment and policy
2. Define reward structure
3. Interact with environment
4. Update the policy based on the reward

Reinforcement Learning

- Sampling experiences
  - For each agent: interact with environment and store the transition in replay buffer
  - Transition = [state, action, rewards, next state]
- Training
  1. Sample batches from replay buffer
  2. Target value = rewards + discount factor * network(next state)
     - <u>Assuming higher accuracy</u>
  3. Loss = MSE(target value, network(state))
     - Backpropagate the loss and update network

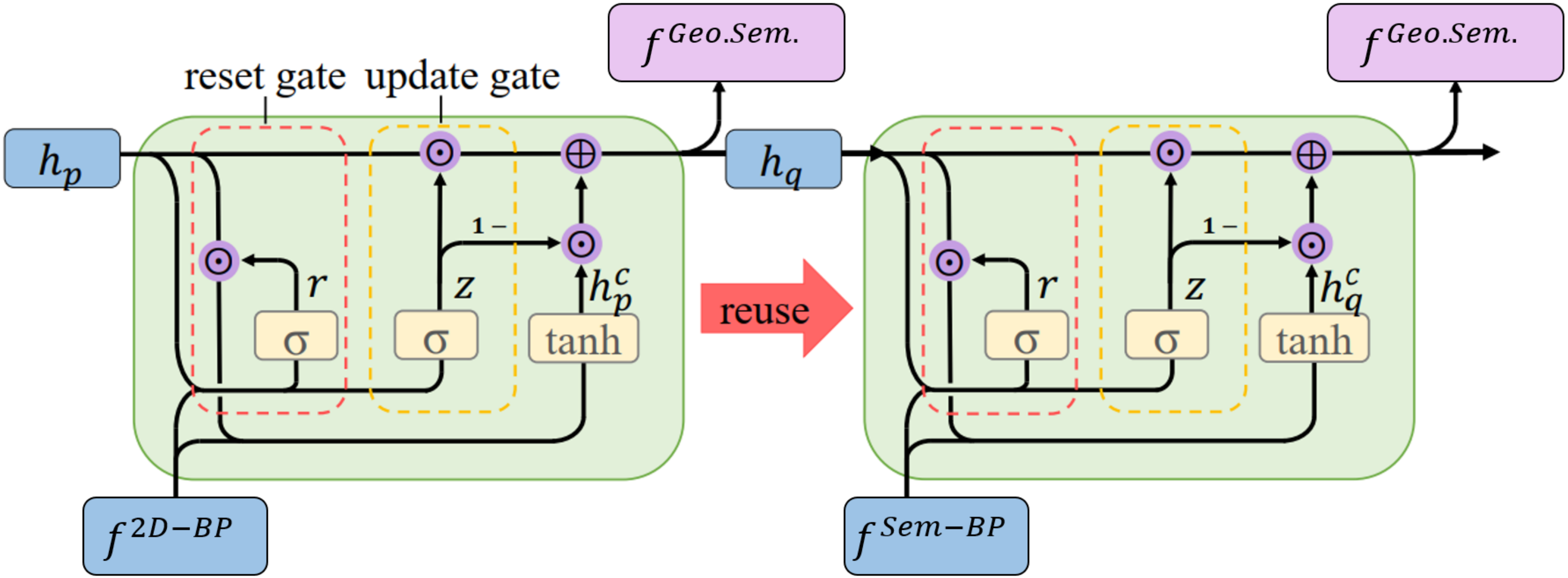
Dueling Deep Q-Network



$$Q^*(s,a;w^A,w^V) = V^*(s;w^V) + A^*(s,a;w^A) - mean_a A^*(s,a;w^A)$$

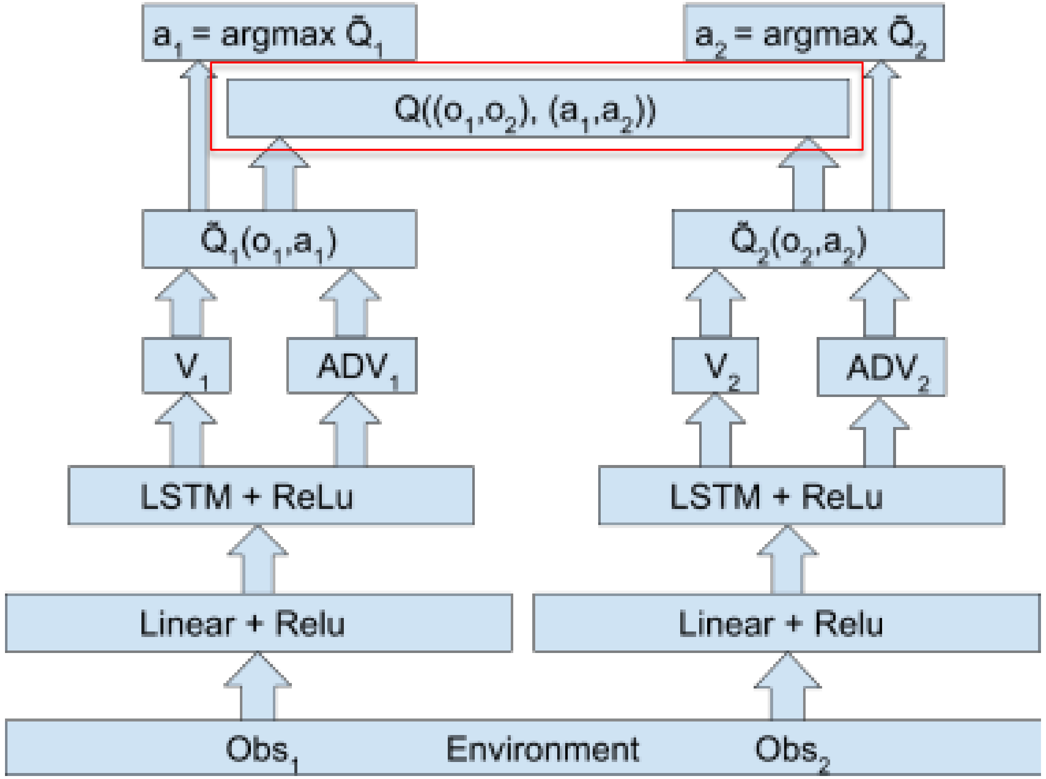
- Output of the final GRU = input of 2 fully connected layers

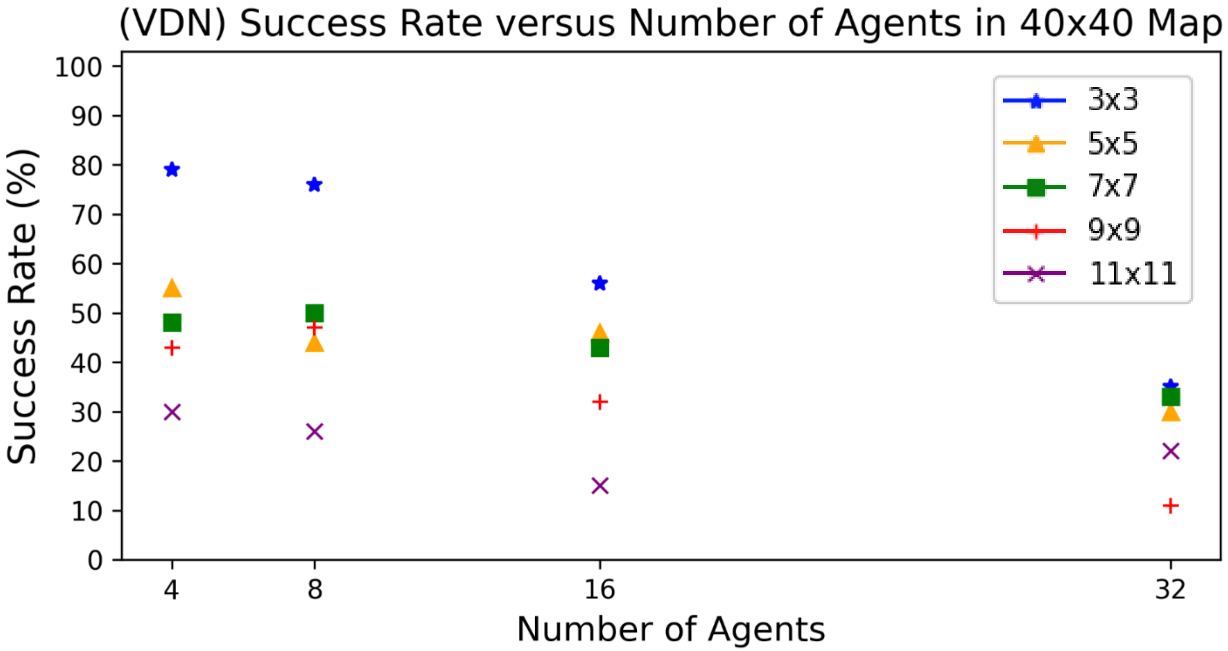
GRU Design



⊙ Element-wise product ⊕ Element-wise add Stream replicate Stream aggregate

FOV Study on Value-Decomposition Networks (VDN)

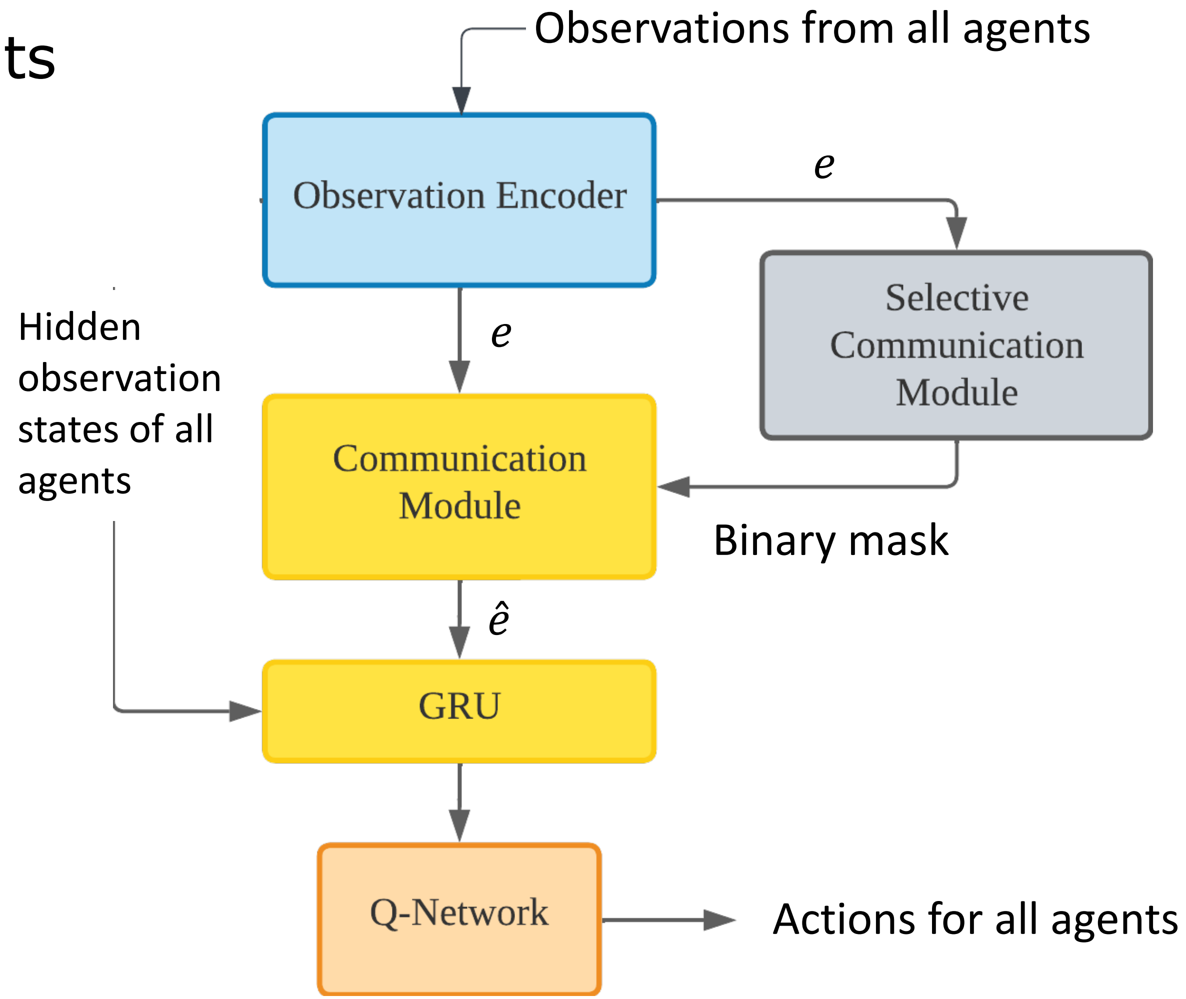




- Similarly, smaller FOV shows better performance in VDN

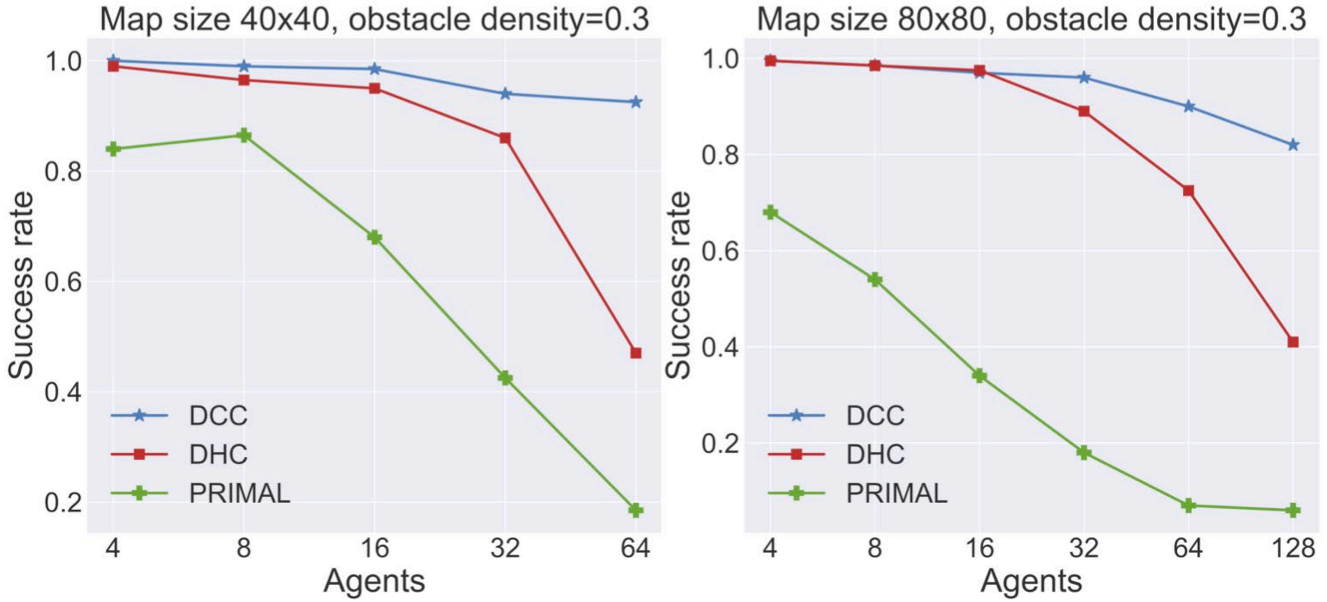
Communication Module (All Agents)

- Calculating attention score across all agents
  - $\mu^h = softmax\left[\frac{W_Q^h e \cdot (W_K^h e)^T}{\sqrt{d_K}}\right]$
  - e : observation embeddings of **all agents**
    - size = $batch\ size \times no. of\ agents \times no. of\ agents \times observation\ embedding\ size$
- Embeddings from communication module
  - $\hat{e} = f_o\left[concat\left[\sum \mu^h W_V^h e, \forall h \in H\right]\right]$
  - f_o: fully connected layer



Z. Ma et al., "Distributed Heuristic Multi-Agent Path Finding with Communication," IEEE International Conference on Robotics and Automation (ICRA), pp. 8699-8705, 2021.

Why Selective Communication



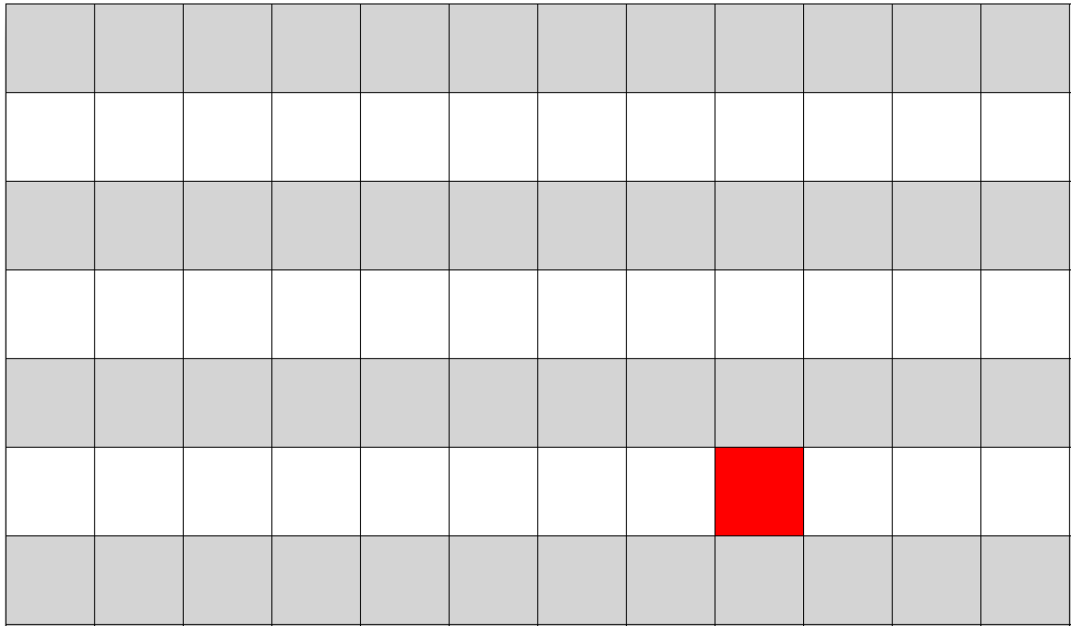
Success rate comparison between DCC, DHC and PRIMAL

- Major difference between DCC & DHC
  - DCC communicate as needed
  - DHC's performance drops drastically when congestion happens
- DHC: broadcast communication

Z. Ma et al., "Learning Selective Communication for Multi-Agent Path Finding," IEEE Robotics and Automation Letters (RA-L), vol. 7, no. 2, pp. 1455-1462, 2022.

From Discrete MAPF to Real World Deployment

Warehouse



Discretized Map

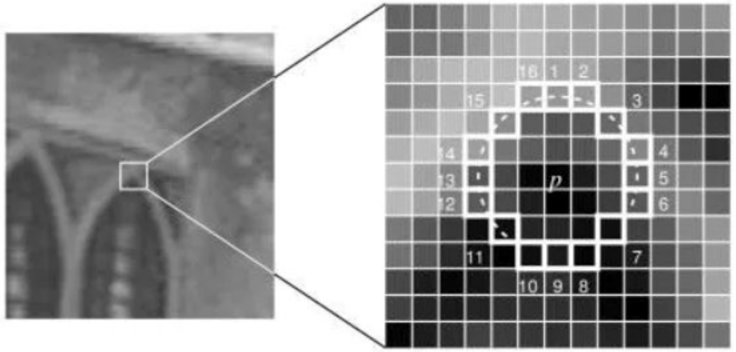
- Digitalize the real world
  - Use the MAPF RL algorithm as a global planner
  - Use algorithms such as TEB planner as a local planner and navigate through complex environment

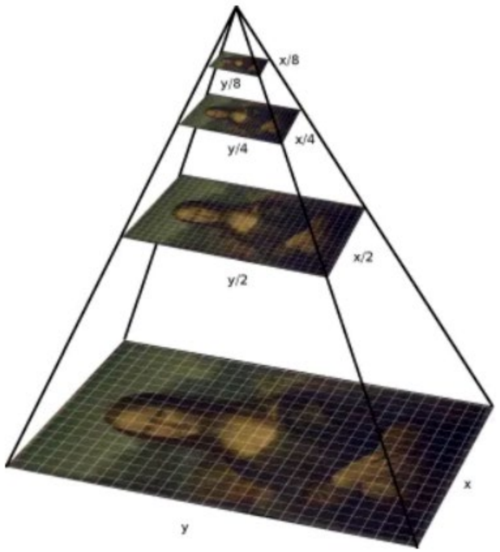
References

- R. Stern et al., ”Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks,” in Symposium on Combinatorial Search, pp. 151–159, 2019.
- Z. Ma et al., “Learning Selective Communication for Multi-Agent Path Finding,” IEEE Robotics and Automation Letters (RA-L), vol. 7, no. 2, pp. 1455-1462, 2022.
- Z. Ma et al., “Distributed Heuristic Multi-Agent Path Finding with Communication,” IEEE International Conference on Robotics and Automation (ICRA), pp. 8699-8705, 2021.
- D. Horgan et al., “Distributed Prioritized Experience Replay,” International Conference on Learning Representations, 2018.

ORB Feature Extractor

- Oriented FAST and Rotated BRIEF (**ORB**) feature extractor
- Features from Accelerated and Segments Test (**FAST**) key point detector

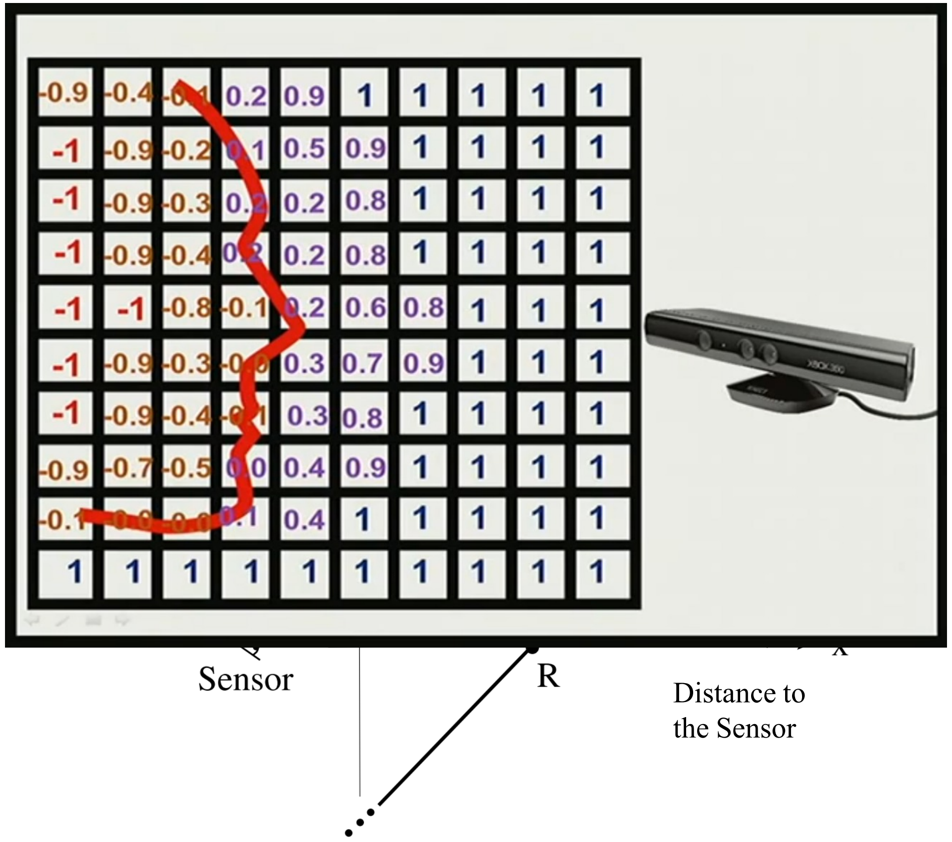




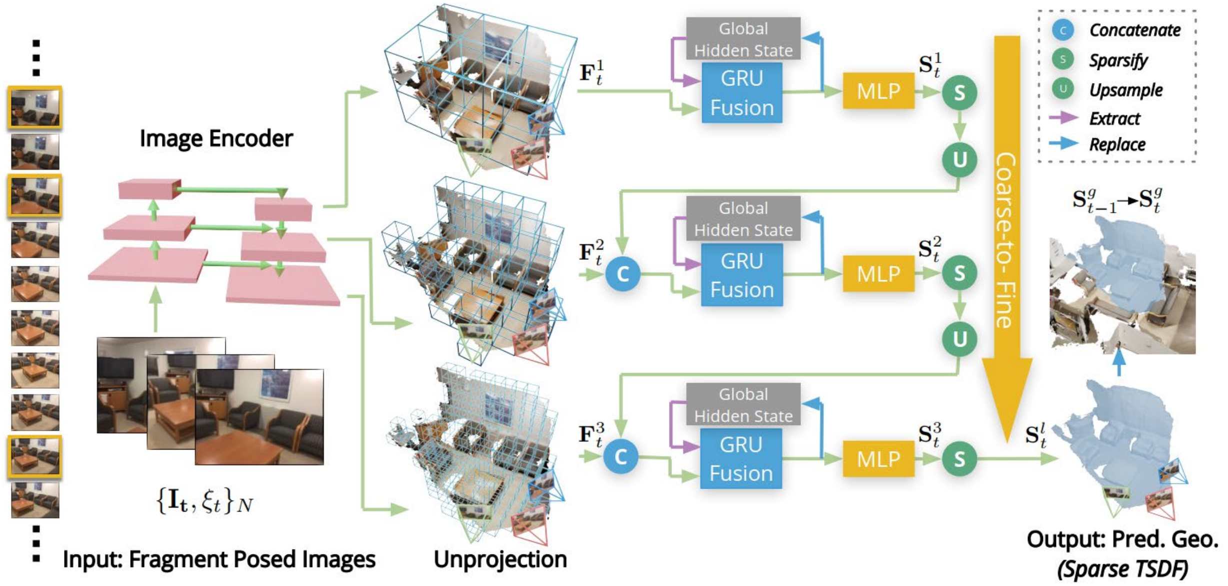
- Binary robust independent elementary feature (**BRIEF**) descriptor

Voxelization: Truncated Signed Distance Function/Field (TSDF)

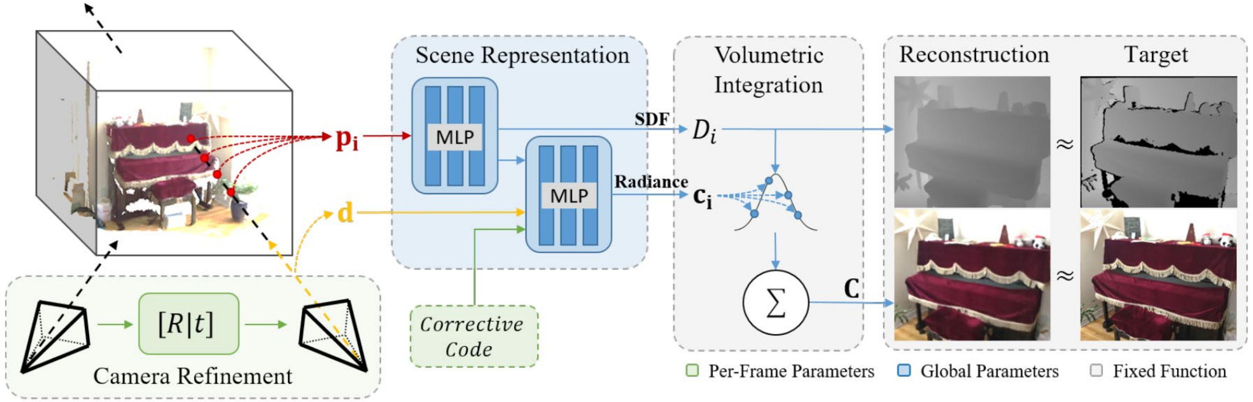
- Consider a bird-eye view of a **surface**
  - SDF: Each voxel center's signed distance to its nearest object surface
    - $sdf_i(\mathrm{x}) = depth_i\big(pic(\mathrm{x})\big) - cam_z(\mathrm{x})$
    - Surface's SDF value is 0
  - TSDF: Truncation into [-1, 1]
    - better occupancy representation and storage



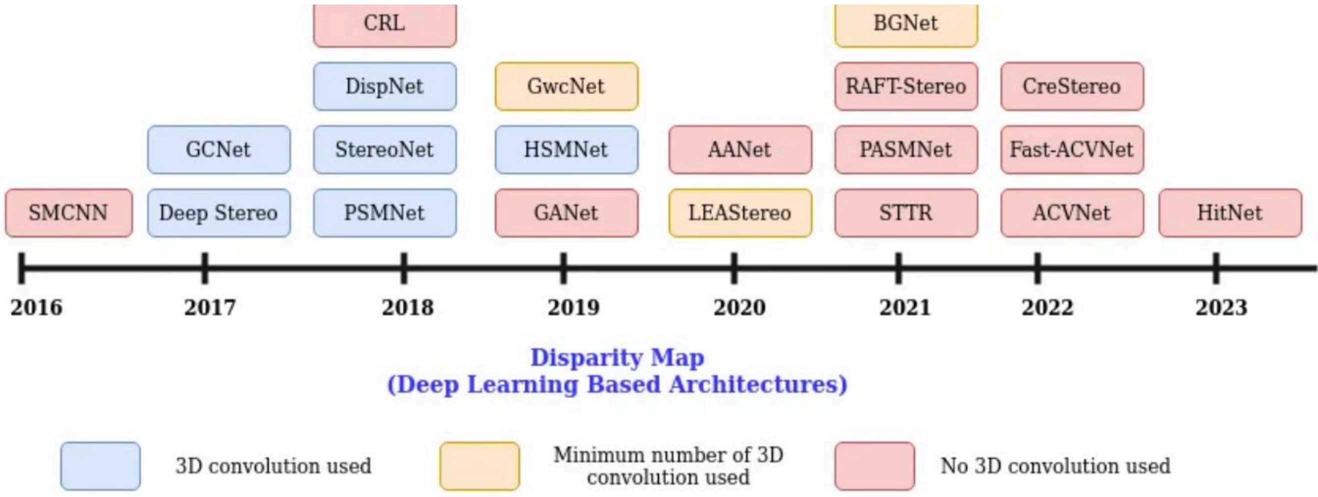
Other 3D Reconstruction Techniques



***NeuralRecon*, Sun et al., CVPR'21**

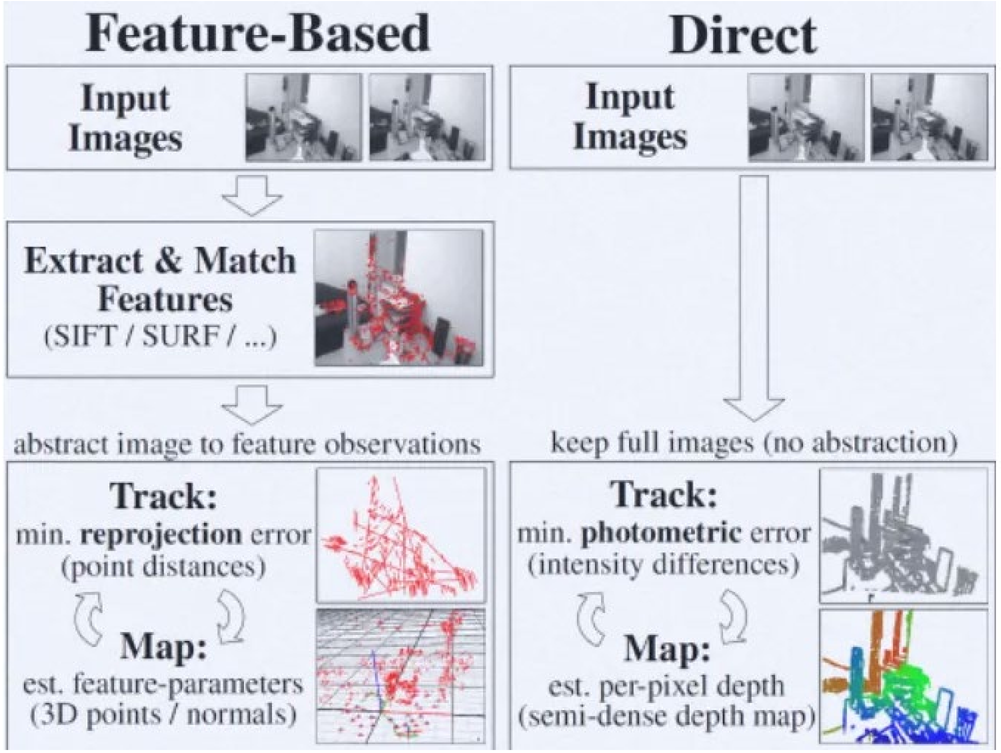


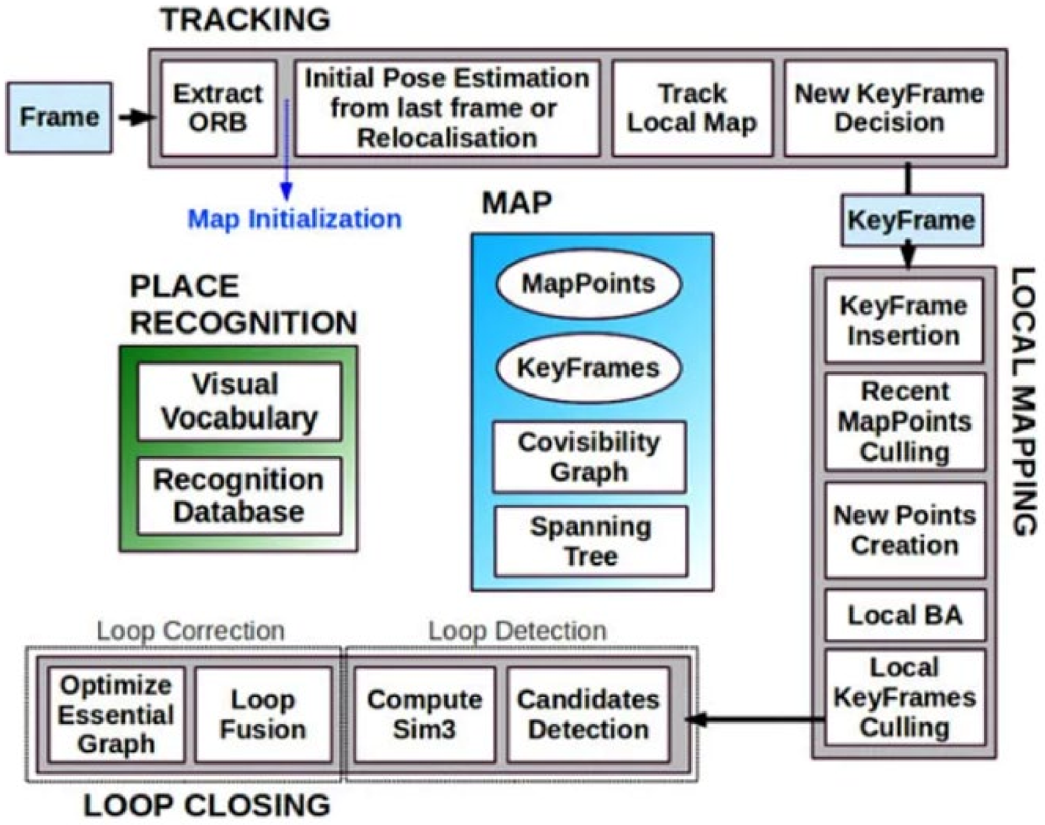
***Neural RGB-D*, Azinović et al., CVPR'22**



SMCNN : Stereo Matching CNN, **GCNet:** Geometry and Context Net, **PSMNet**:Pyramid Stereo Matching Network,
DispNet: DisparityNet, **CRL:** Cascade Residual Learning, **GANet:** Guided Aggregation Net,
HSMNet: Hierarchical Stereo Matching, **GwcNet:** Group-wise Correlation StereoNet, **AANet:** Adaptive Aggregation Network
LEAStereo: Learning Effective Architecture Stereo, **STTR:** Stereo Transformer, **BGNet:**Bilateral Grid Network
PASMNet: Parallax-attention stereo matching network, **ACVNet:** Attention Concatenation Volume
CreStereo: Cascade Recurrent Network, **HitNet:** Hierarchical Iterative Tile Refinement Network

Other VSLAM Techniques

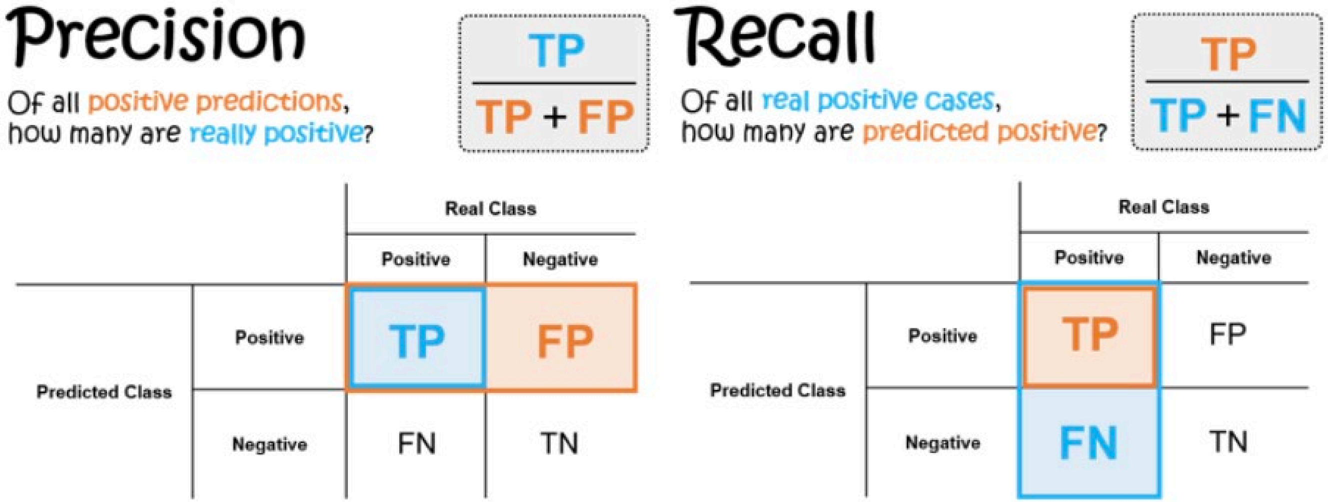




ORB-SLAM2

- Indirect vs direct SLAM
- Alternative: ORB-SLAM2

Performance metric: F-score



$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\mathrm{TP}}{\mathrm{TP} + \frac{1}{2}(\mathrm{FP} + \mathrm{FN})}$$

TP = number of true positives

FP = number of false positives

FN = number of false negatives

- Widely used metric in classification tasks
- Measuring a model’s predictive performance

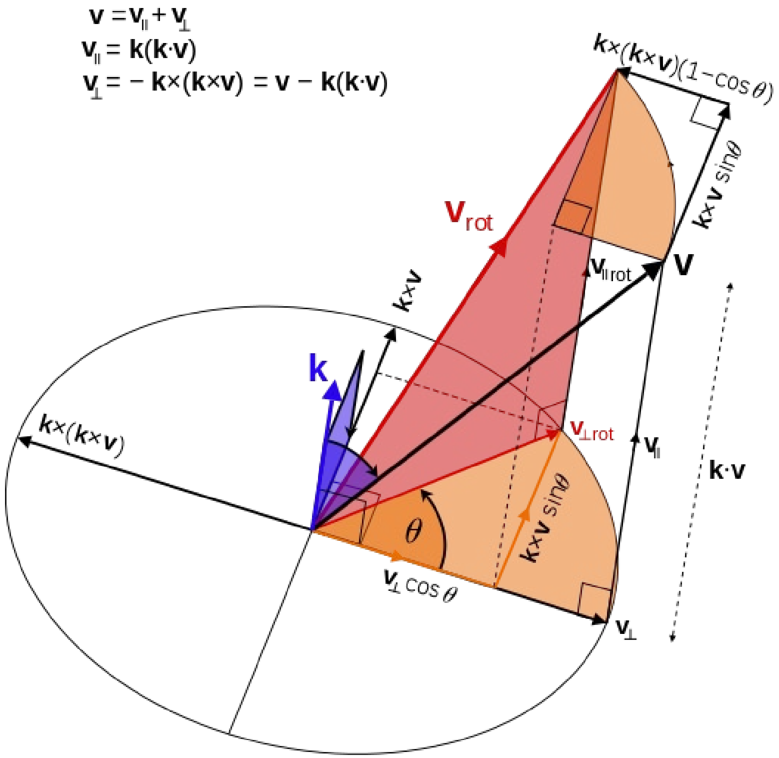
Rodrigues' Rotation

$$\begin{bmatrix}(\mathbf{k}\times\mathbf{v})_x\\(\mathbf{k}\times\mathbf{v})_y\\(\mathbf{k}\times\mathbf{v})_z\end{bmatrix}=\begin{bmatrix}k_yv_z-k_zv_y\\k_zv_x-k_xv_z\\k_xv_y-k_yv_x\end{bmatrix}=\begin{bmatrix}0&-k_z&k_y\\k_z&0&-k_x\\-k_y&k_x&0\end{bmatrix}\begin{bmatrix}v_x\\v_y\\v_z\end{bmatrix}.\qquad \mathbf{K}=\begin{bmatrix}0&-k_z&k_y\\k_z&0&-k_x\\-k_y&k_x&0\end{bmatrix}.$$

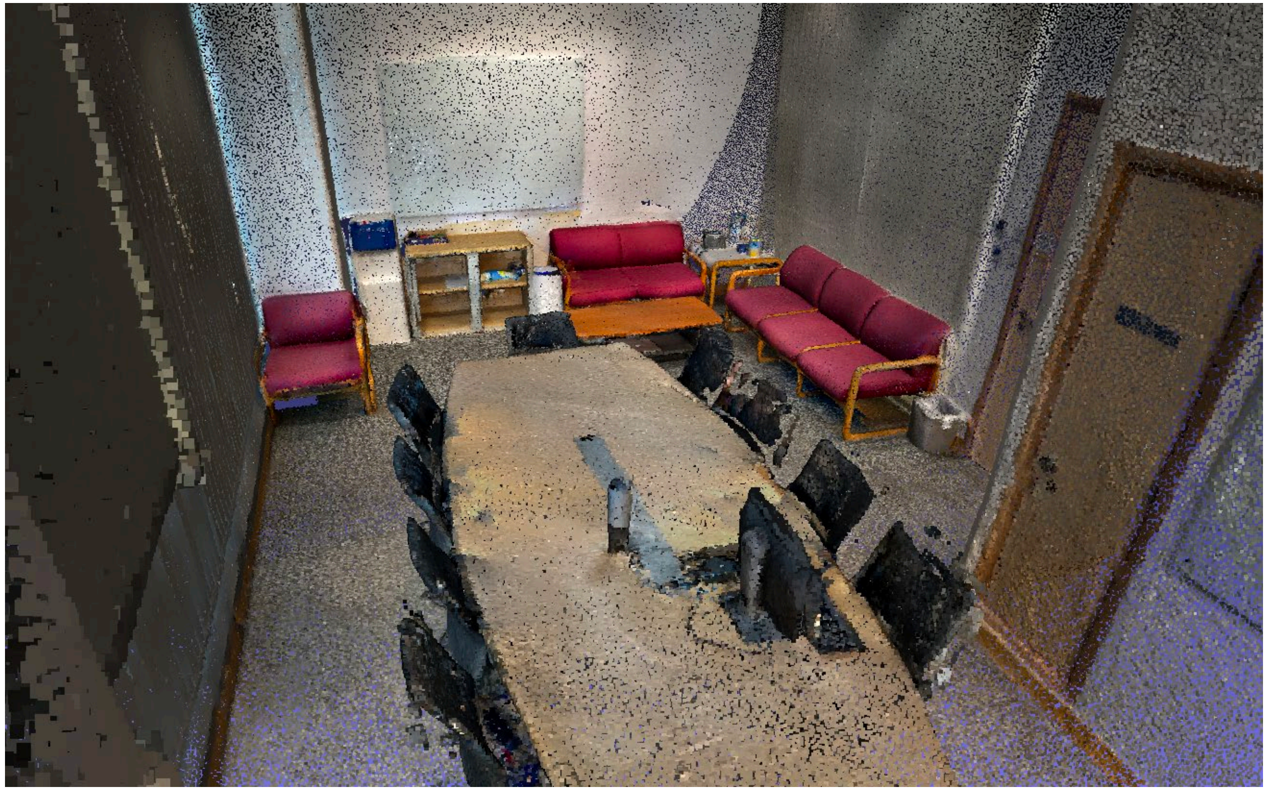
$$\mathbf{k}\times\mathbf{v}=\mathbf{K}\mathbf{v},\qquad \mathbf{k}\times(\mathbf{k}\times\mathbf{v})=\mathbf{K}(\mathbf{K}\mathbf{v})=\mathbf{K}^2\mathbf{v}\,.$$

$$\boxed{\mathbf{R}=\mathbf{I}+(\sin\theta)\mathbf{K}+(1-\cos\theta)\mathbf{K}^2}$$

$$\mathbf{v}_{\mathrm{rot}}=\mathbf{v}+(1-\cos\theta)\mathbf{k}\times(\mathbf{k}\times\mathbf{v})+\sin(\theta)\mathbf{k}\times\mathbf{v}.$$

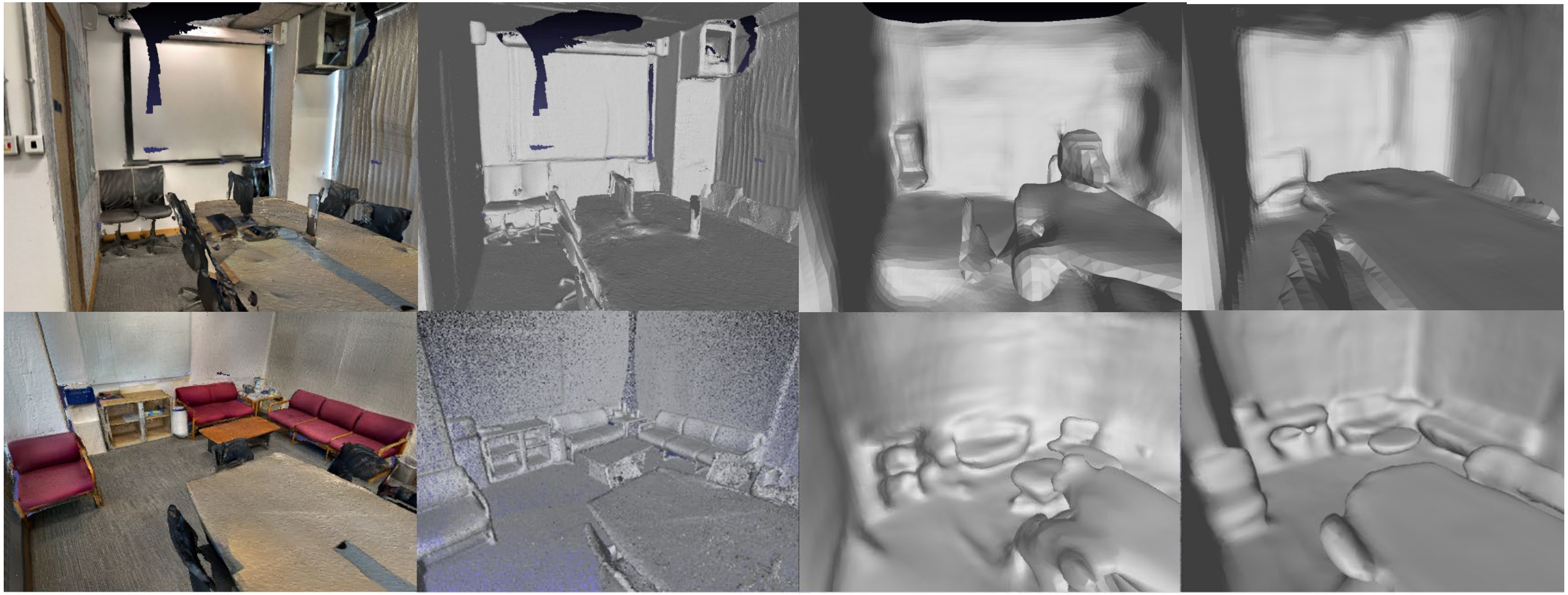


Generating Ground Truth Models



Ground truth data obtained from a LiDAR sensor

Qualitative 3D Reconstruction Results



Ground Truth (with texture) | Ground Truth (without texture) | Perspective Camera | 360 Camera

- 3D model comparison

References

- X. Song et al, “Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving," in Proceedings of the IEEE/CVF Conference on CVPR (pp. 5452-5462).
- S. Sumikura et al., “OpenVSLAM: A versatile visual SLAM framework,” in *Proceedings of the 27th ACM International Conference on Multimedia* (pp. 2292-2295), 2019.
- Z. Murez et al., “Atlas: End-to-end 3d scene reconstruction from posed images,” in Proceedings of the ECCV 2020 (Part VII 16 (pp. 414-431), 2020.
- Behind the Robot: HITT's Construction Site Monitoring Husky UGV - Clearpath Robotics: https://clearpathrobotics.com/blog/2021/10/behind-the-robot-hitts-construction-site-monitoring-husky-ugv/
- Digital Giza: Tomb of Queen Meresankh III (G 7530-7540): http://giza.fas.harvard.edu/giza3d/?mode=matterport&m=d42fuVA21To

More on Future Works

- Applications in multi-agent 3D reconstruction
  - Multiple UAVs doing 3D recon: https://www.intechopen.com/chapters/68371