Super Ray based Updates for Occupancy Maps

Youngsun Kwon, Donghyuk Kim, and Sung-eui Yoon

Source code is available at
http://sglab.kaist.ac.kr/projects/SuperRay

ICRA 2016 IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION

Content

- Background
- Related Work
- Problem Definition
- Our Approach
- Result
- Conclusion

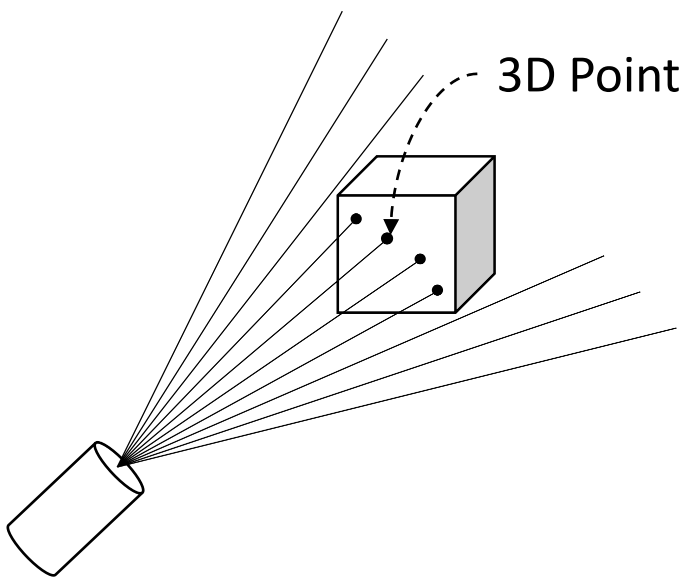
Background

- **Navigation using depth sensor**

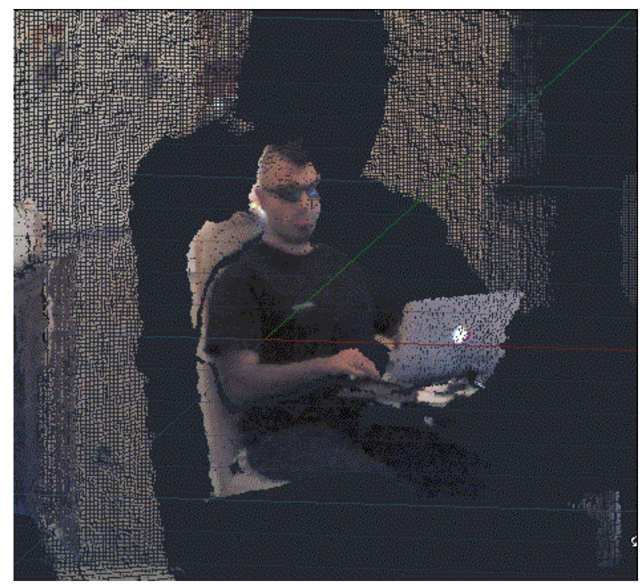
Daniel et al, Humanoids, 2012

Background

- **Depth sensor generates point clouds**
  - Consist of a large amount of points with noise
  - Provide useful geometric information of environment



Schematic Illustration

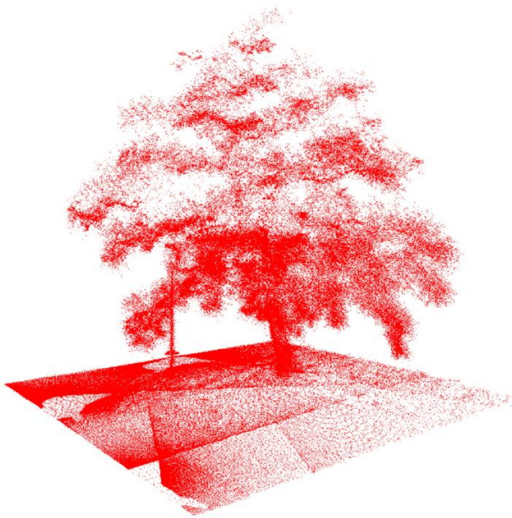


Real Example of Point Clouds

http://through-the-interface.typepad.com

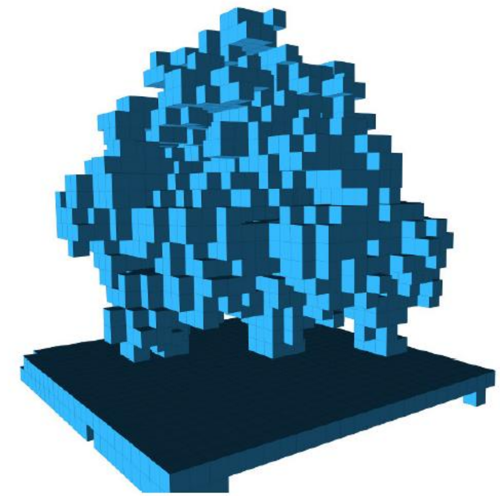
Background

- **General flow for using point clouds**



Point clouds

Update

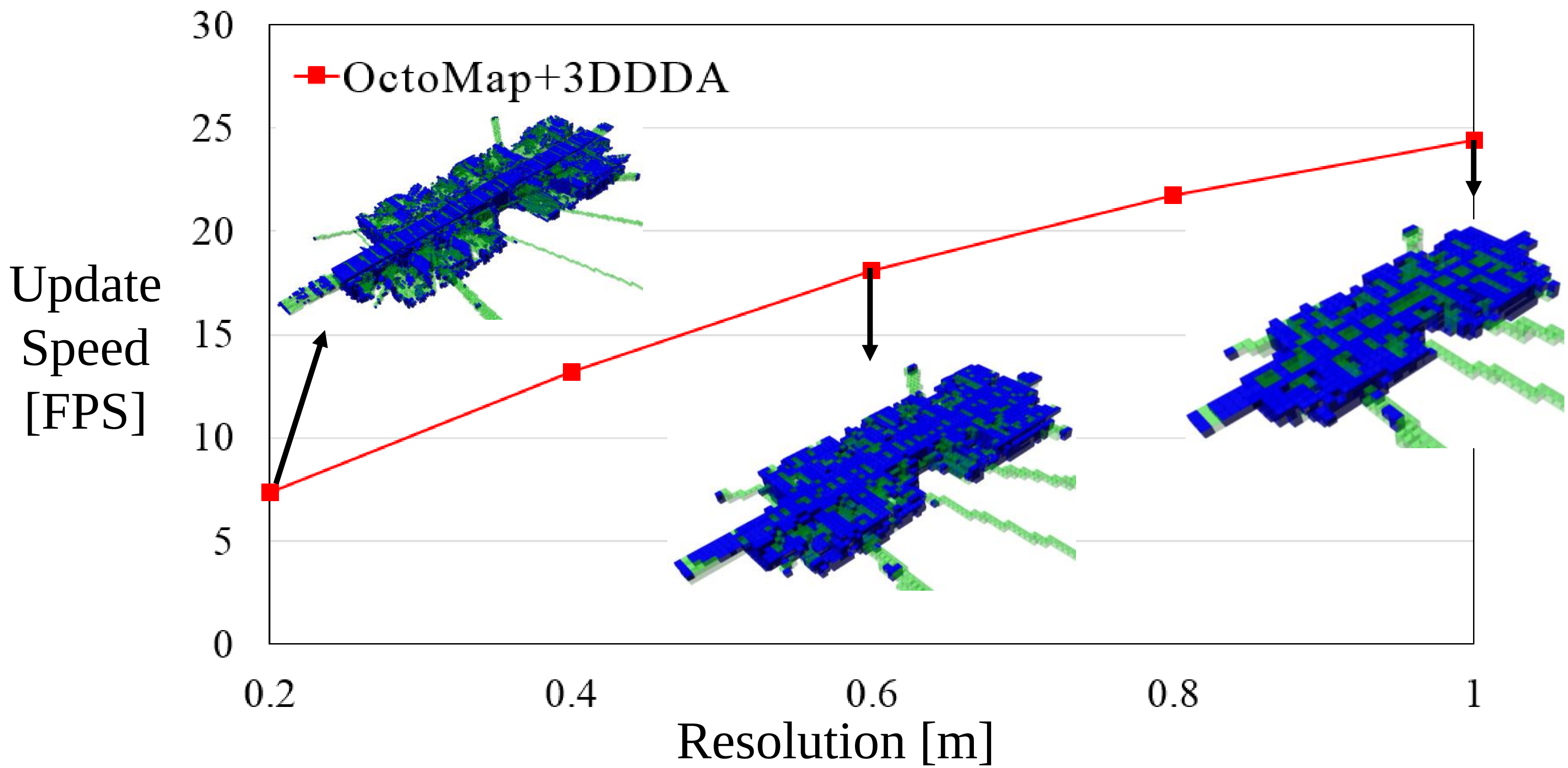


Map representation
(grids or octrees)

Applications:
e.g. Path Planning and Collision Detection

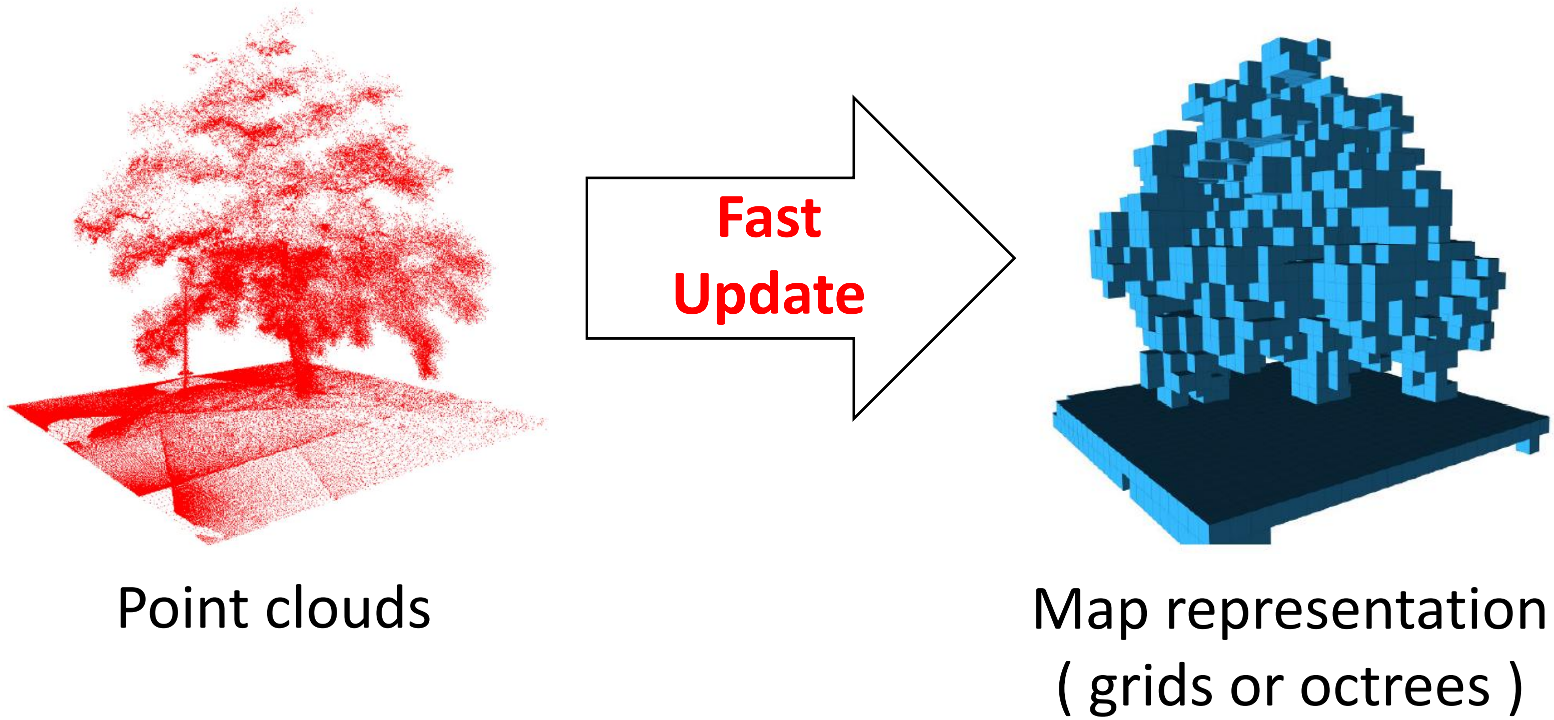
Research Goal

- **Update speed vs. Representation accuracy**
  - Issues for both **real-time** and **high quality** are important



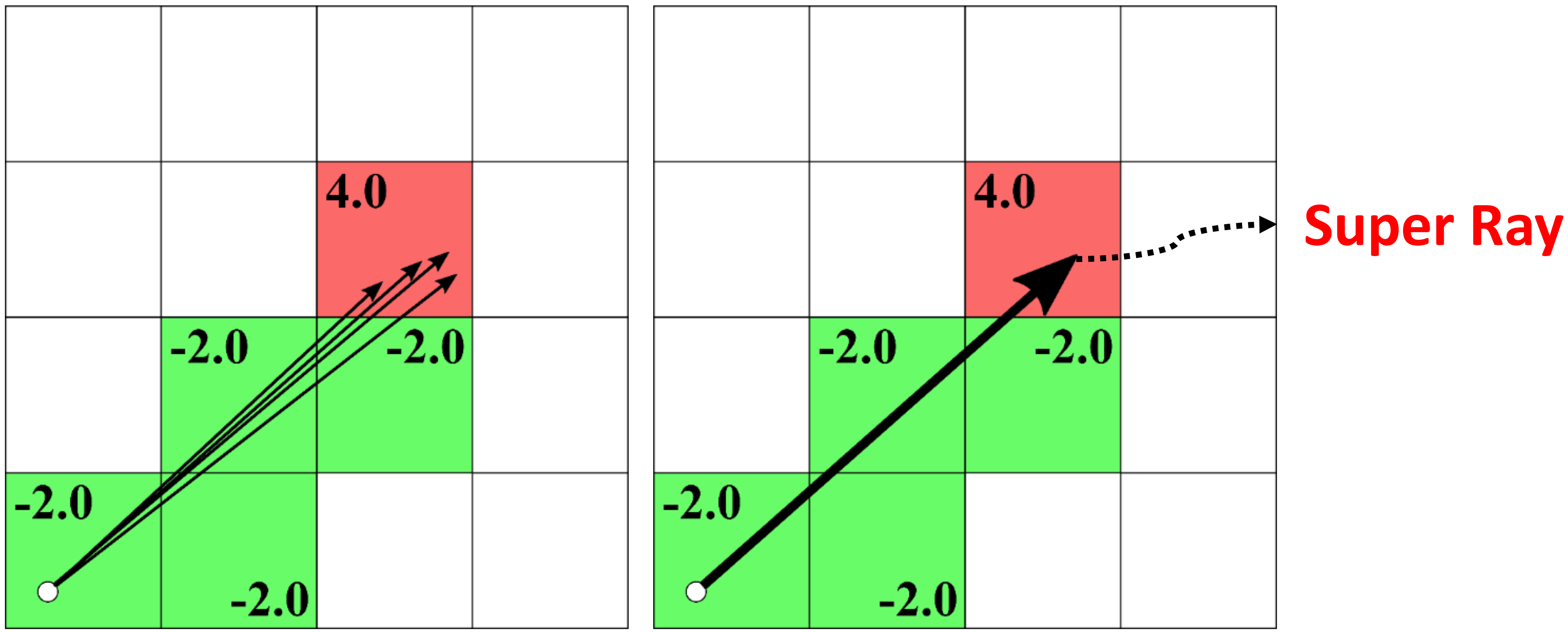
Research Goal

- **Accelerate update speed of map without degrading the representation accuracy**



Introduction

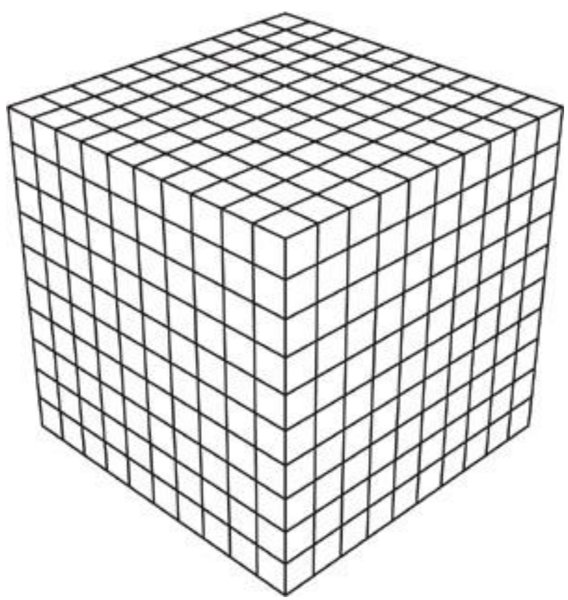
- **Super Ray based Updates**
  - Enable **2.5 times** on average **performance improvement** over the state-of-the-art update method **without degrading** the representation accuracy



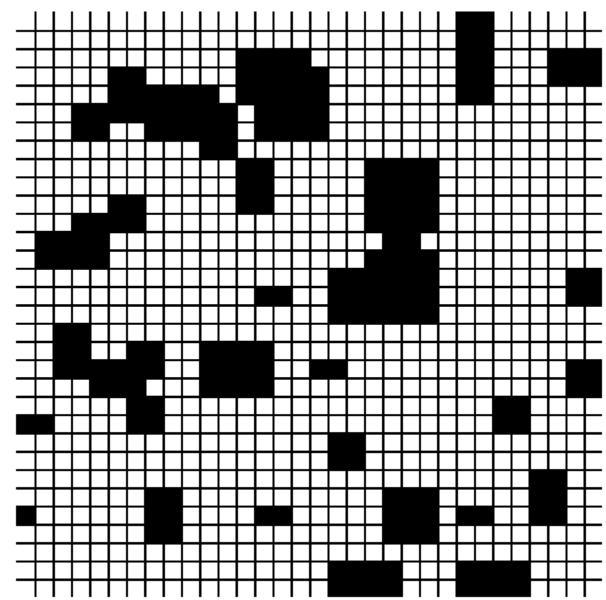
State-of-the-art method **Ours**

Related Work

- **Map Representation**
  - **Grid Map [Roth-Tabak et al., *Computer, 1989*]**
    - Models a space using grid cells
    - Requires a large size of memory



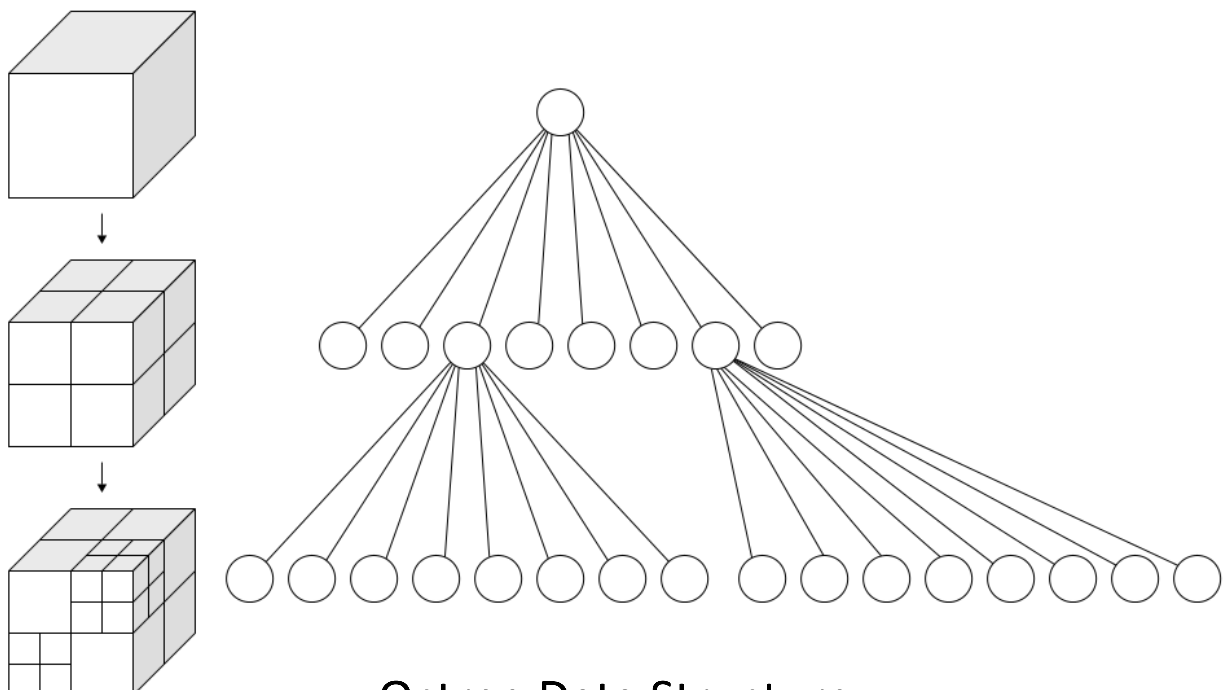
3D Grid Map



2D Grid Map

Related Work

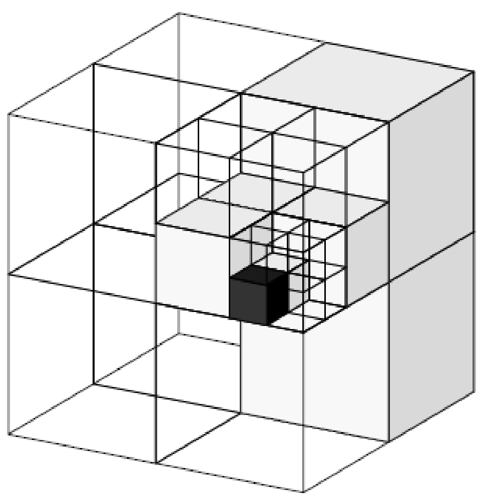
- **Map Representation**
  - **Octree Map [Payeur et al., *ICRA, 1999*]**
    - Divides a 3-D space into 8 sub-spaces recursively



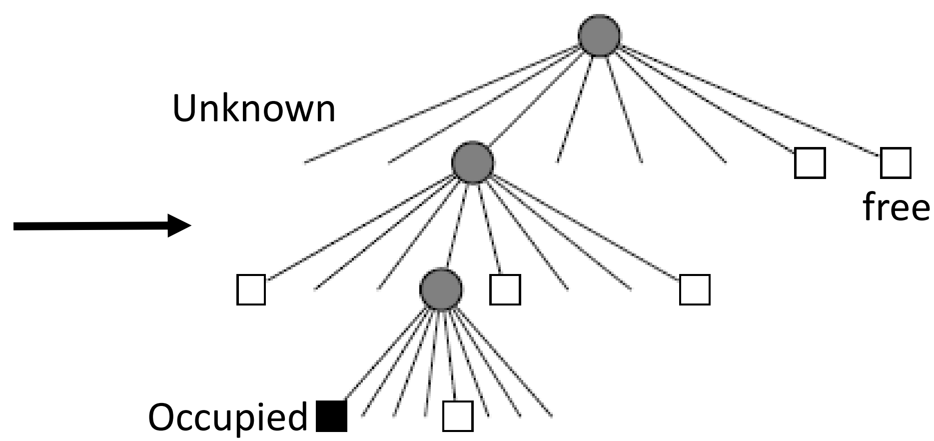
Octree Data Structure

Related Work

- **Occupancy Map Representation**
  - **OctoMap [Wurm et al., *ICRA, 2010*]**
    - Uses the Octree Map
    - Employs an **occupancy probability** to represent an occupied state (free, occupied, and unknown) of a cell



3D OctoMap



Octree representation with states

Related Work

- **Occupancy Map Representation**
  - **OctoMap [Wurm et al., *ICRA, 2010*]**
    - Occupancy probability of cell n given measurement $z_{1:t}$

$$\boldsymbol{L(\,n\,|\,z_{1:t}\,) = L(\,n\,|\,z_{1:t-1}\,) + L(\,n\,|\,z_t\,)}$$

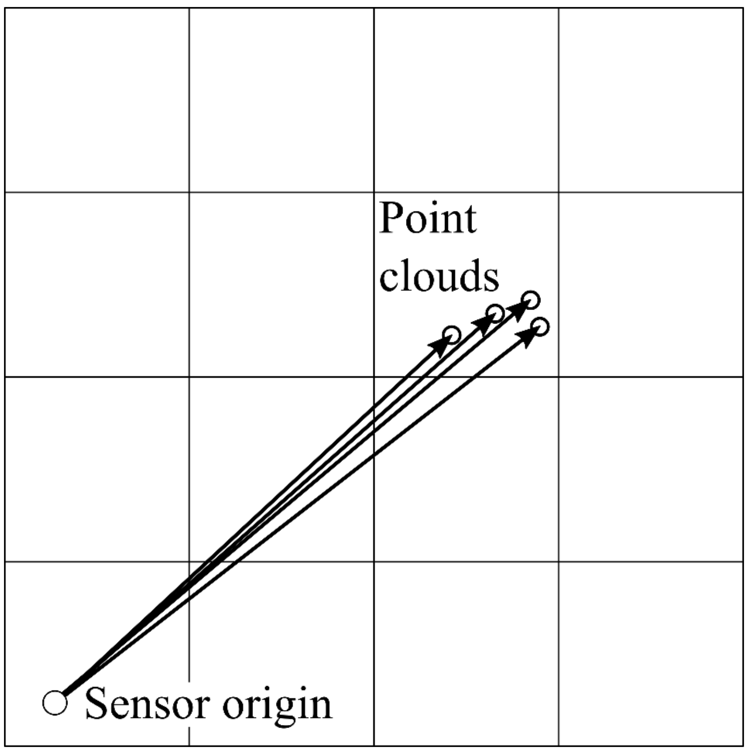
Occupancy probability of the cell n at time step $t-1$

New sensor measurement z_t to be updated at time step t

$$L(\,n\,|\,z_t\,) = \begin{cases} l_{occ} & occupied\ state \\ l_{free} & free\ state \end{cases}$$

Problem Definition

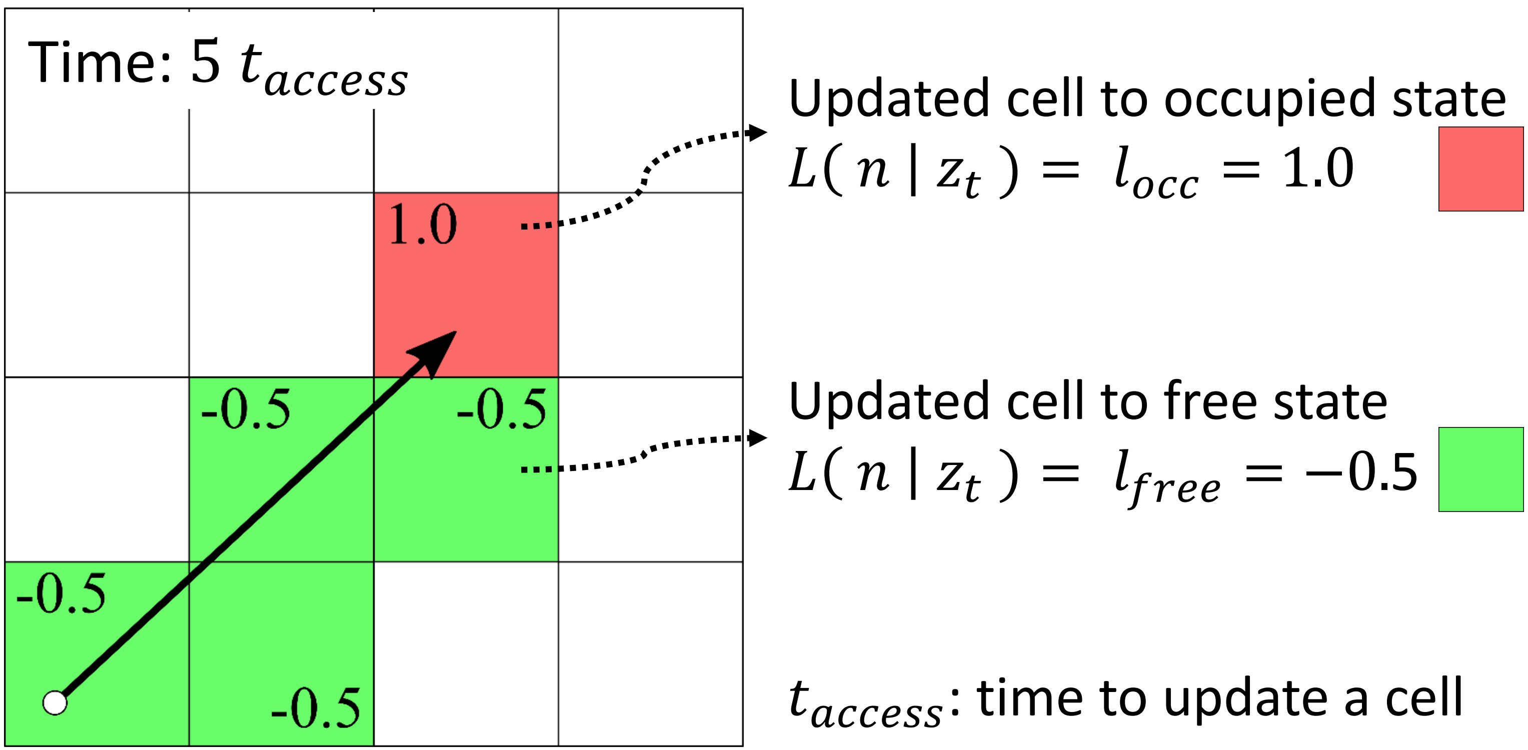
- **It takes long time to update map**
  - 3DDDA Algorithm [J. Amanatides et al., *Eurographics*, *1987*]



- Associate **a ray** with a point starting from the sensor origin
- To compute which cells should be update, **traverse** cells along the ray

Problem Definition

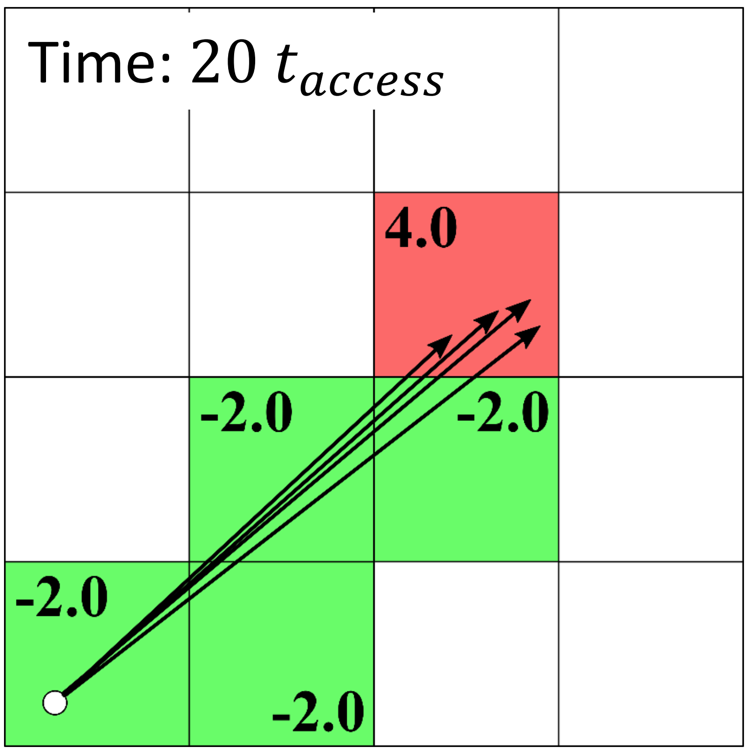
- **It takes long time to update point clouds**
  - 3DDDA Algorithm [J. Amanatides et al., *Eurographics, 1987*]



t_{access}: time to update a cell

Problem Definition

- **It takes long time to update point clouds**
  - 3DDDA Algorithm [J. Amanatides et al., *Eurographics*, *1987*]



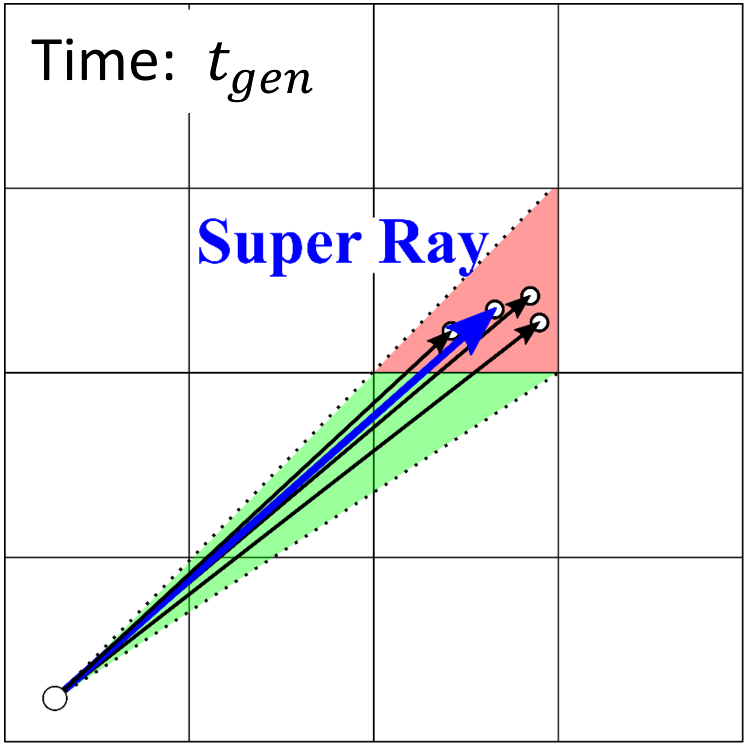
- Visit **the same cells multiple times** for multiple rays

t_{access}: time to update a cell

Key Idea of Our Approach

- **Propose a novel concept: Super Ray**
  - A representative ray for set of points that traverse the same cells

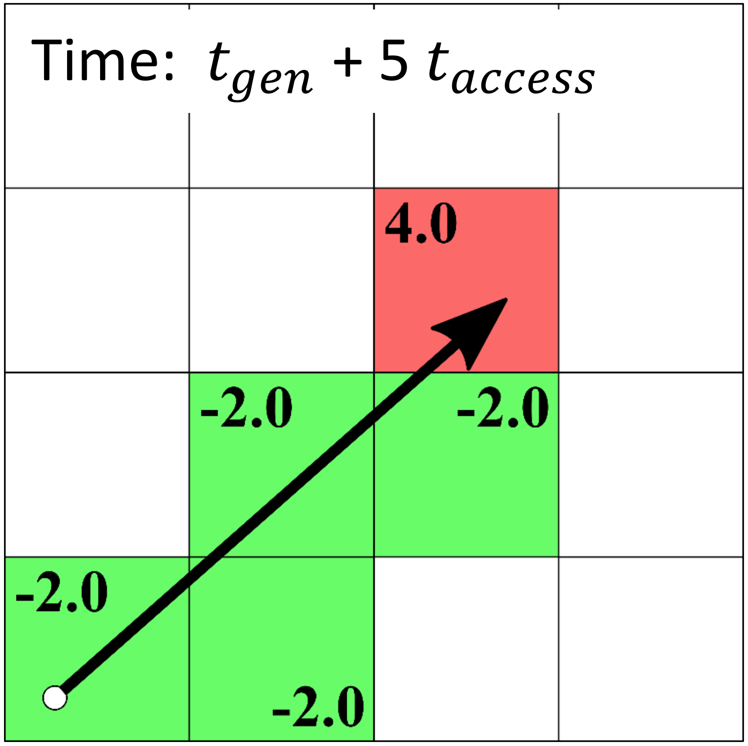
- Collect points associated with rays that **traverse the same cells**



t_{gen}: overhead to generate super rays

Key Idea of Our Approach

- **Propose a novel concept: Super Ray**
  - A representative ray for set of points that traverse the same set of cells
    - The super ray traverses cells **only a single time**



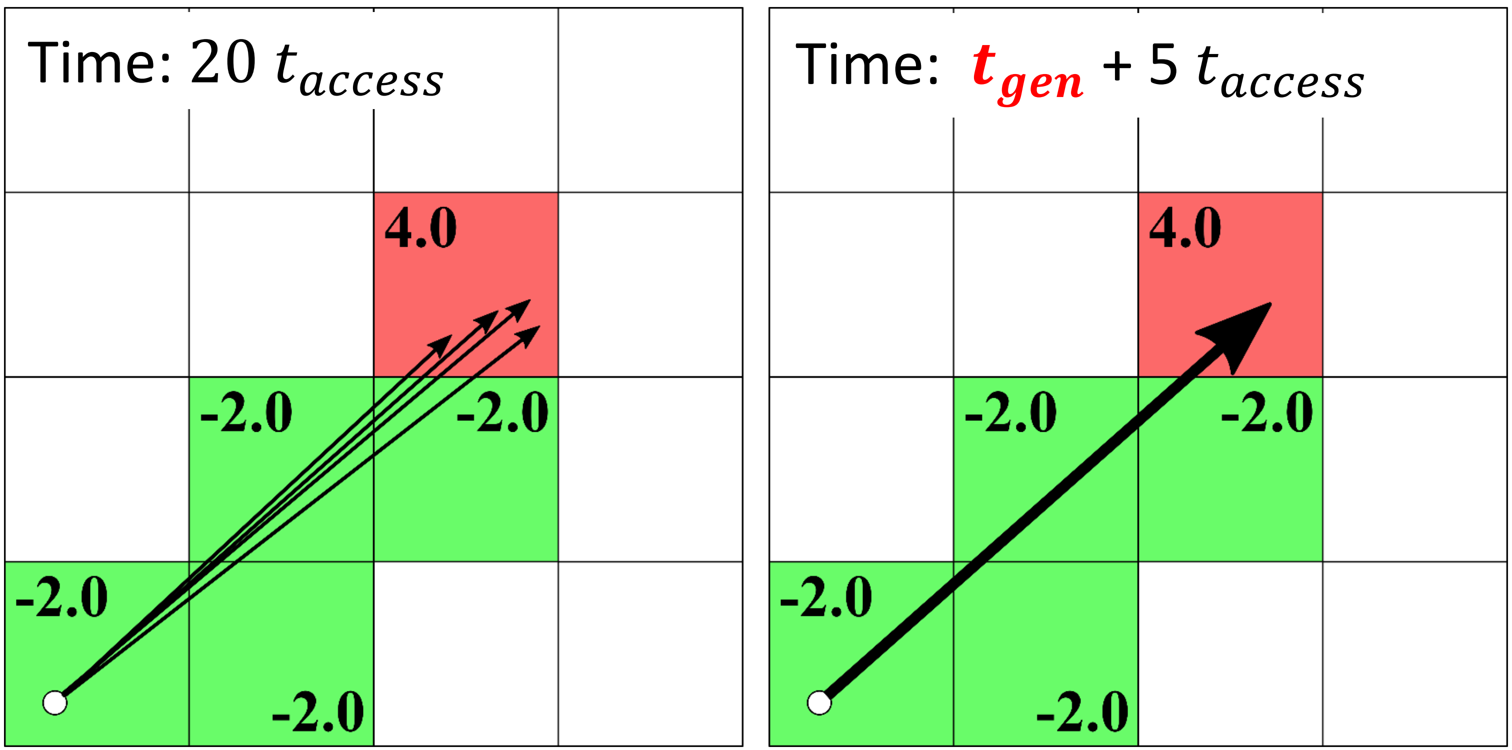
Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} = 4.0 \\ w * l_{free} = -2.0 \end{cases}$$

t_{gen}: overhead to generate super rays

Key Idea of Our Approach

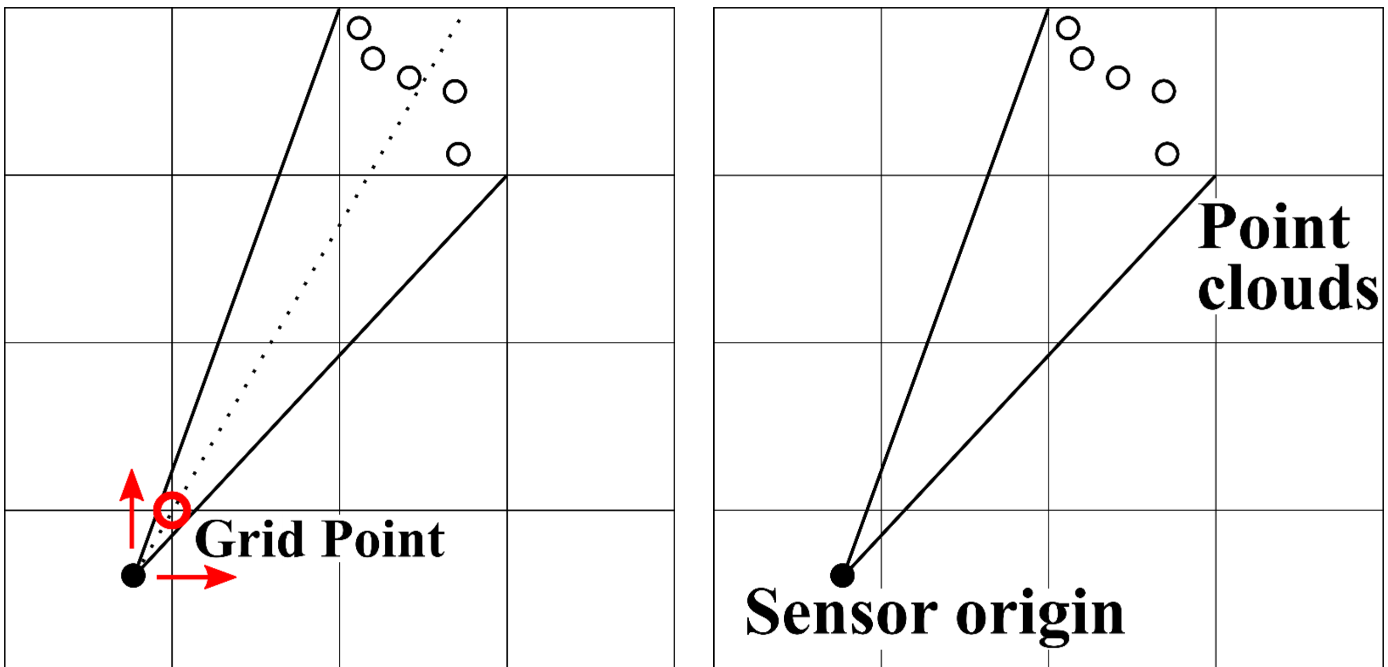
- **Benefits of our approach**
  - **Faster performance** with **the same representation accuracy**
  - Novel feature over the prior works



State-of-the-art method **Ours**

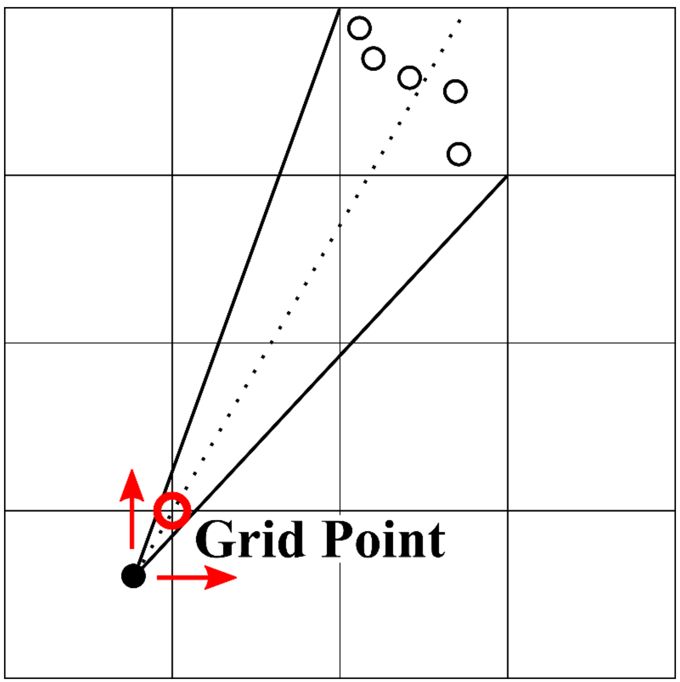
Generate Super Rays

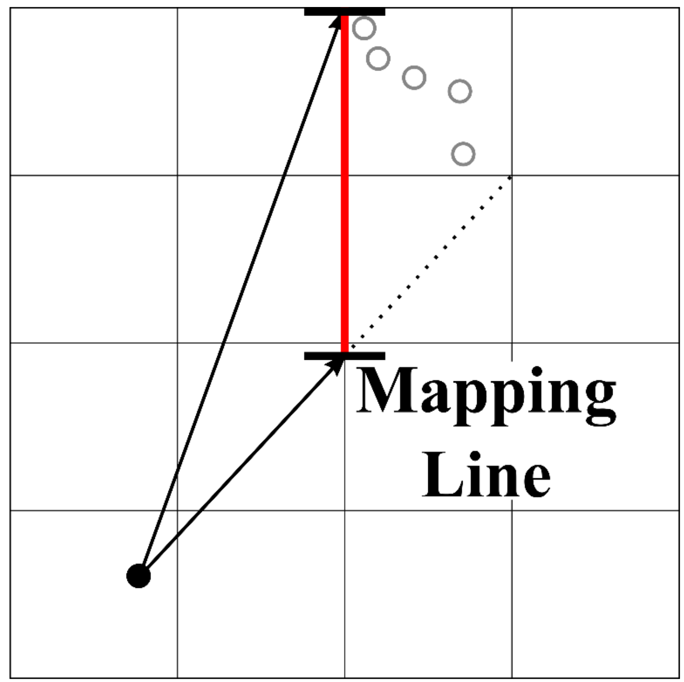
- **1. Generate a mapping line**
  - Define regions where rays **traverse the same cells**
  - Traversal patterns of cells differ along **grid points**
  - Segments of mapping line are associated to the regions



Generate Super Rays

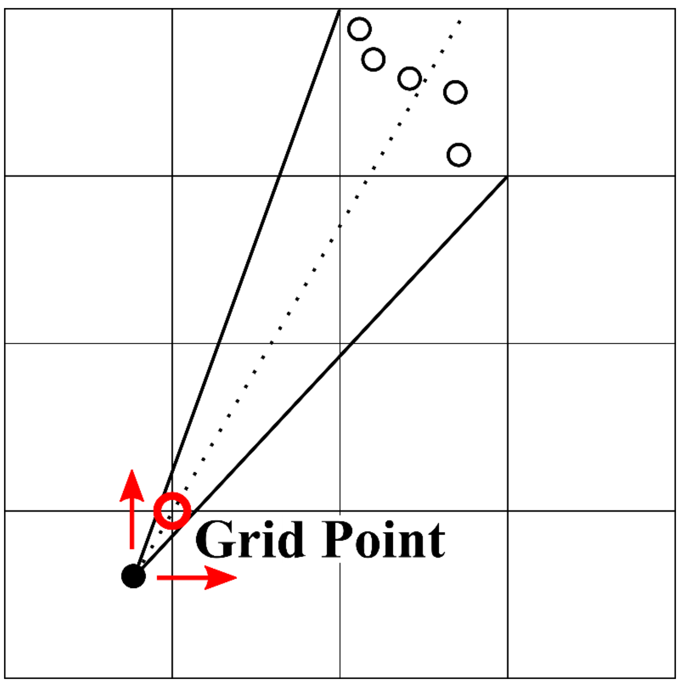
- **1. Generate a mapping line**
  - Define regions where rays **traverse the same cells**
  - Traversal patterns of cells differ along **grid points**
  - Segments of mapping line are associated to the regions

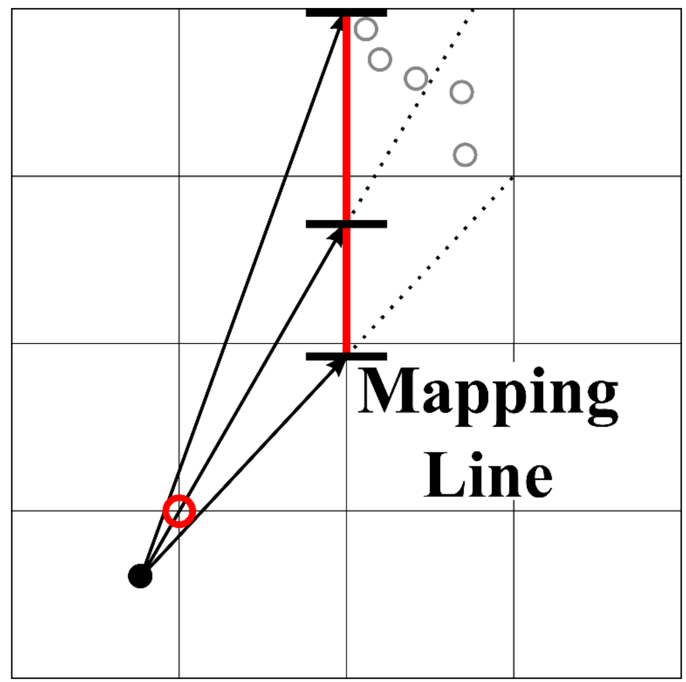




Generate Super Rays

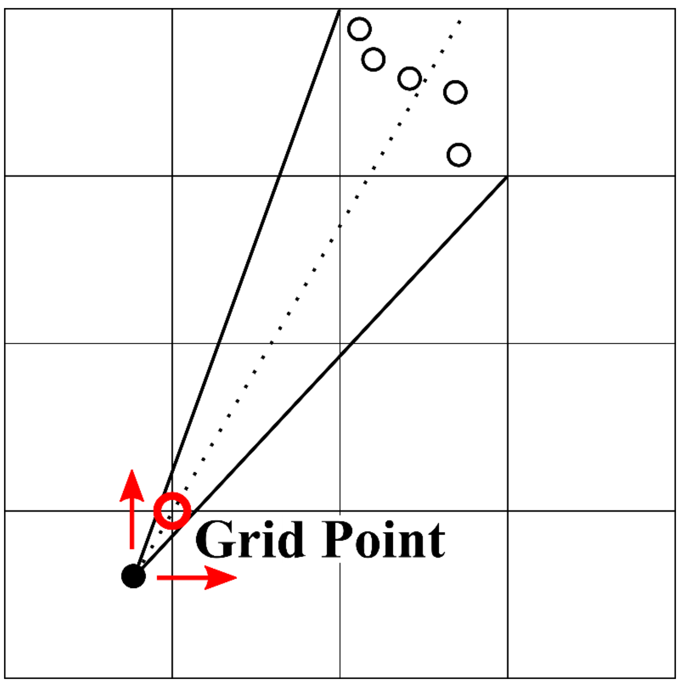
- **1. Generate a mapping line**
  - Define regions where rays **traverse the same cells**
  - Traversal patterns of cells differ along **grid points**
  - Segments of mapping line are associated to the regions

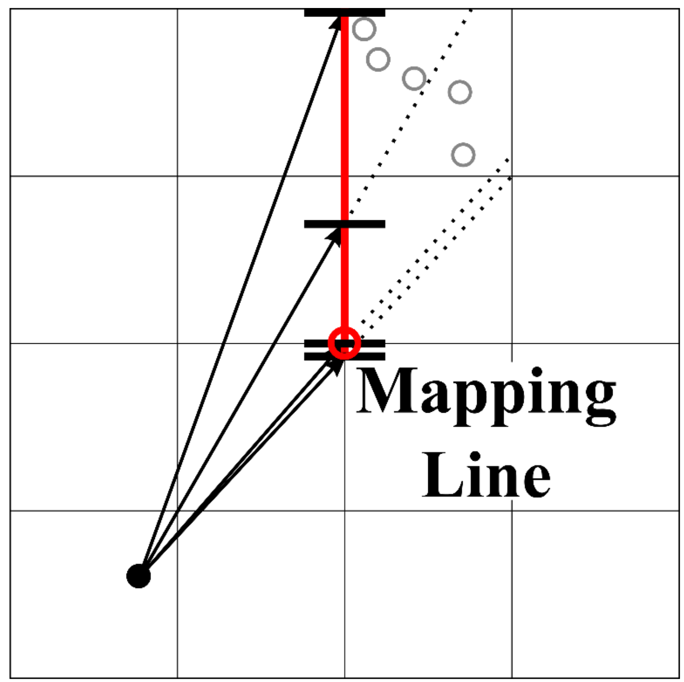




Generate Super Rays

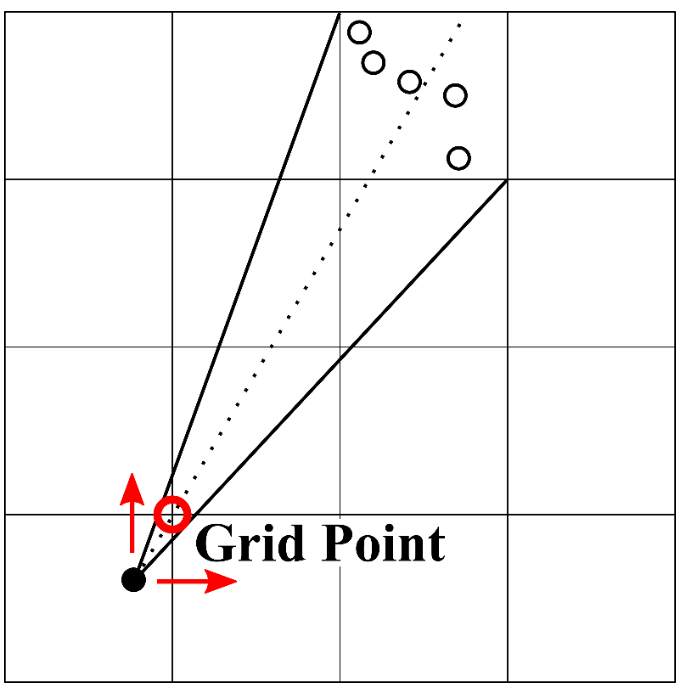
- **1. Generate a mapping line**
  - Define regions where rays **traverse the same cells**
  - Traversal patterns of cells differ along **grid points**
  - Segments of mapping line are associated to the regions

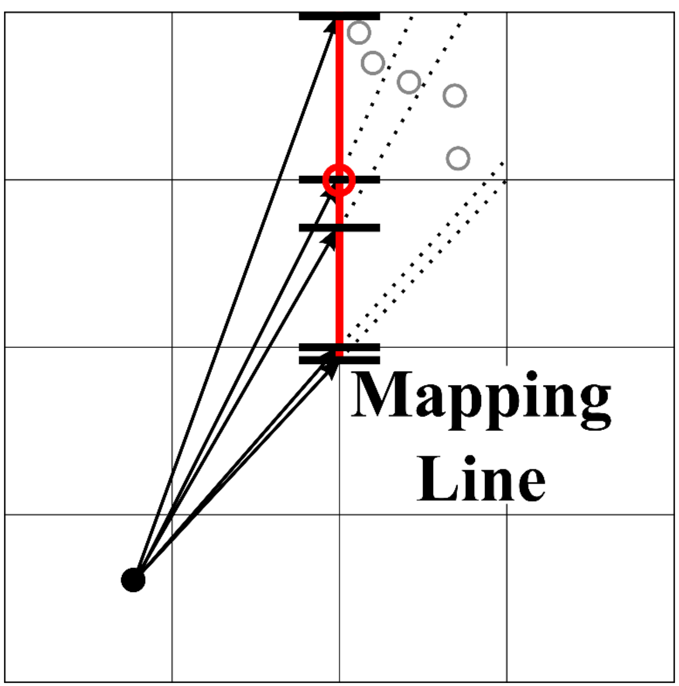




Generate Super Rays

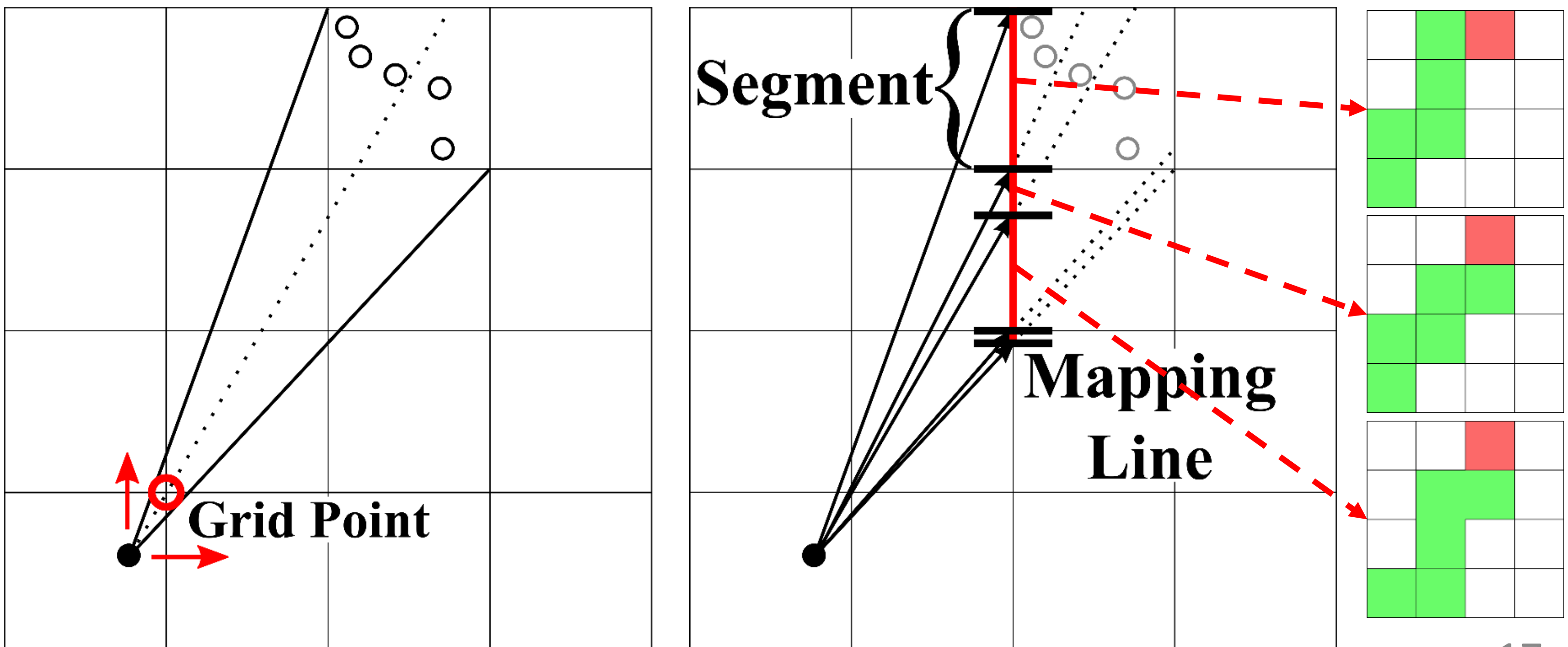
- **1. Generate a mapping line**
  - Define regions where rays **traverse the same cells**
  - Traversal patterns of cells differ along **grid points**
  - Segments of mapping line are associated to the regions





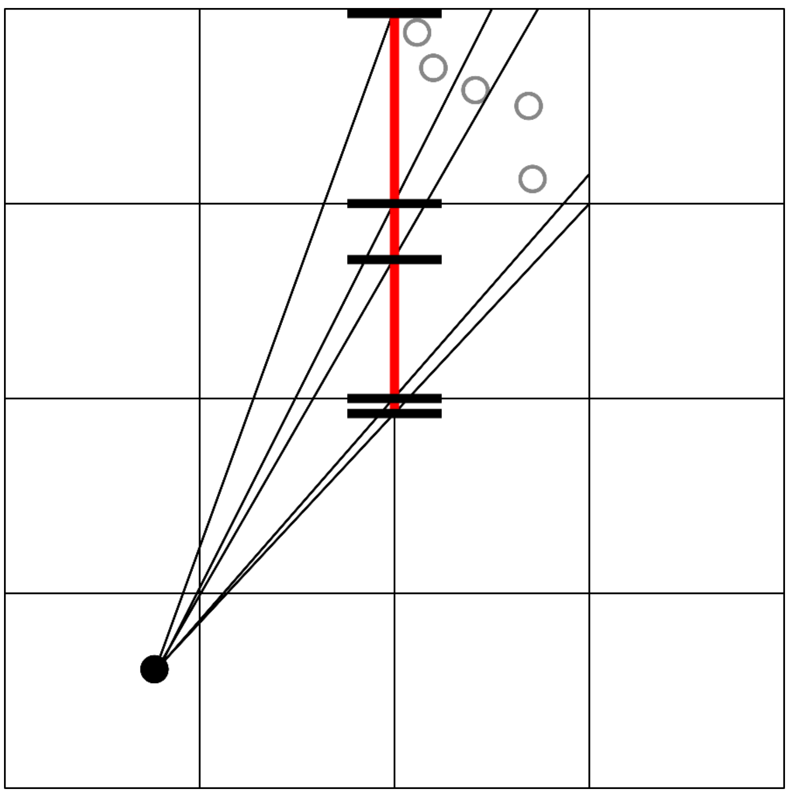
Generate Super Rays

- **1. Generate a mapping line**
  - Define regions where rays **traverse the same cells**
  - Traversal patterns of cells differ along **grid points**
  - Segments of mapping line are associated to the regions



Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



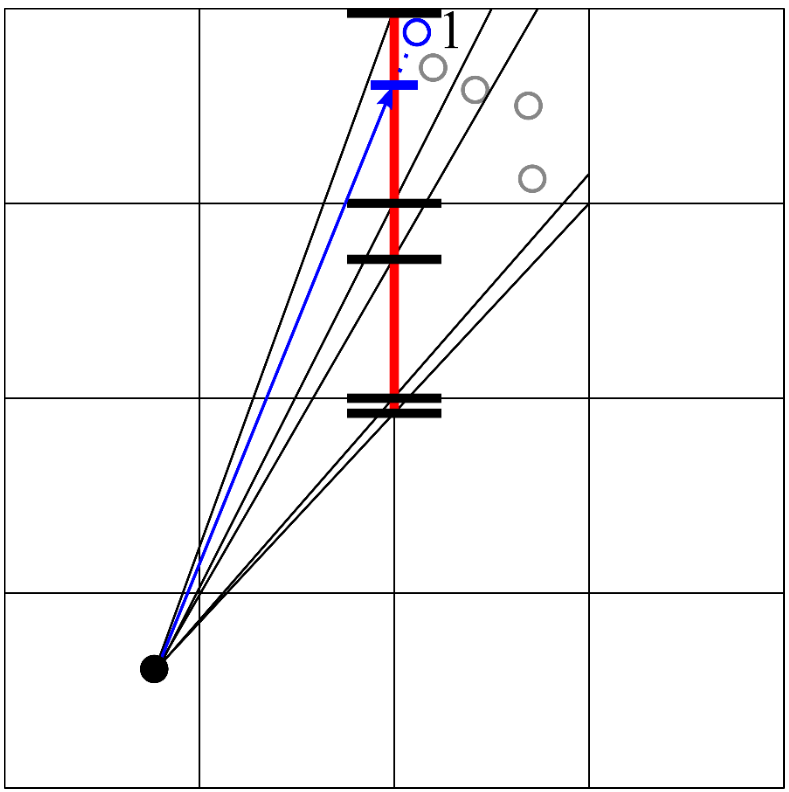
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



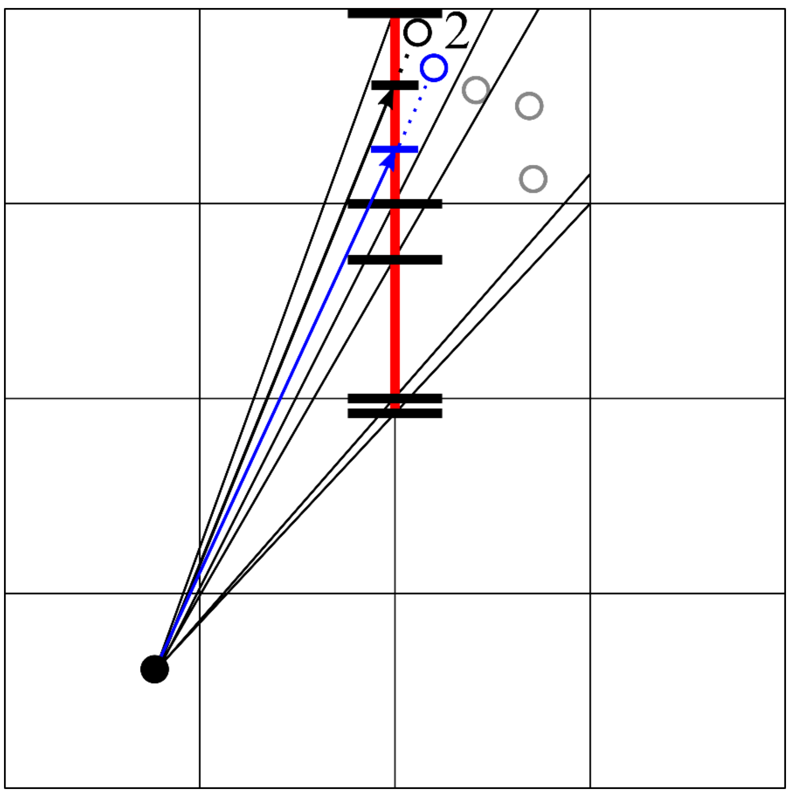
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



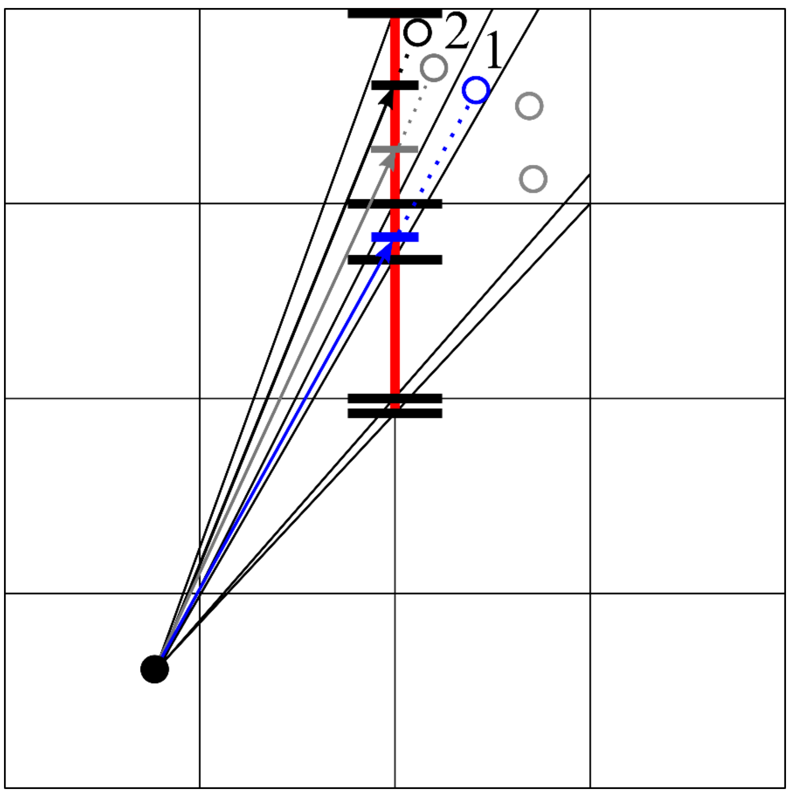
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



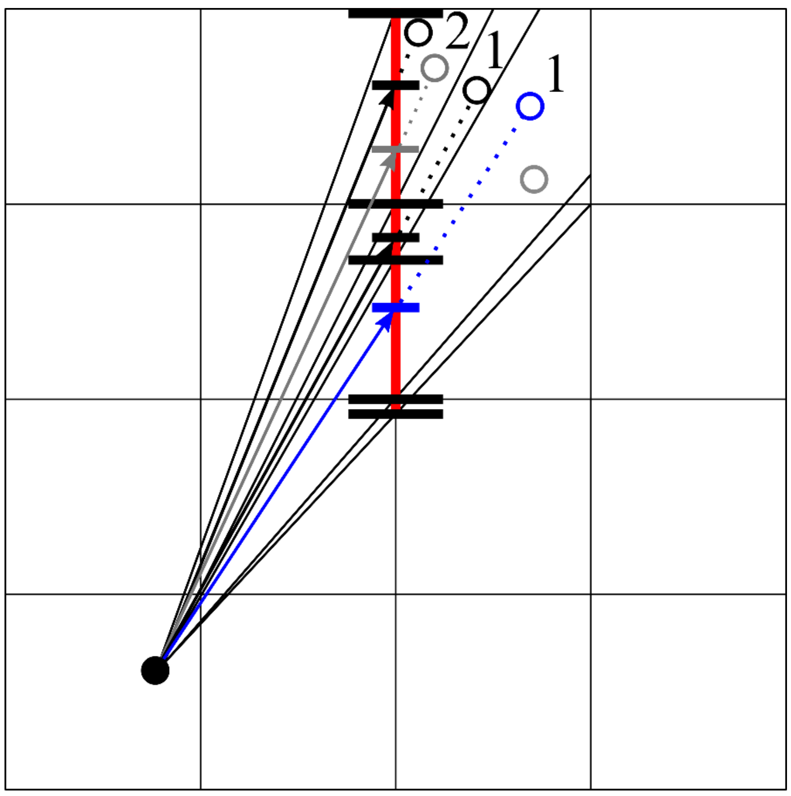
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



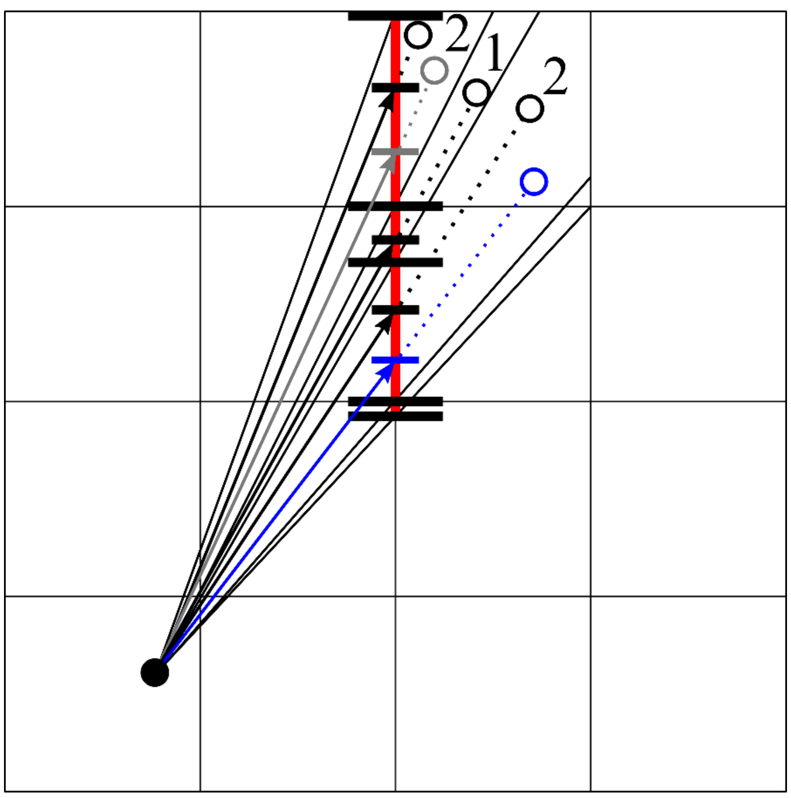
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



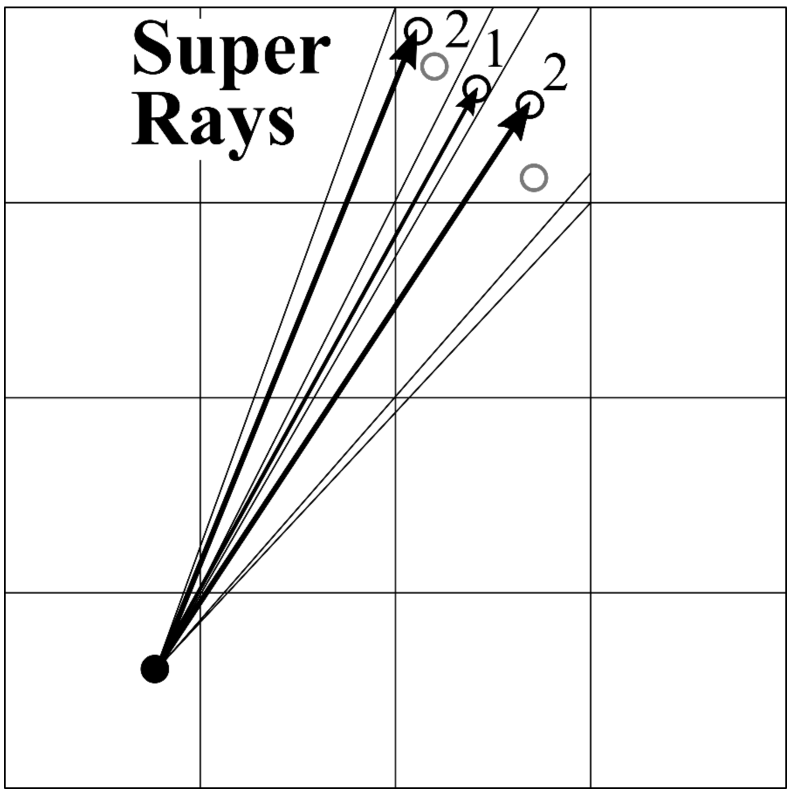
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(\, n \mid z_t \,) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Generate Super Rays

- **2. Generate super rays using mapping line**
  - Map points to a segment of the mapping line



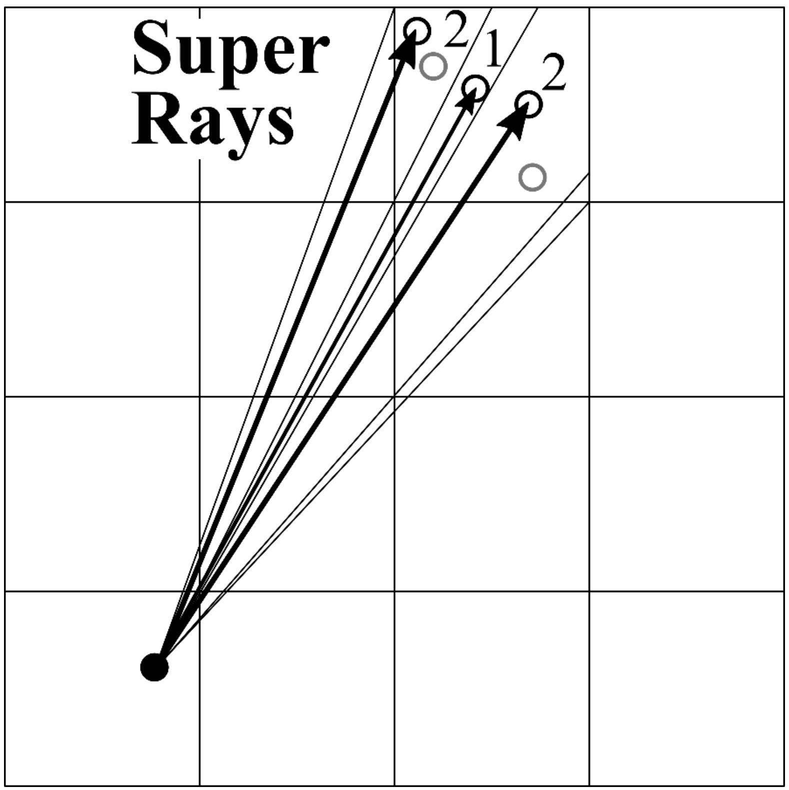
- The numbers in frustums represent the **weight *w***

Weighted measurement

$$L(n \mid z_t) = \begin{cases} w * l_{occ} \\ w * l_{free} \end{cases}$$

Update Maps using Super Rays

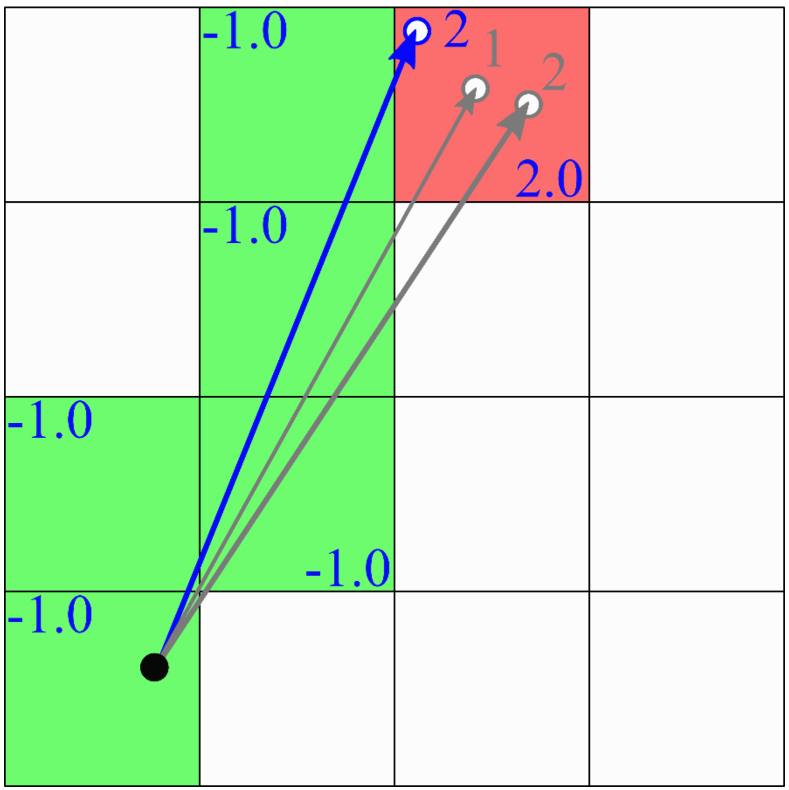
- **3. Update super rays to map representation**
  - Update super rays instead of point clouds



- Our method builds the occupancy map **faster** than prior work
- Our method builds **the same map** with a map generated from point clouds

Update Maps using Super Rays

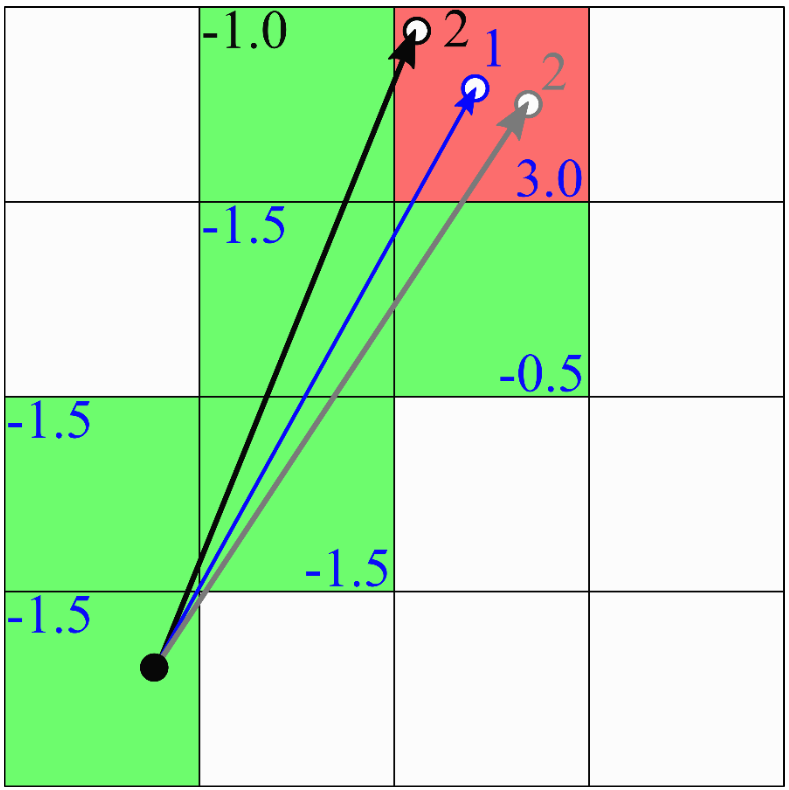
- **3. Update super rays to map representation**
  - Update super rays instead of point clouds



- Our method builds the occupancy map **faster** than prior work
- Our method builds **the same map** with a map generated from point clouds

Update Maps using Super Rays

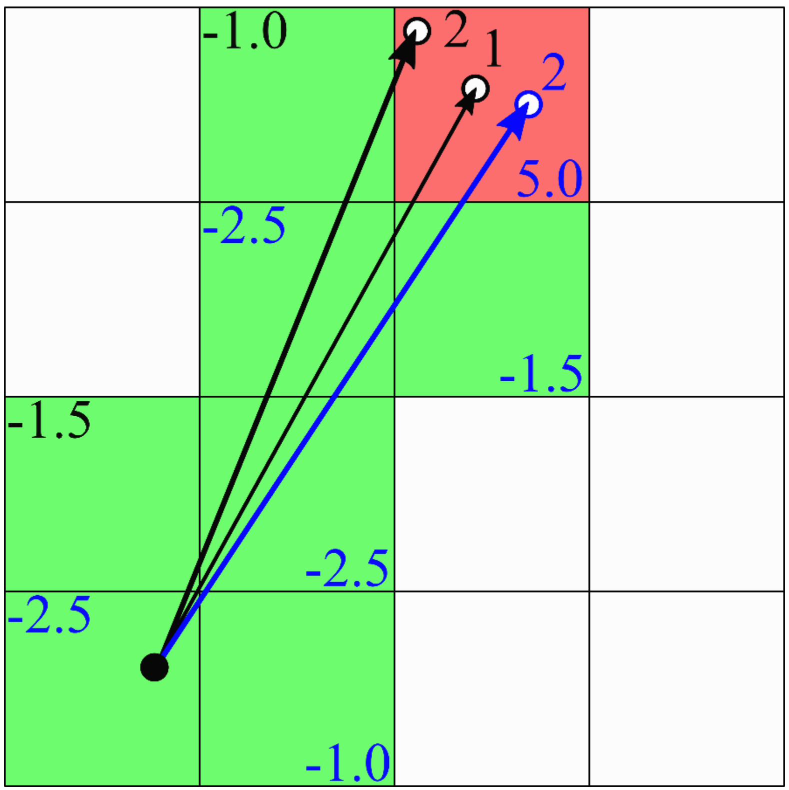
- **3. Update super rays to map representation**
  - Update super rays instead of point clouds



- Our method builds the occupancy map **faster** than prior work
- Our method builds **the same map** with a map generated from point clouds

Update Maps using Super Rays

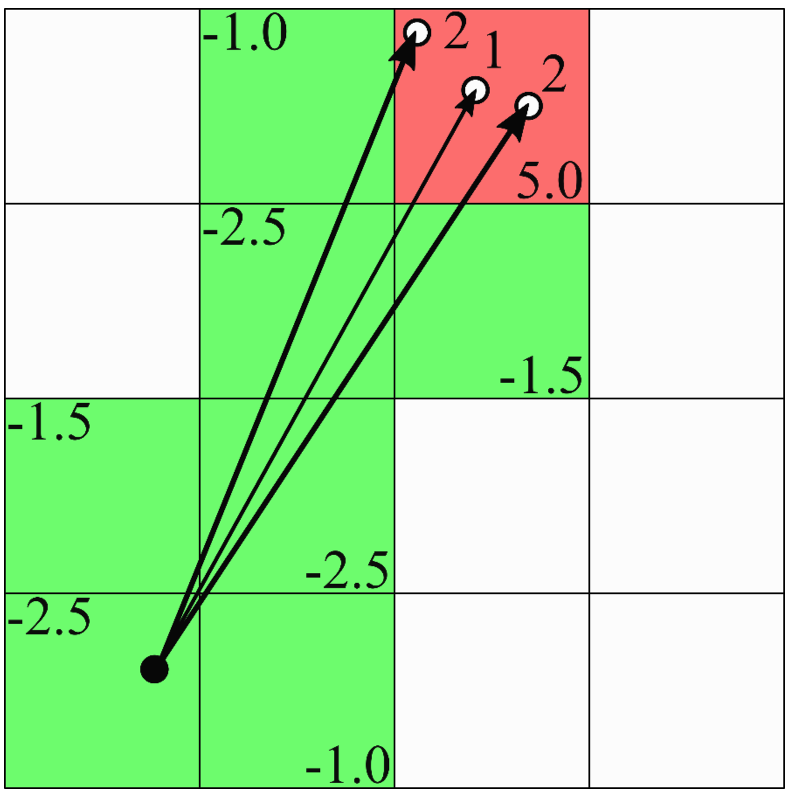
- **3. Update super rays to map representation**
  - Update super rays instead of point clouds



- Our method builds the occupancy map **faster** than prior work
- Our method builds **the same map** with a map generated from point clouds

Update Maps using Super Rays

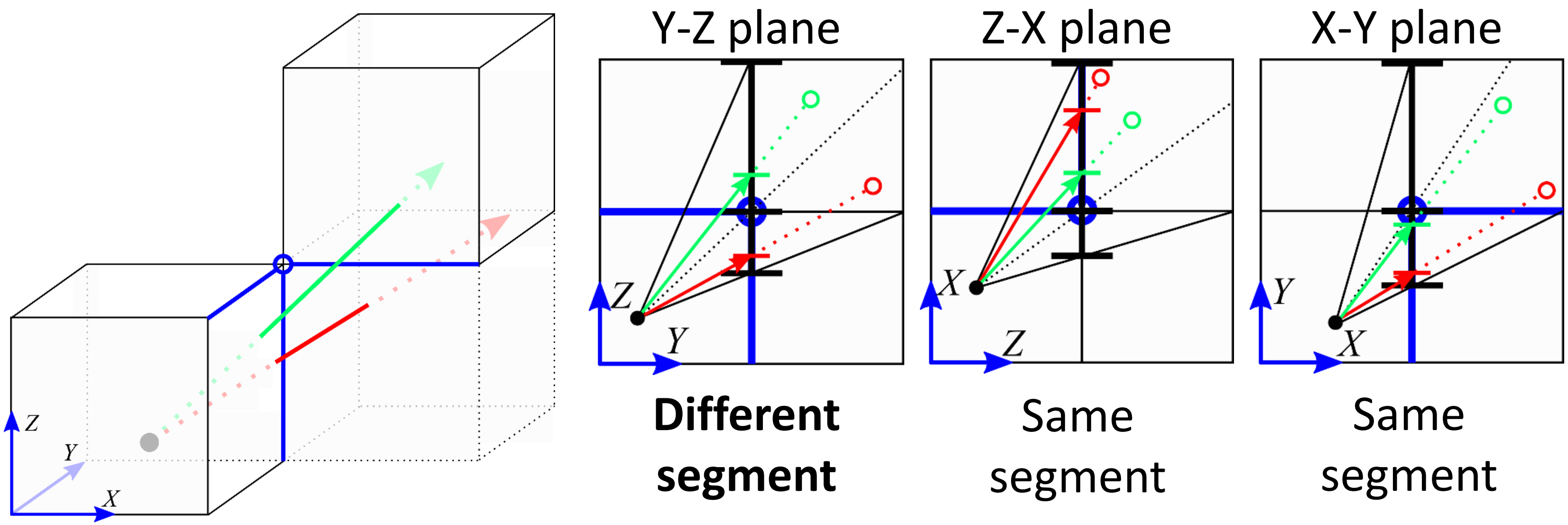
- **3. Update super rays to map representation**
  - Update super rays instead of point clouds



- Our method builds the occupancy map **faster** than prior work
- Our method builds **the same map** with a map generated from point clouds

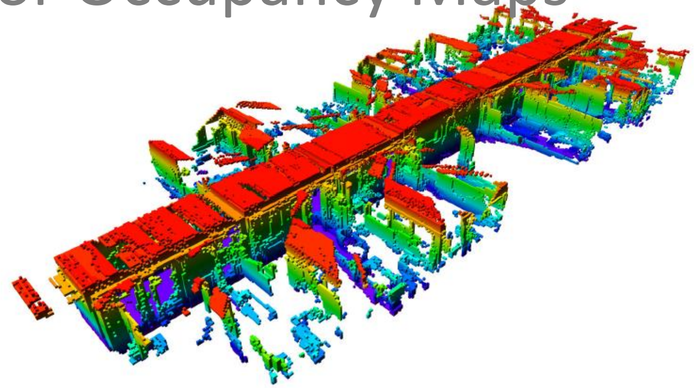
Generate Super Rays

- **Extend 2-D case to 3-D case**
  - Traversal patterns of cells differ along **edges of grid points**
  - Solve the complex 3-D problem using three simple 2-D problems (**three mapping lines**)

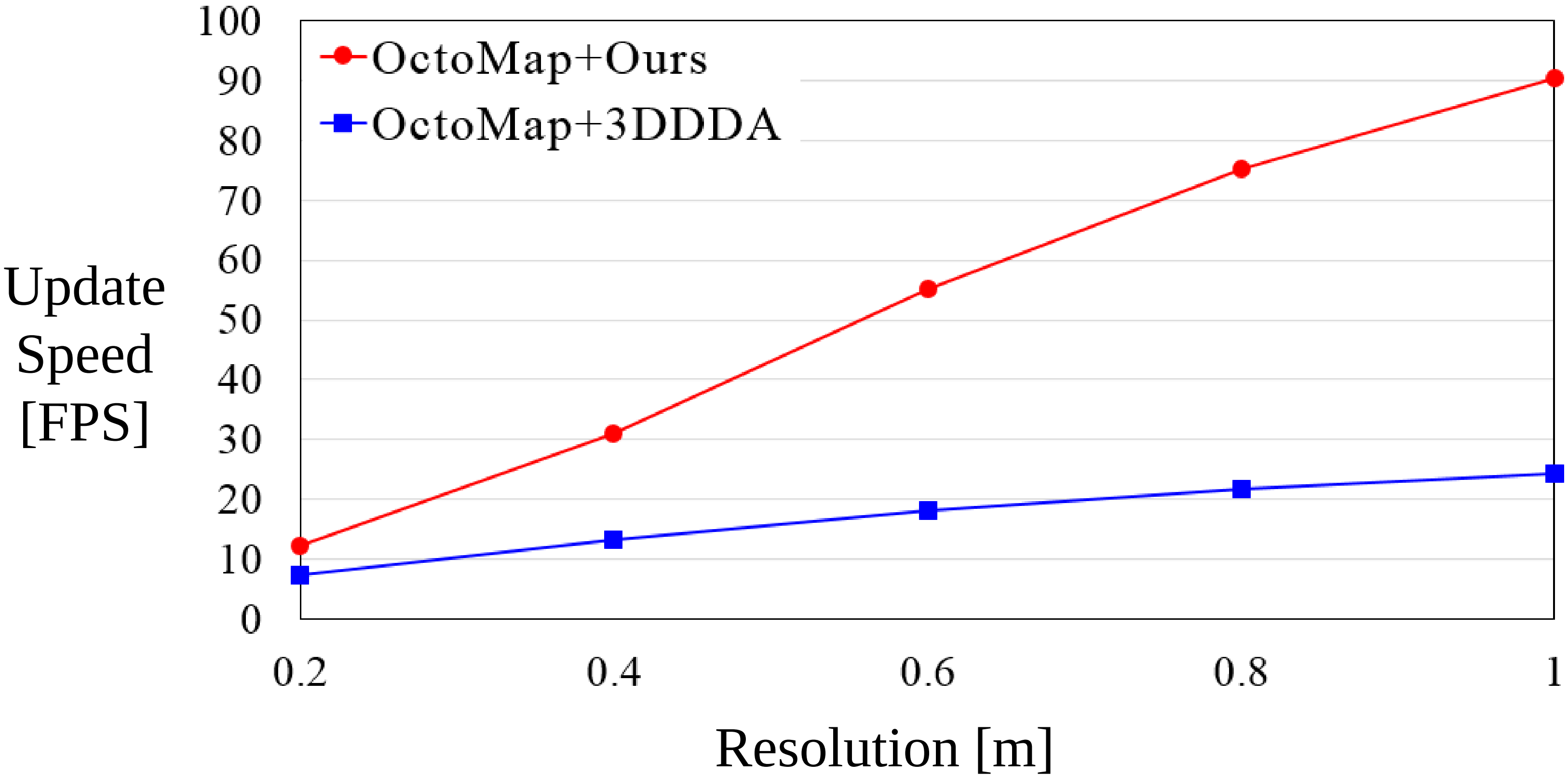


An example of generating two super rays in 3-D

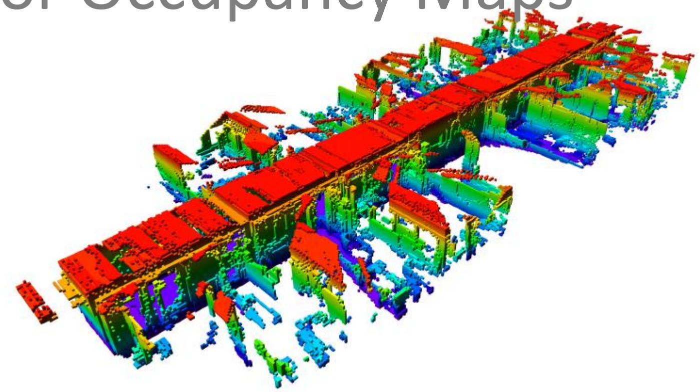
Result - Indoor



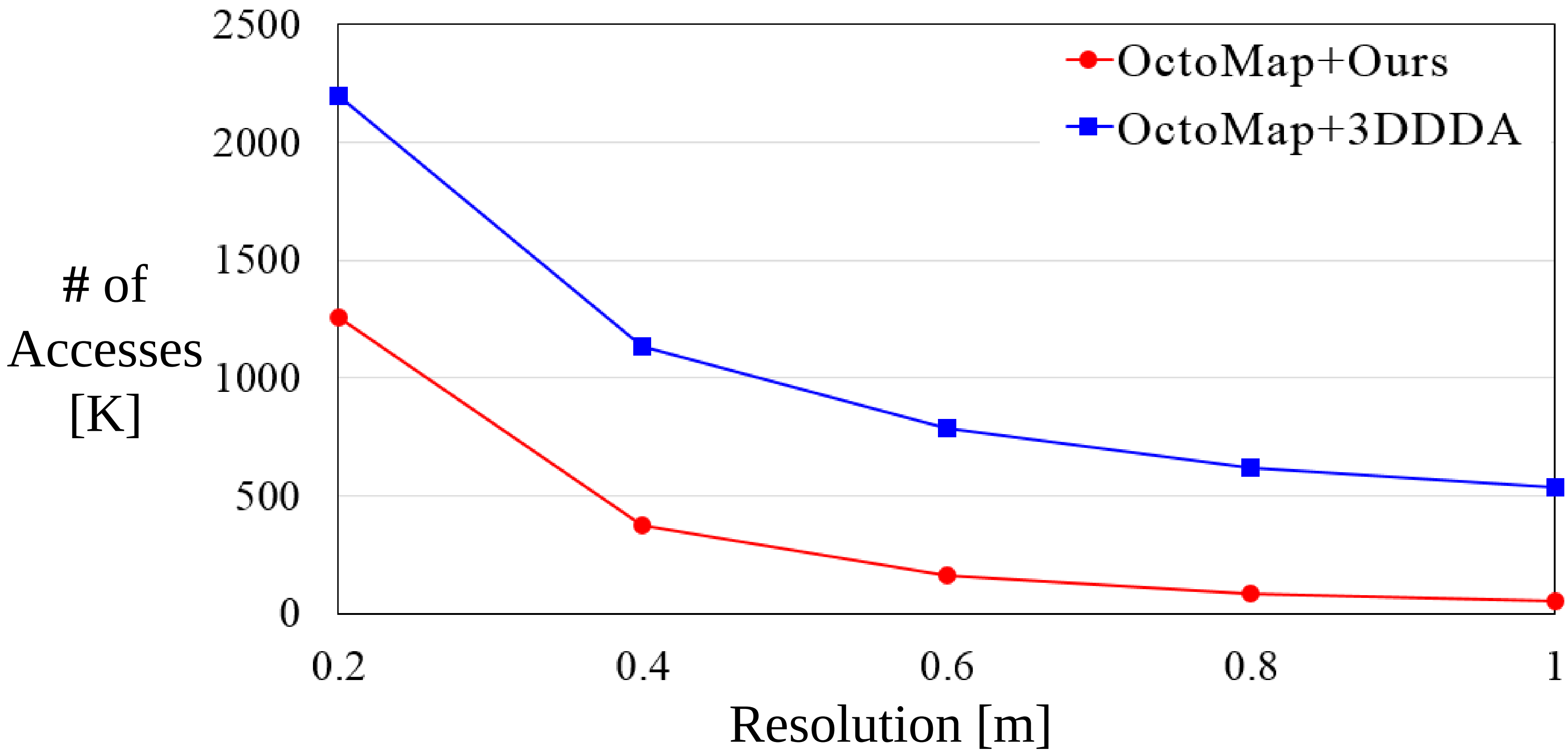
- **Update Speed [FPS]**
  - Our method **improves** performance on avg. **2.8 times**

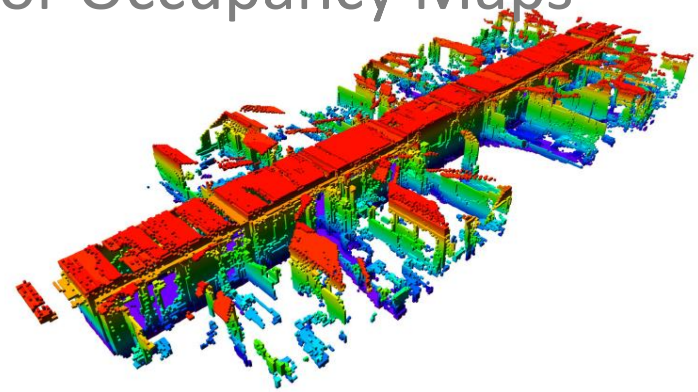


Result - Indoor



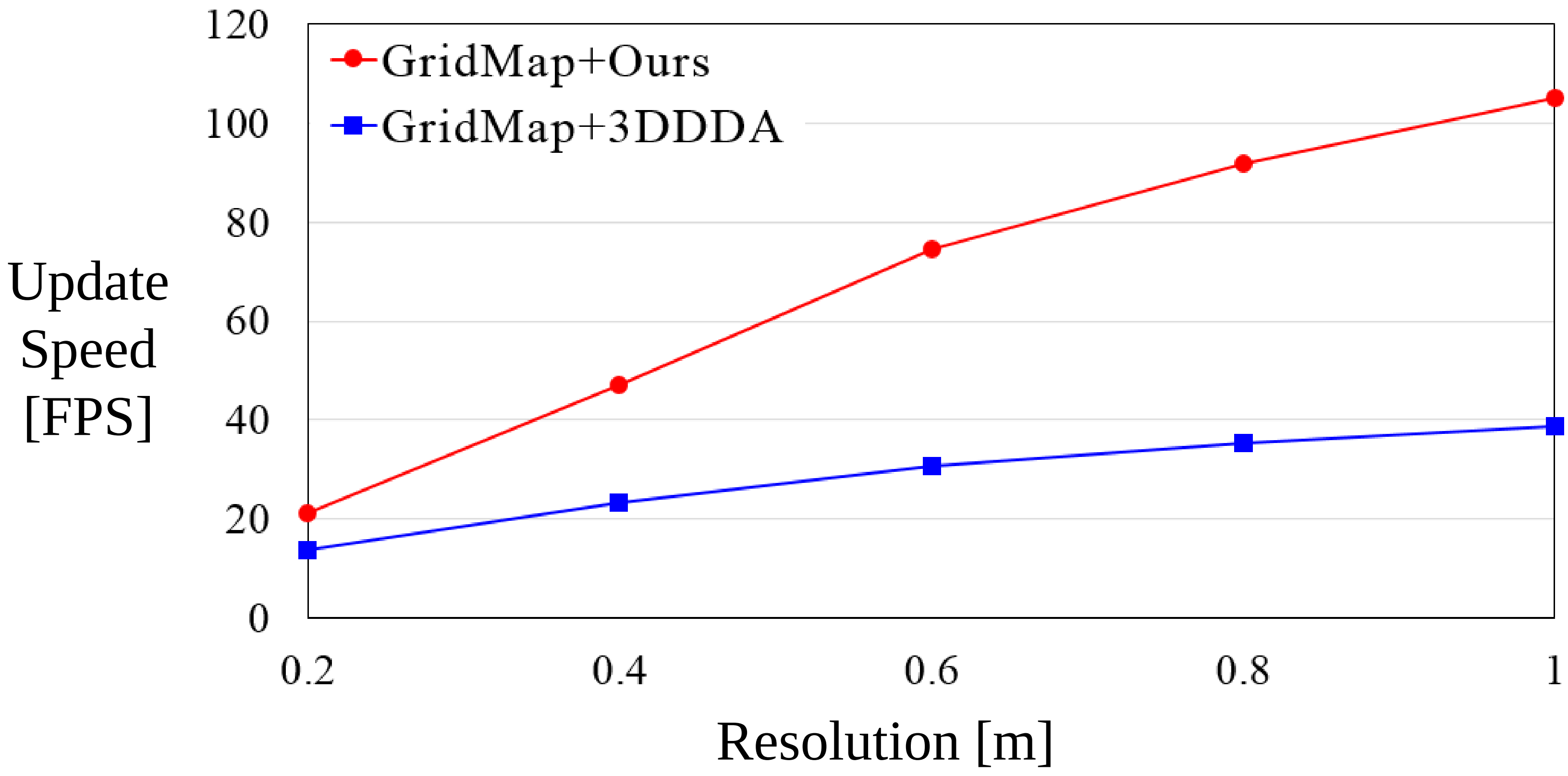
- **Avg. # of accesses [K]**
  - Our method **reduces** # of accesses to **73.1%** on avg.



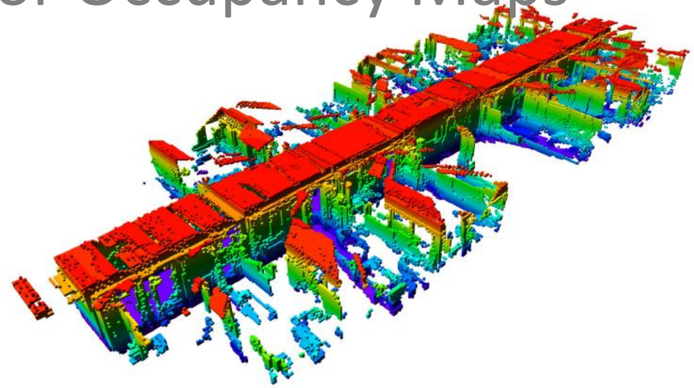


Result - Indoor

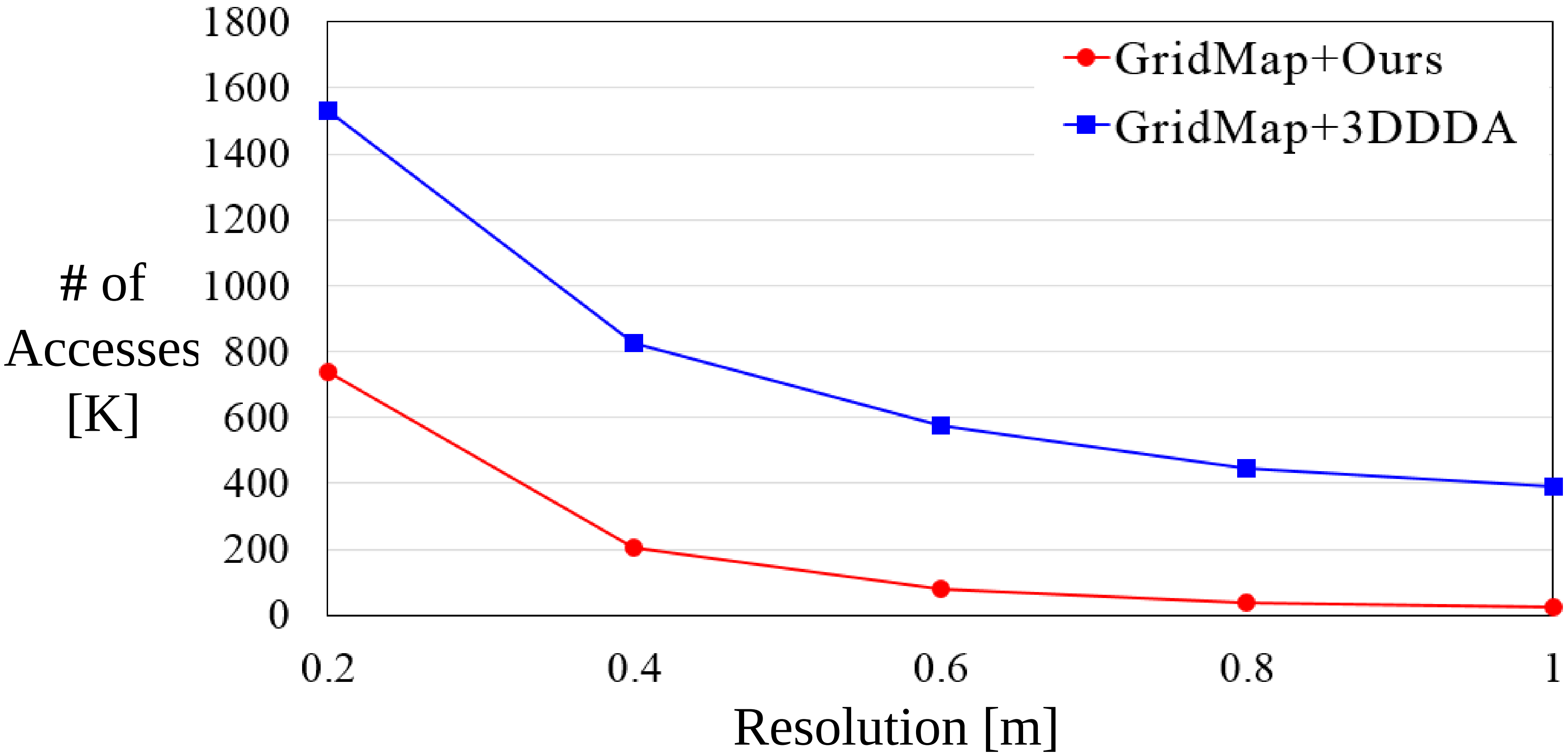
- **Update Speed [FPS]**
  - Our method **improves** performance on avg. **2.3 times**



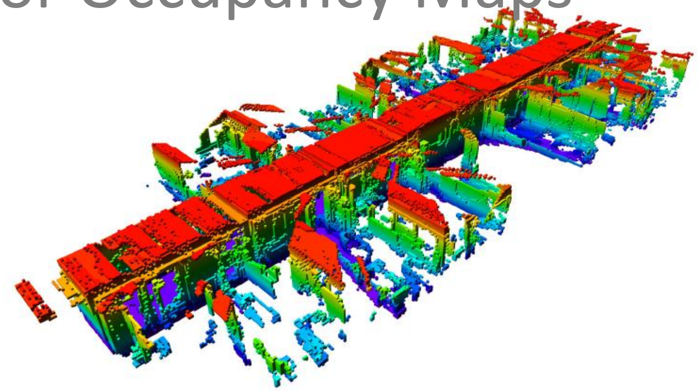
Result - Indoor



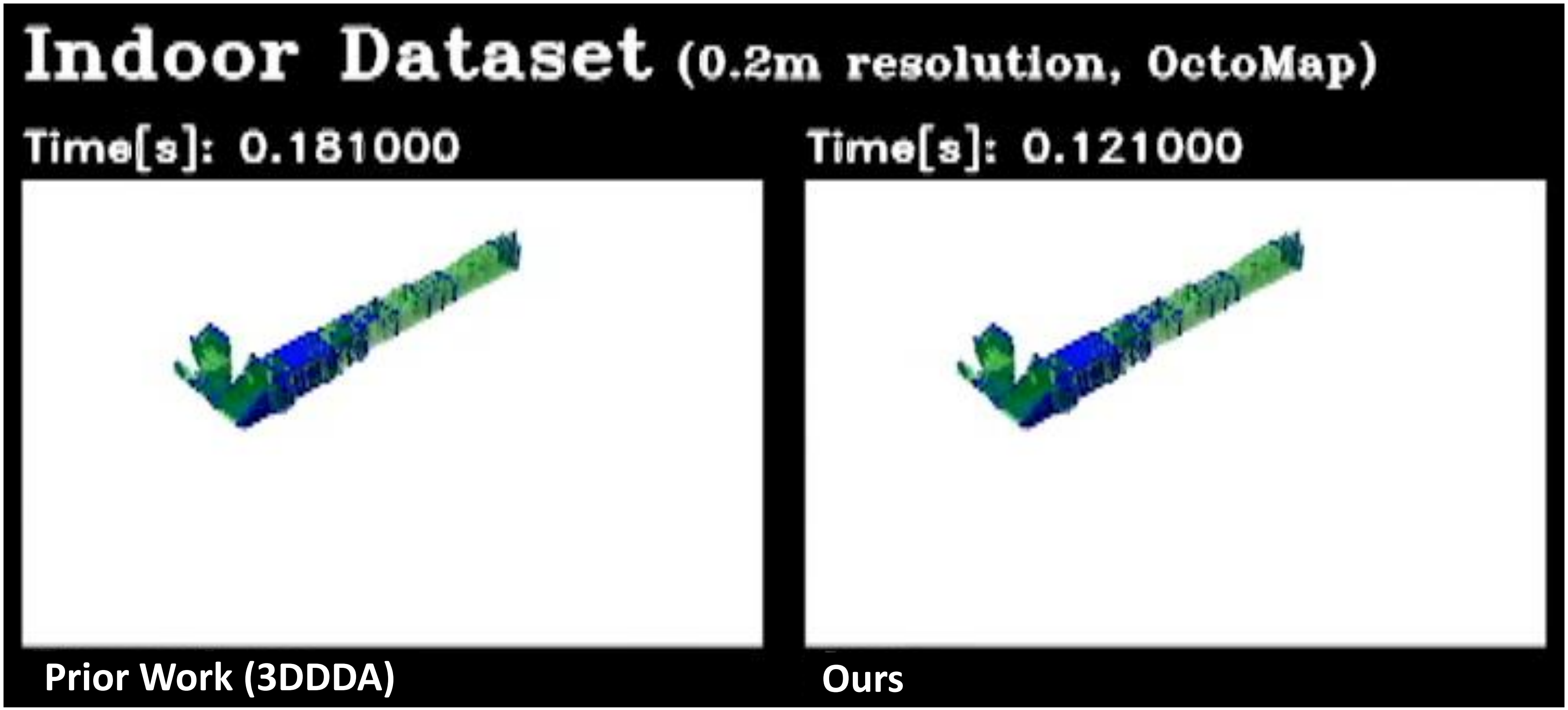
- **Avg. # of accesses [K]**
  - Our method **reduces** # of accesses to **79.7%** on avg.



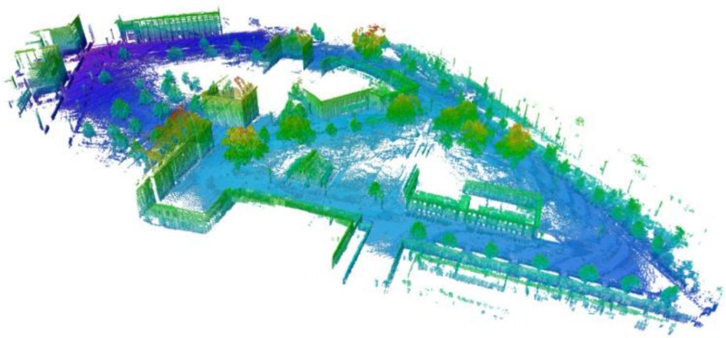
Result - Indoor



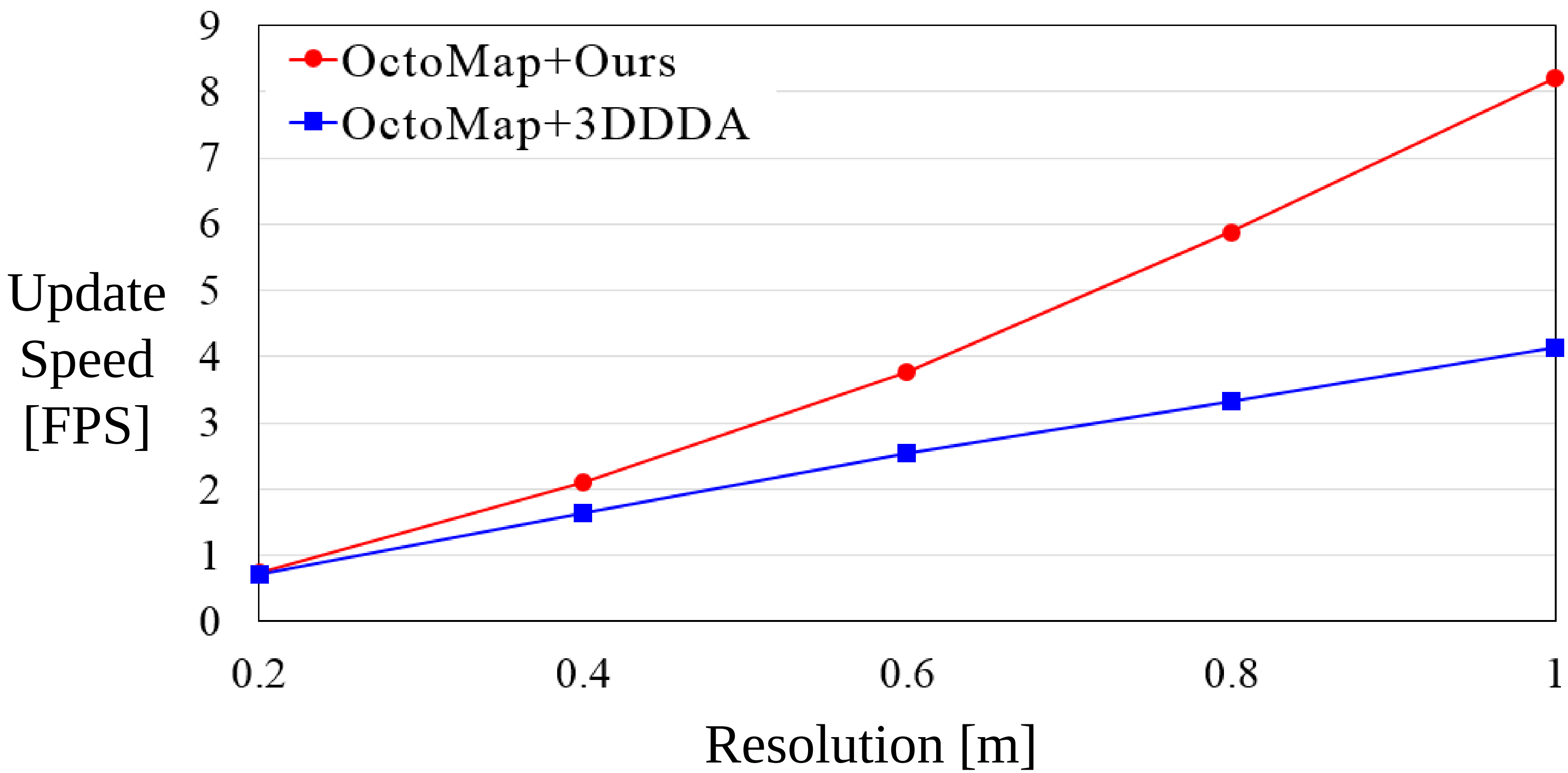
- **Update Speed [FPS]**



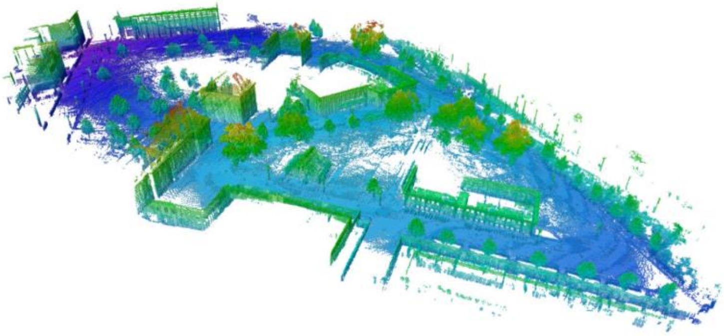
Result - Outdoor



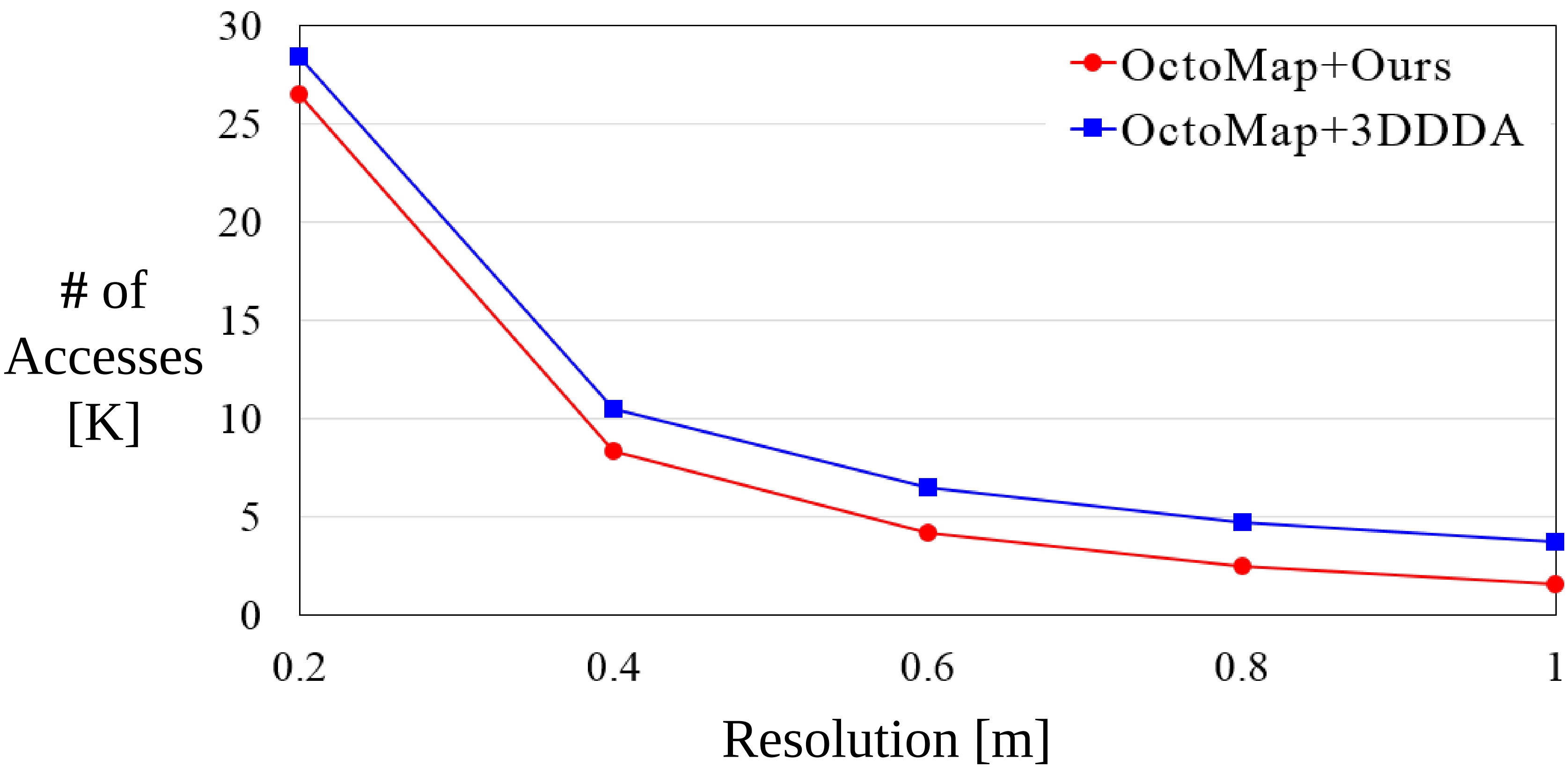
- **Update Speed [FPS]**
  - Our method **improves** performance on avg. **1.5 times**



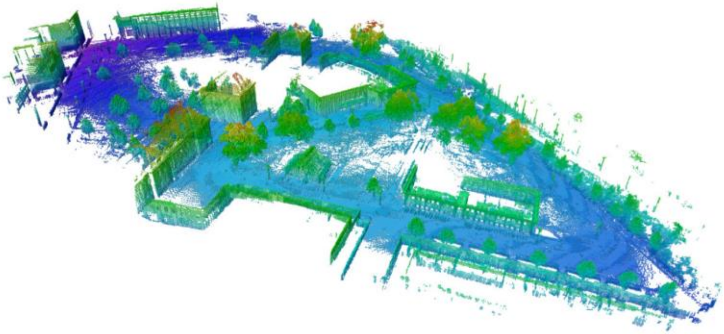
Result - Outdoor



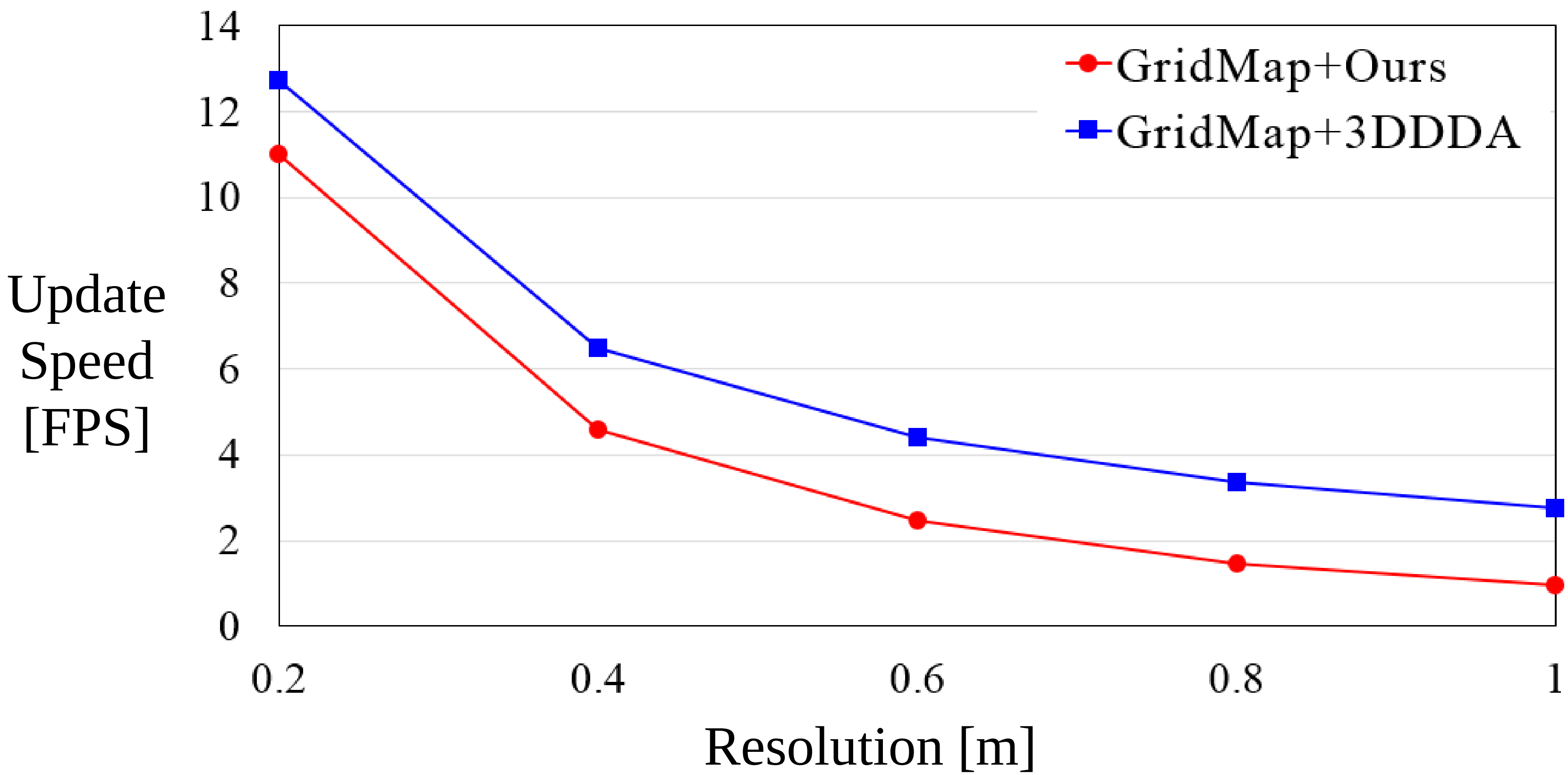
- **Avg. # of accesses [M]**
  - Our method **reduces** # of accesses to **33.3 %** on avg.



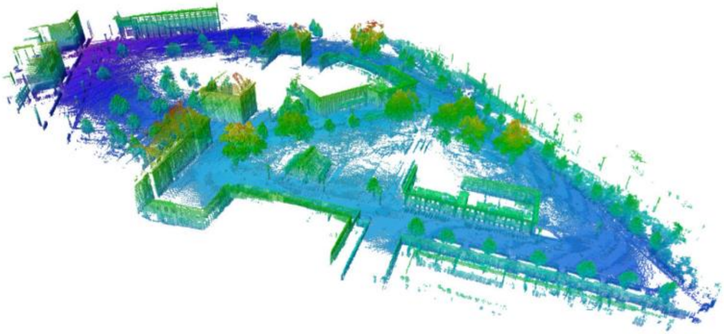
Result - Outdoor



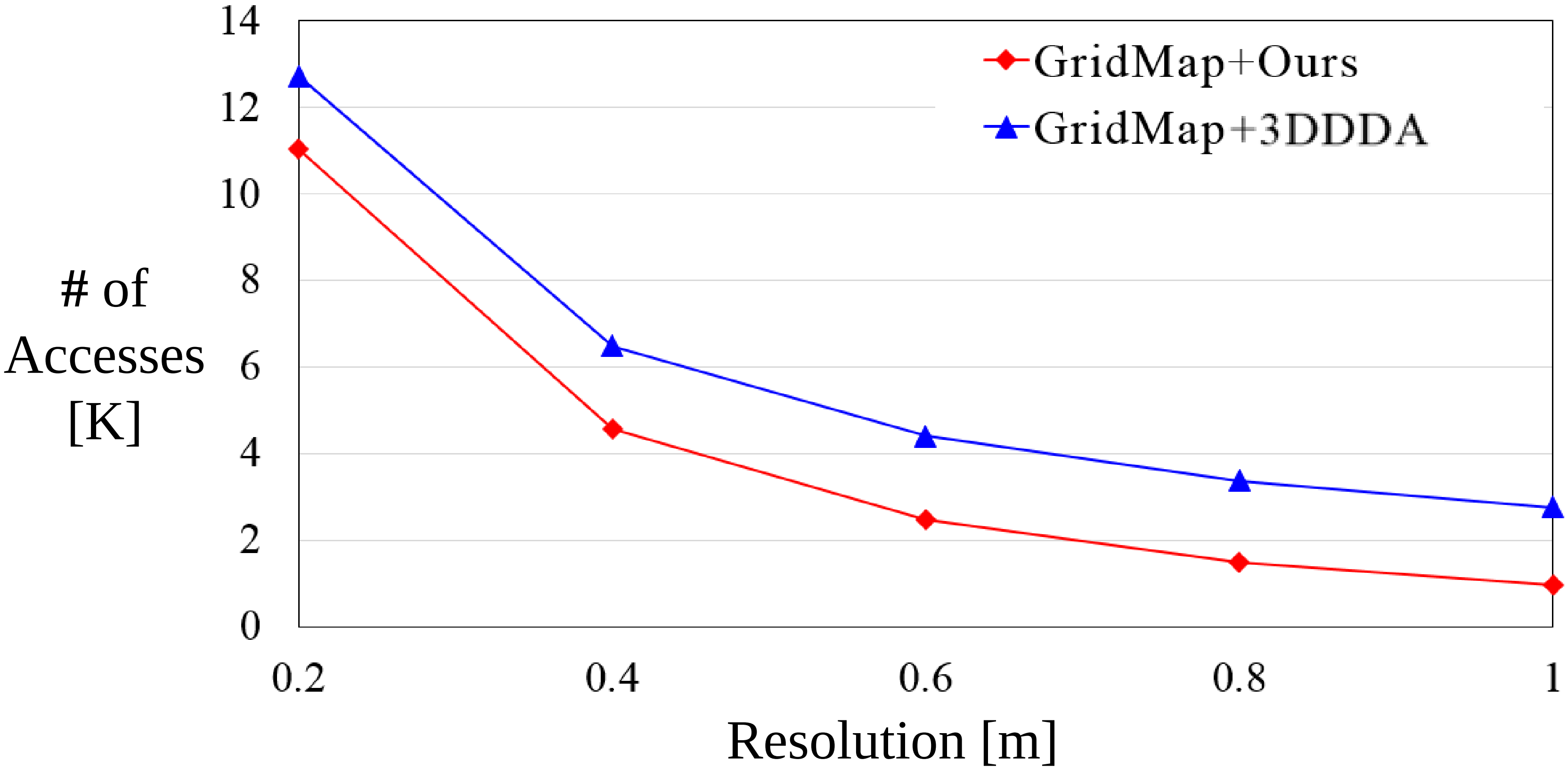
- **Update Speed [FPS]**
  - Our method **improves** performance on avg. **1.4 times**



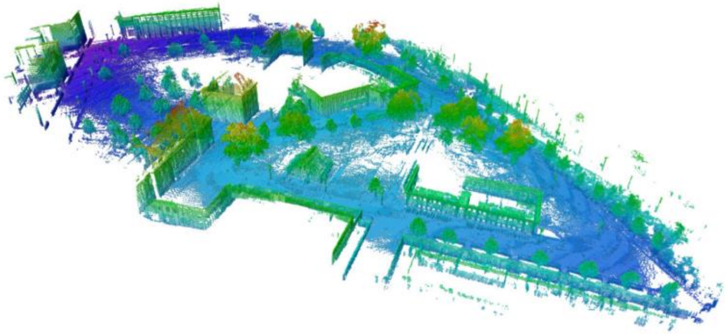
Result - Outdoor



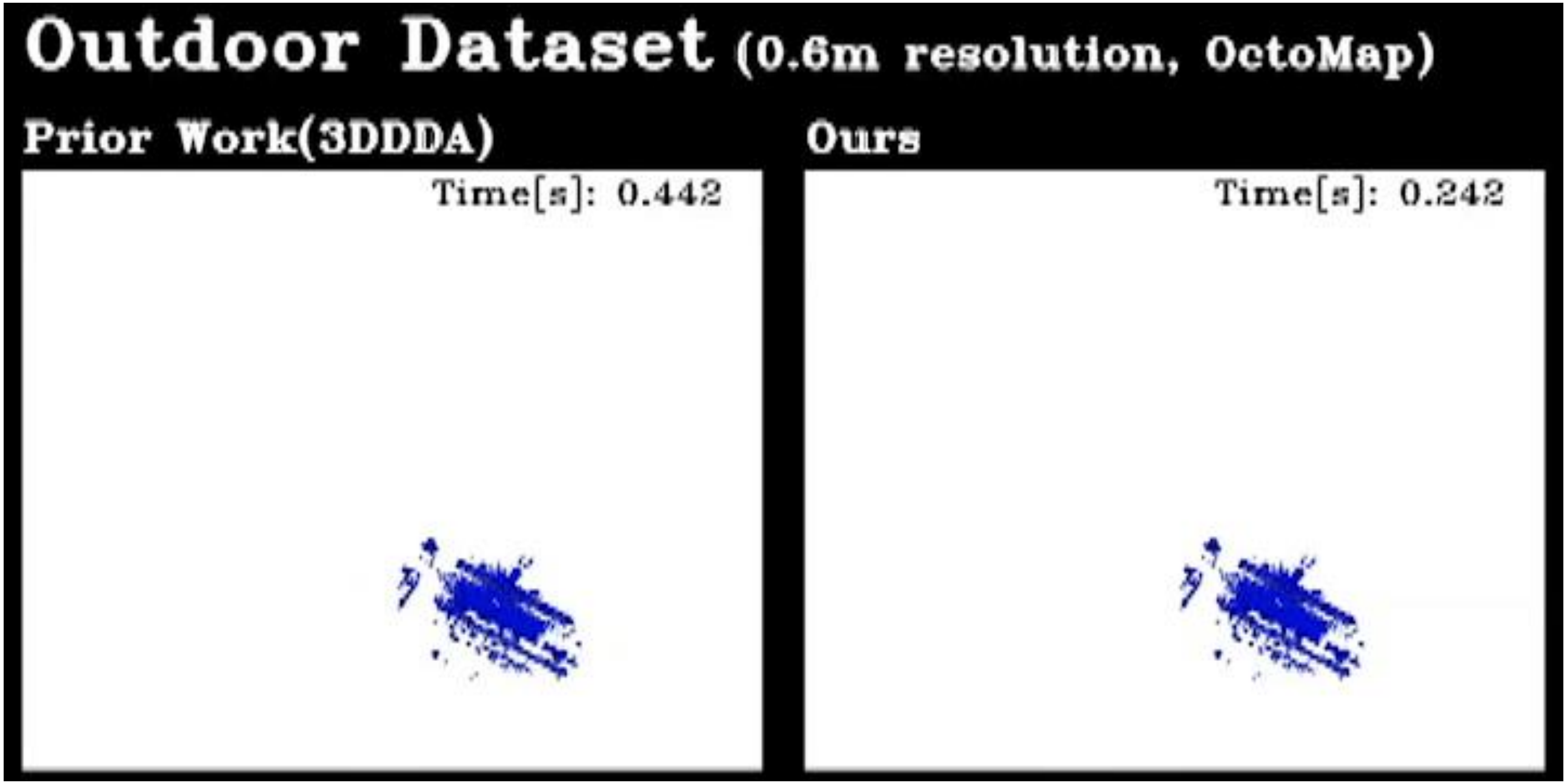
- **Avg. # of accesses [M]**
  - Our method **reduces** # of accesses to **41.7 %** on avg.



Result - Outdoor



- **Update Speed [FPS]**



Conclusion

- **Super Ray based Updates for Occupancy Maps**
  - **Super Ray** is a representative ray for set of points that traverse the same set of cells
  - **Mapping line** is an efficient method to generate super rays out of point clouds
  - Achieve **2.5 times** on average **performance improvement** over the state-of-the-art update method
  - Source code is available at http://sglab.kaist.ac.kr/projects/SuperRay

Thank you

Acknowledgements

Advisor Sung-Eui Yoon & SGLAB members

Appendix. A

- **The number of generated super rays**

# of Points	Indoor [89,446]		Outdoor [247,817]	
Evaluation	# of Super Rays	# of Points / Super Ray	# of Super Rays	# of Points / Super Ray
0.2m	25064	3.6	150453	1.6
0.4m	10668	8.3	102076	2.4
0.6m	5106	17.5	72191	3.4
0.8m	3072	29.1	52906	4.7
1.0m	2073	43.1	40833	6.1

Appendix. B

- **Summary Table of Result**

Indoor Dataset															
Resolution	0.2m			0.4m			0.6m			0.8m			1.0m		
Evaluation	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]
OctoMap + 3DDDA	7.3	0	137.6 (2195K)	13.2	0	76.3 (1132K)	18.1	0	55.6 (788K)	21.7	0	46.2 (619K)	24.4	0	41.1 (538K)
OctoMap + Ours	**12.1**	16.6	67.7 (1260K)	**31.1**	12.6	20.2 (373K)	**55.2**	10.2	8.2 (160K)	**75.2**	9.2	4.3 (88K)	**90.5**	8.6	2.5 (52K)
GridMap + 3DDDA	13.6	0	74.0 (1531K)	23.4	0	43.0 (826K)	30.6	0	32.9 (576K)	35.4	0	28.3 (448K)	38.8	0	25.8 (392K)
GridMap + Ours	**21.0**	16.3	32.1 (739K)	**46.9**	12.3	9.3 (205K)	**74.7**	9.9	3.6 (80K)	**91.8**	9.1	1.9 (40K)	**105.2**	8.4	1.2 (23K)

Outdoor Dataset															
Resolution	0.2m			0.4m			0.6m			0.8m			1.0m		
Evaluation	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]	**FPS**	Proc. [ms]	Update [ms]
OctoMap + 3DDDA	0.7	0	1516.1 (28.4M)	1.6	0	639.5 (10.5M)	2.5	0	412.9 (6.5M)	3.3	0	314.7 (4.8M)	4.1	0	252.7 (3.8M)
OctoMap + Ours	**0.7**	68.3	1395.8 (26.5M)	**2.1**	57.0	449.1 (8.3M)	**3.8**	51.1	231.8 (4.2M)	**5.9**	44.5	137.5 (2.5M)	**8.2**	41.3	89.0 (1.6M)
GridMap + 3DDDA	1.4	0	783.1 (12.7M)	3.3	0	321.6 (6.5M)	5.1	0	207.7 (4.4M)	6.5	0	162.1 (3.4M)	7.7	0	136.1 (2.8M)
GridMap + Ours	**1.4**	65.9	708.3 (11.0M)	**4.0**	57.7	211.9 (4.6M)	**7.1**	50.2	100.8 (2.5M)	**10.2**	43.9	61.3 (1.5M)	**13.3**	40.2	39.8 (1.0M)