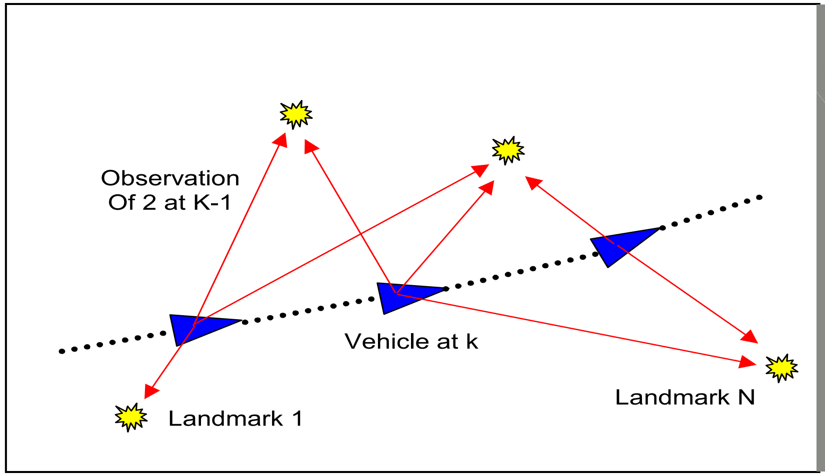
SPATIAL AWARENESS

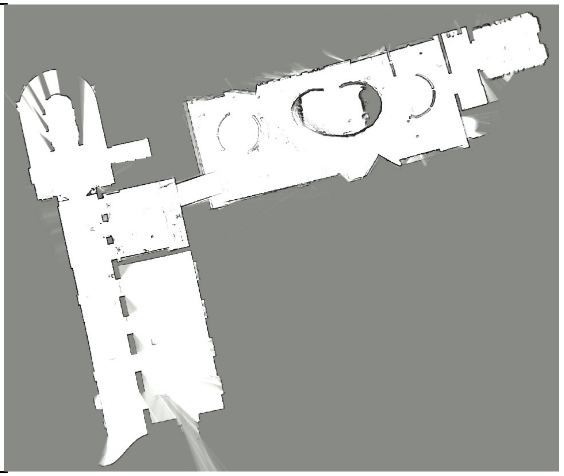
Taxonomy of Spatial Representations

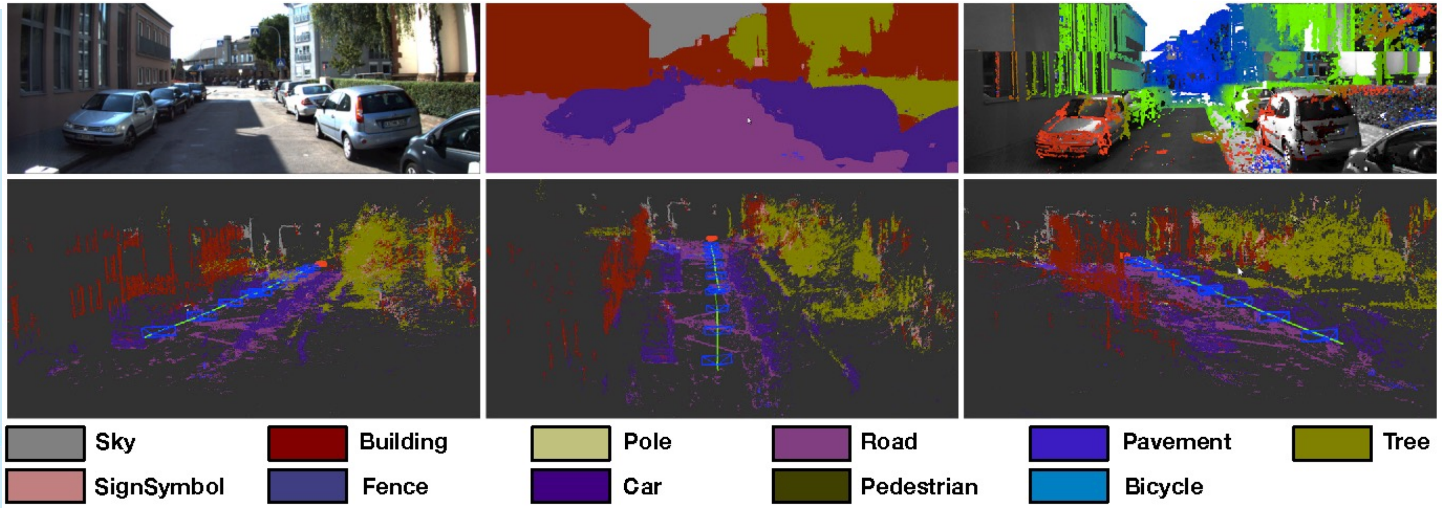
Teresa Vidal-Calleja

MAP REPRESENTATIONS

- Metric
- Topological
- Semantic





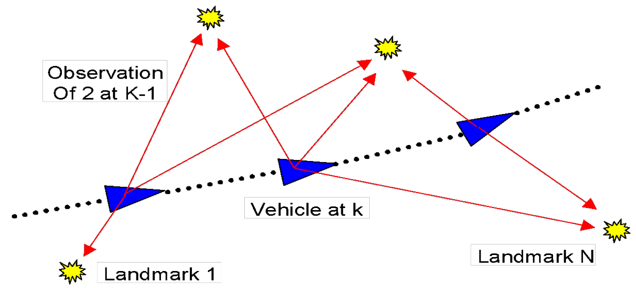


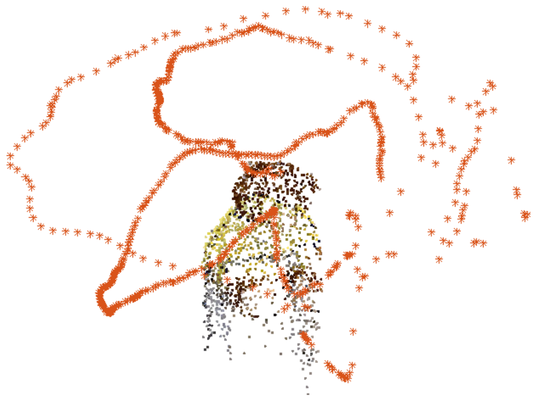
METRIC MAPS

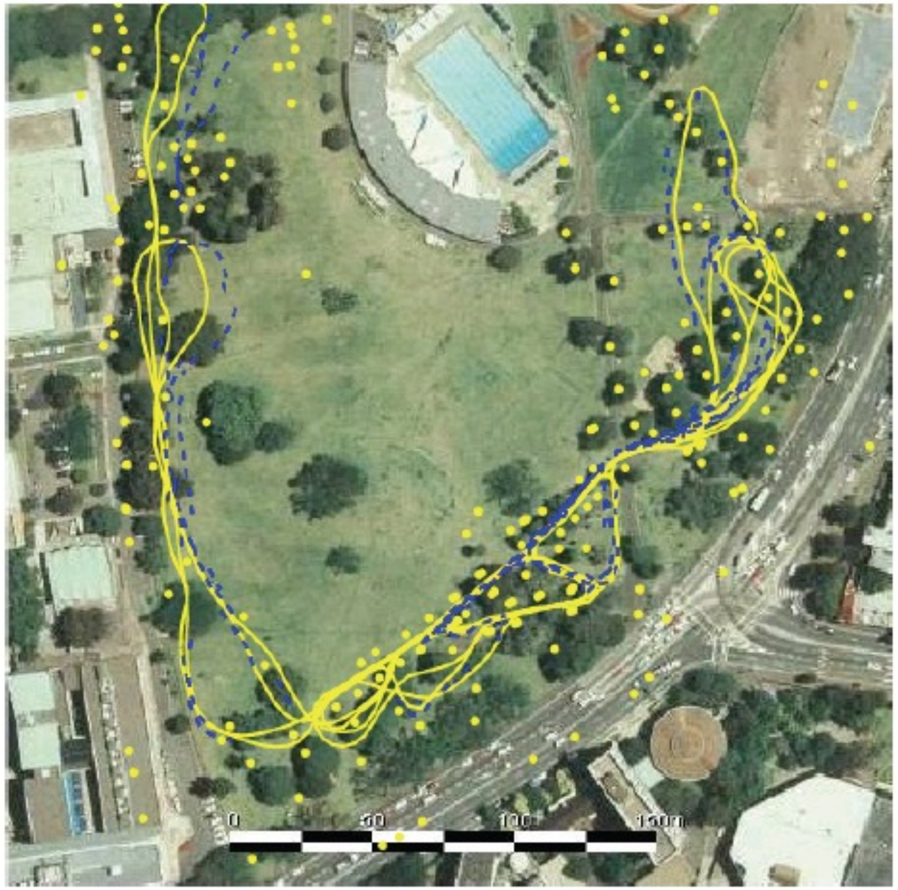
- Sparse (features) - excellent for localisation
- Occupancy – good for navigation
- Signed Distance functions – excellent for visualisation once the mesh is recovered and excellent for navigation
- Meshes – excellent for manipulation and visualisation

SPARSE REPRESENTATIONS

Feature representations



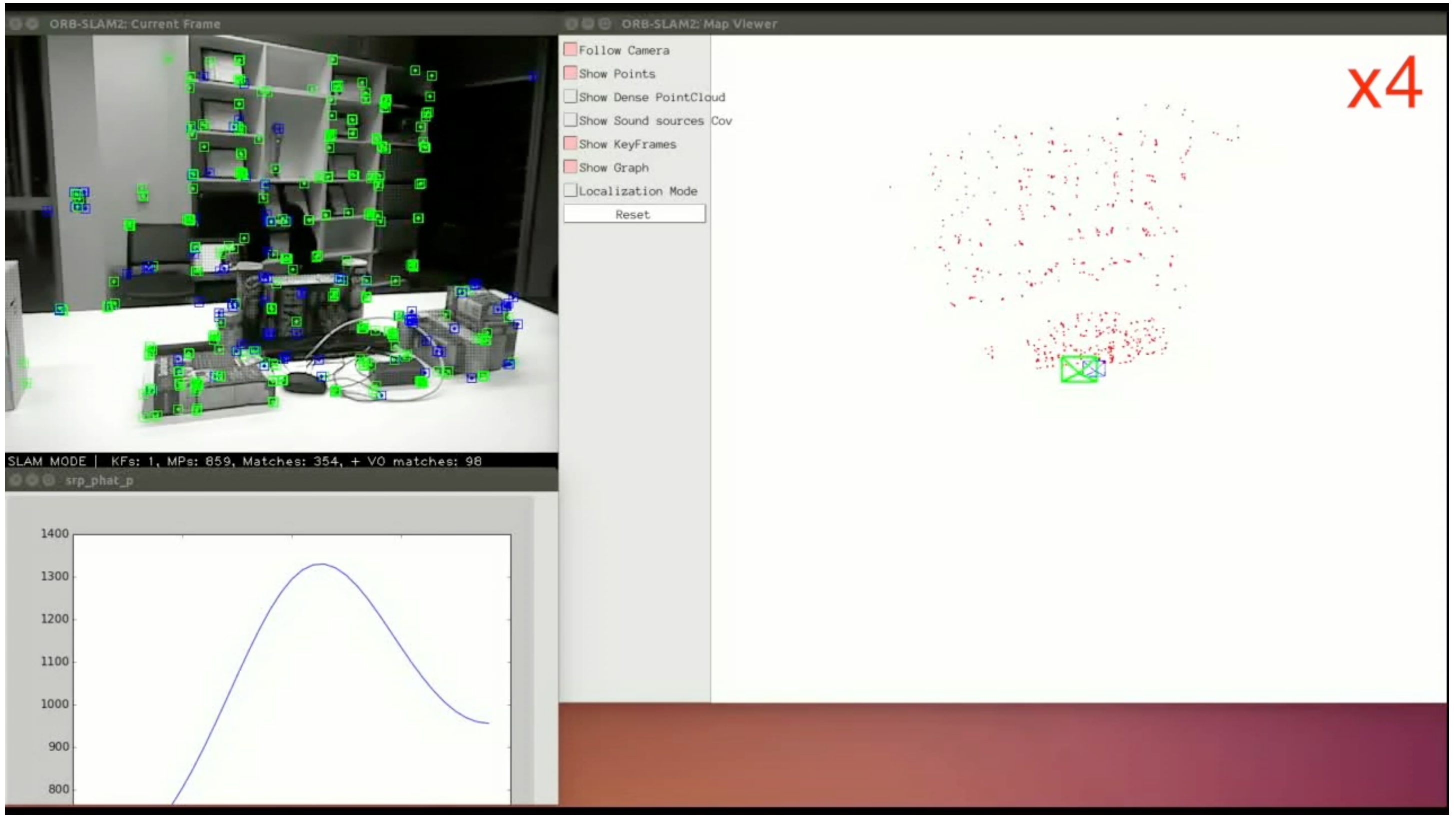




Courtesy by E. Nebot

FEATURE MAPS

- Landmark Based
- Kalman filter or Optimisation based systems
- Compact/parametric representation
- Multiple feature observations to improve position estimates

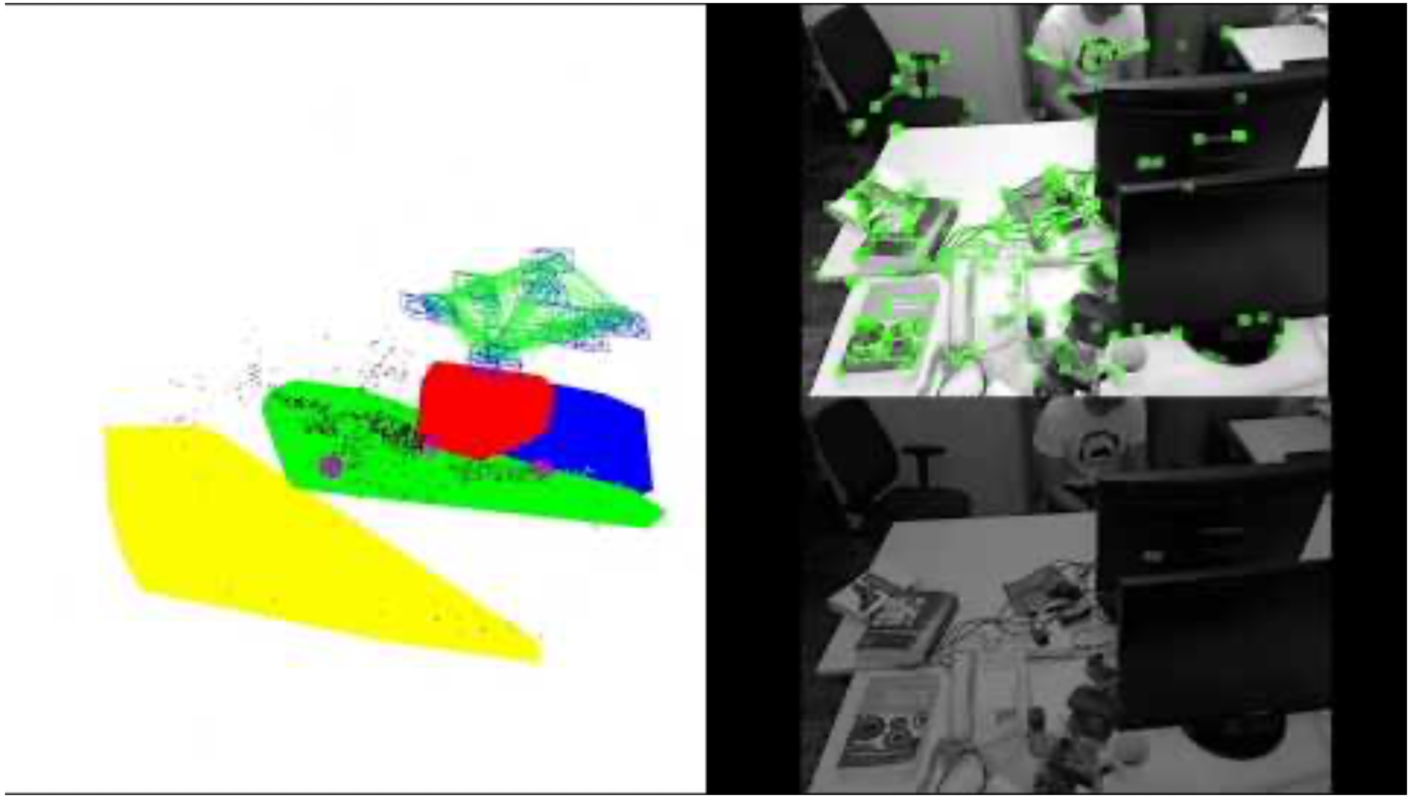


Based on ORBSLAM2

Mur and Tardos, 2018

SPARSE REPRESENTATION – PLANES/OBJECTS

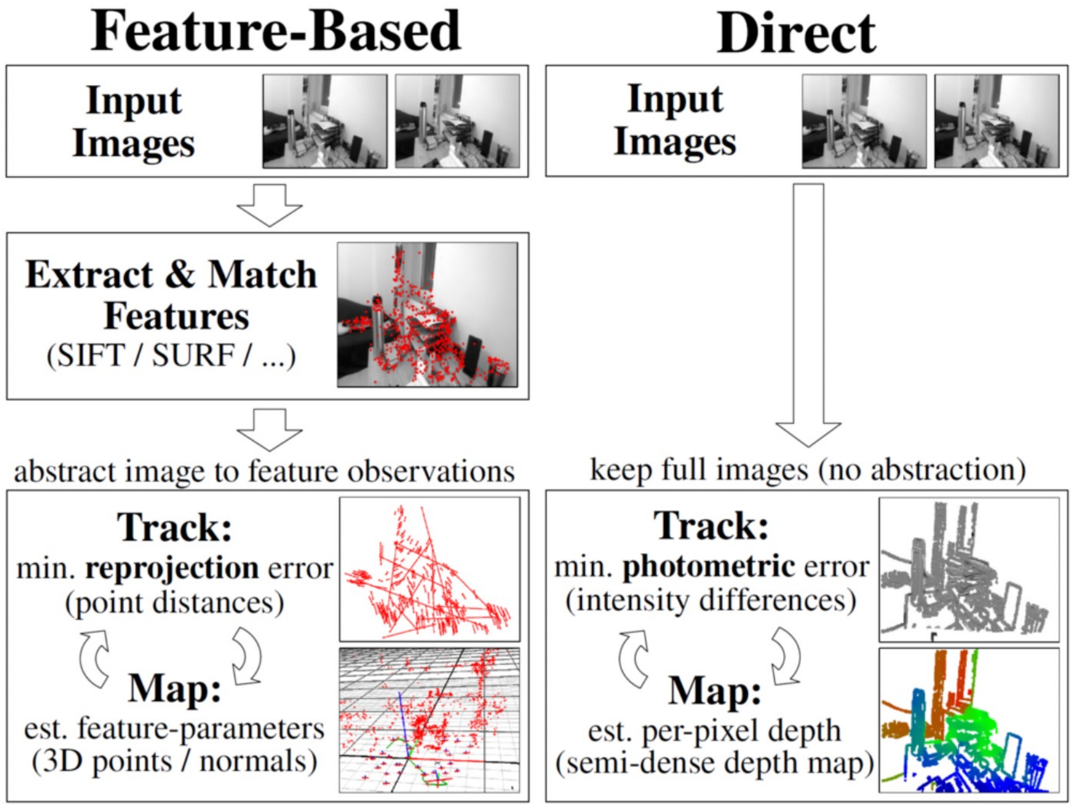
- Parametric representations
- Filter and Optimisation

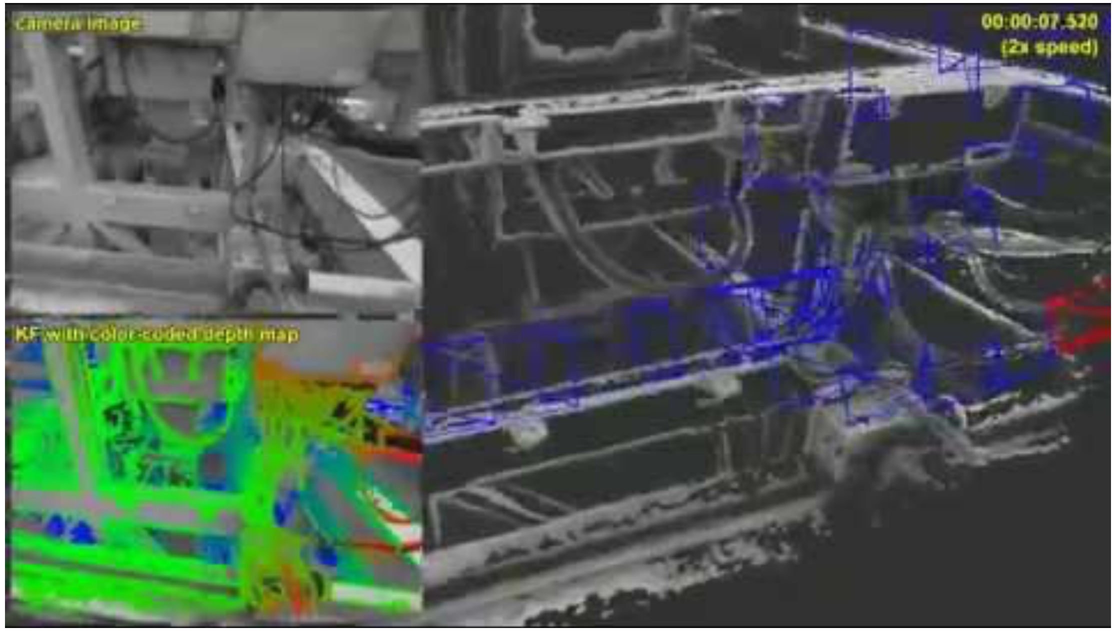


https://www.youtube.com/watch?v=dR-rB9keF8M

Hosseinzadeh et al 2018

SEMI-DENSE REPRESENTATION





https://youtu.be/GnuQzP3gty4

Engel et al 2017

DENSE REPRESENTATIONS

Estimated Quantities

Assuming point clouds as inputs

- Occupancy
- Surface
- Distance fields /Implicit Surface

SPACE ABSTRACTIONS

- Points
- Surfels
- NDT
- Mesh
- Voxels
- Continuous functions

Explicit Surface

SPACE ABSTRACTIONS

- Points
- Surfels
- NDT
- Mesh
- Voxels
- Continuous functions

Explicit Surface

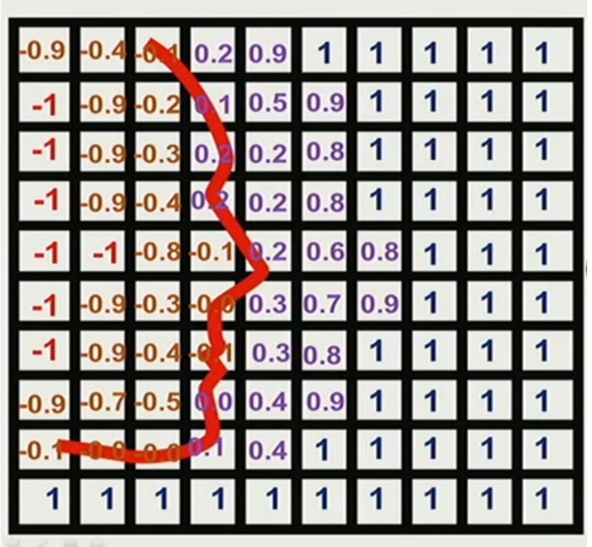
Occupancy

SPACE ABSTRACTIONS

- Points
- Surfels
- NDT
- Mesh
- Voxels
- Continuous functions

Explicit Surface

Implicit Surface and Distance Fields

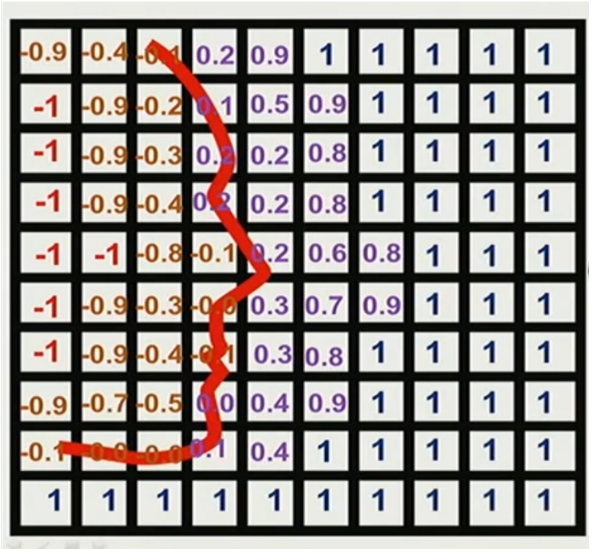


SPACE ABSTRACTIONS

- Points
- Surfels
- NDT
- Mesh
- Voxels
- Continuous functions

Explicit Surface

Implicit Surface and
Distance Fields



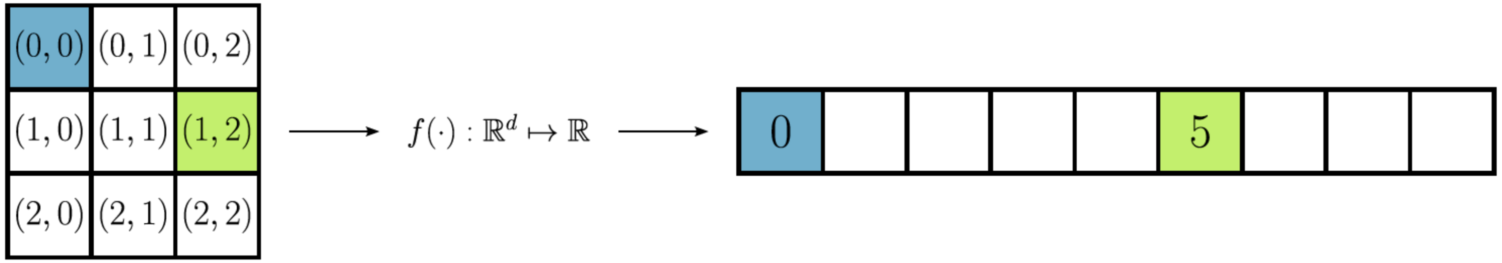
Marching cubes

Conversions

DATA STRUCTURES

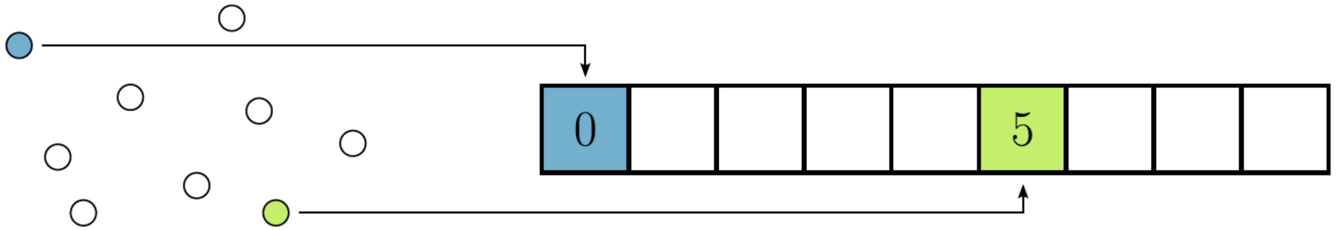
- Naïve
- Hash Maps
- Trees
  - Octrees
  - VDB tree structure
- Hybrid methods

NAÏVE STRUCTURE



(a)

Organised Point Cloud



(b)

Unorganised Point Cloud

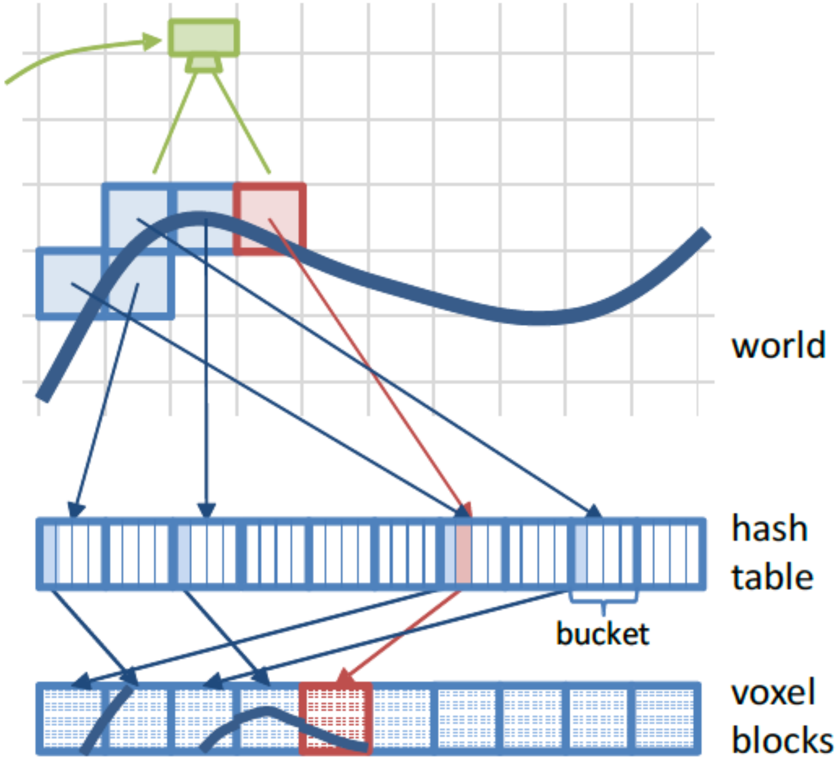
Reijgwart et al, 2024

HASH MAPS

Conceptually, an infinite uniform grid partitions the world map from integer world coordinates to hash buckets, which store a small array of pointers to regular grid voxel blocks.

Each voxel block contains a grid of values

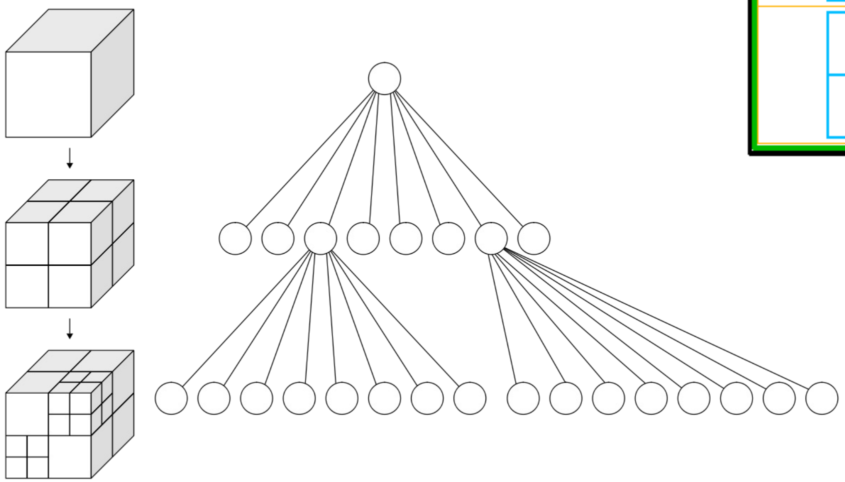
When information for the red block gets added, a collision appears which is resolved by using the second element in the hash bucket

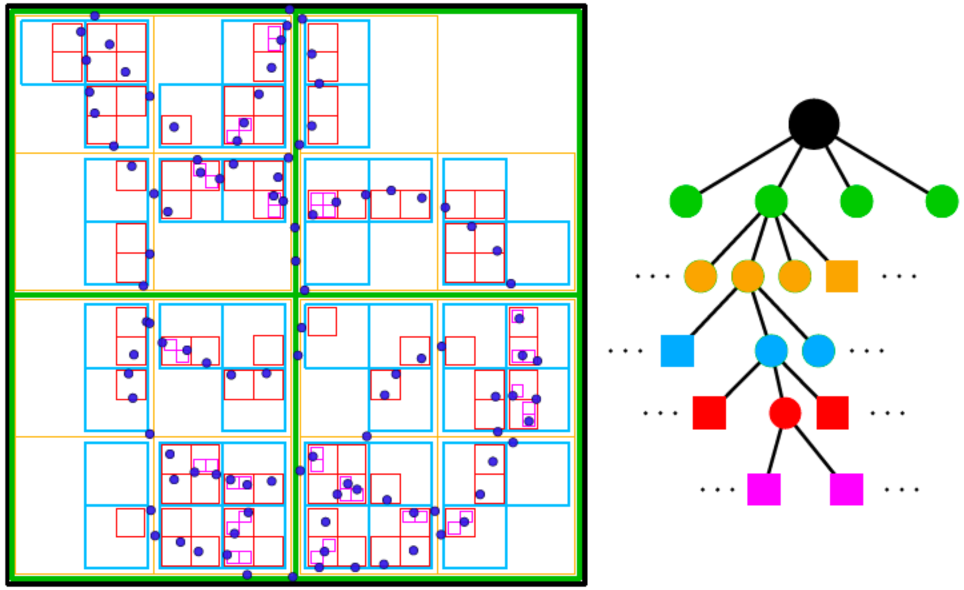


Niessner et al, 2013

OCTREES

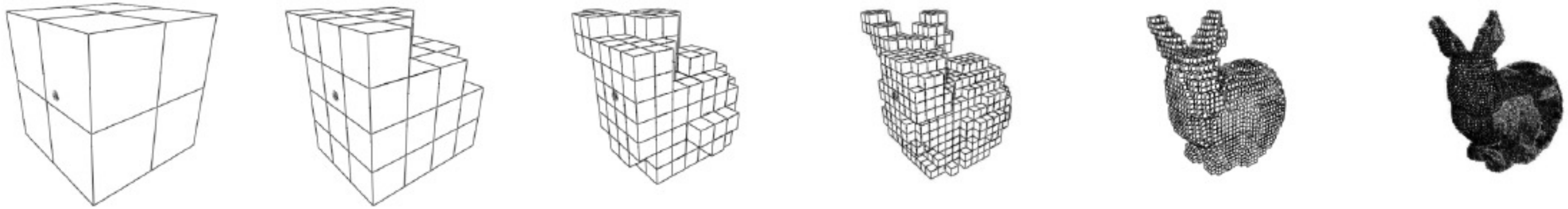
- Internal node has exactly 8 children
- Supports multi-resolution data
- Limitation in their memory overhead and cell lookup time being proportional to their height





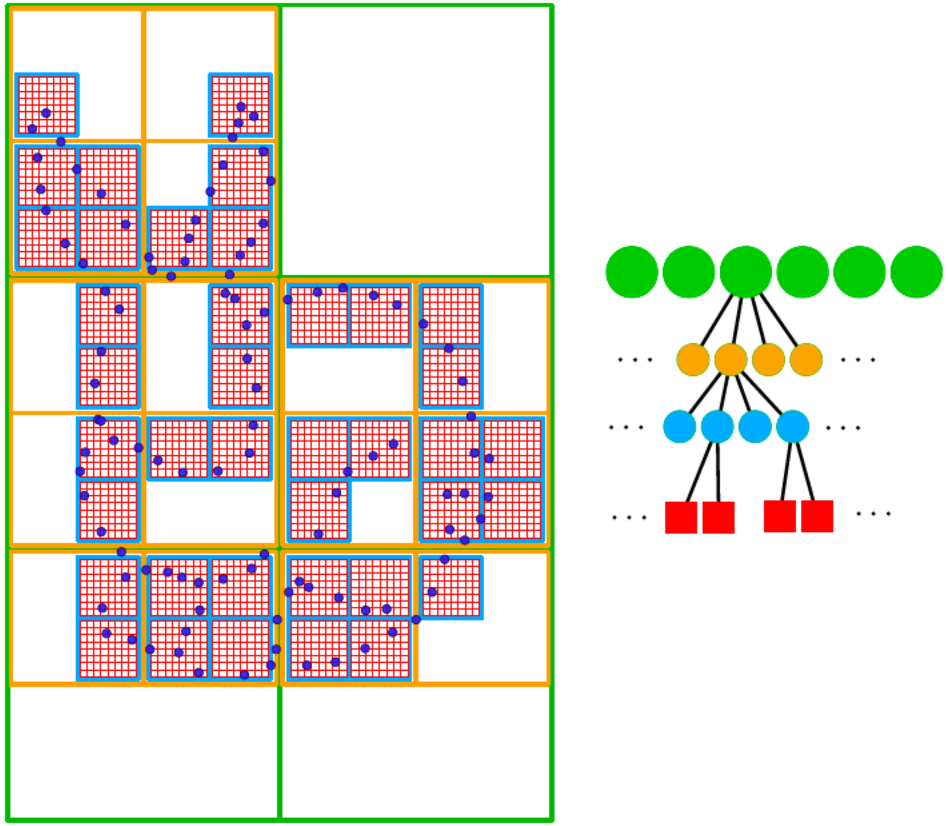
Vizzio et al, 2023

Octree



VDB TREE STRUCTURE

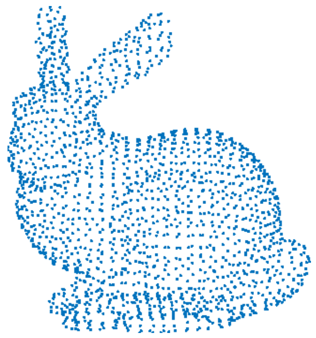
- A hybrid volumetric data structure
- It uses a hash-table at the highest level, similar to voxel-block hashing, but stores a tree in each hash-block instead of a 3D array
- By extension of the hash blocks having a fixed width, the maximum tree height is constant as well
- Memory efficient



VDB tree structure

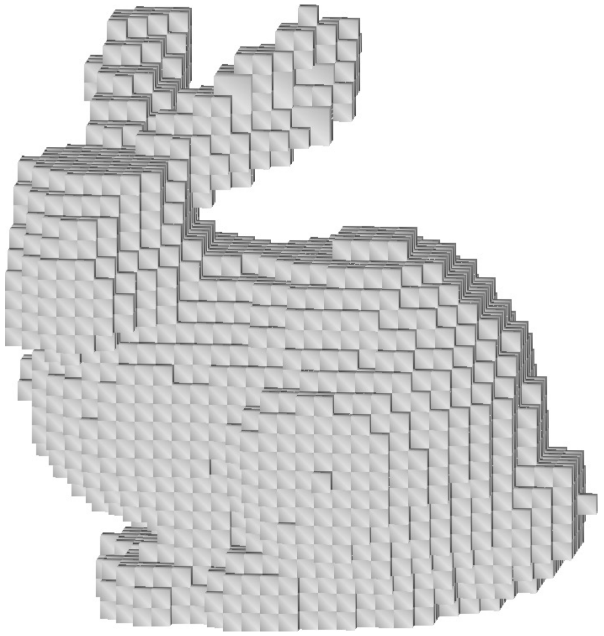
Vizzio et al, 2023

DENSE REPRESENTATIONS



Estimated Quantities

- Occupancy
- Surface
- Distance fields /Implicit Surface



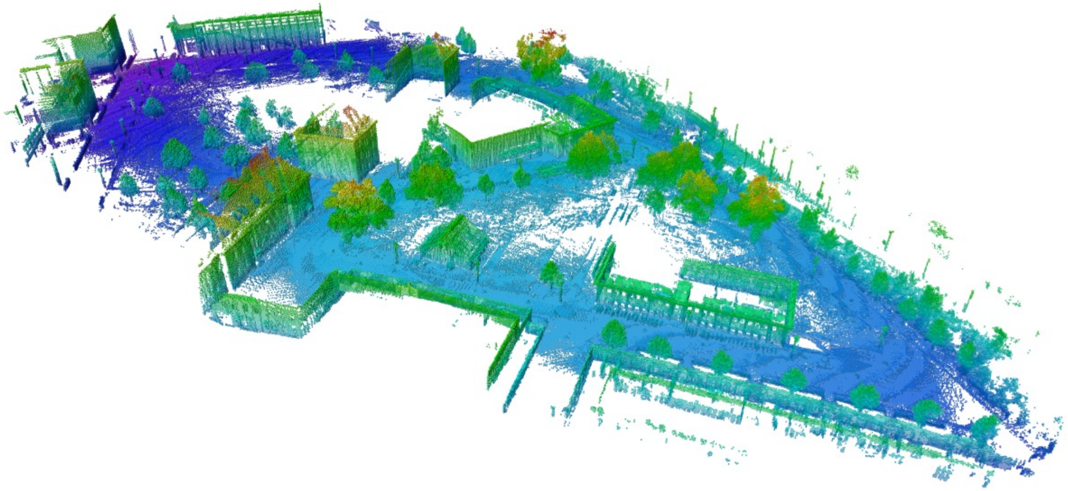
OCCUPANCY MAPS

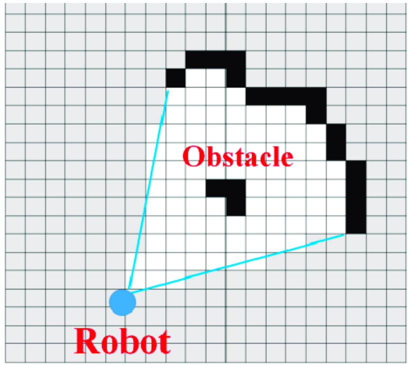
- Discretise the world into grid cells (2D) or voxels (3D)
- Each cell is assumed to be occupied or free space
- Non-parametric model
- Arrays, quadtrees, octrees structure
- Does not rely on a feature detector

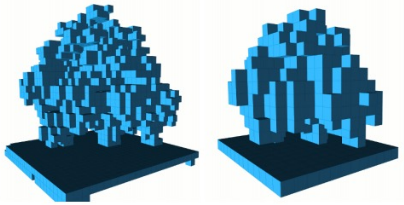
$$o(m_k|\boldsymbol{z}_{1:t}) = \frac{p(m_k|\boldsymbol{z}_{1:t})}{1 - p(m_k|\boldsymbol{z}_{1:t})}$$

Log-odds $$l(m_k|\boldsymbol{z}_{1:t}) = l(m_k|\boldsymbol{z}_{1:t} - 1) + l(m_k|z_t)$$

$$l(\cdot|\cdot) = \log(o(\cdot|\cdot))$$







Elfes and Moravec, 1985
Hornung et al, 2013

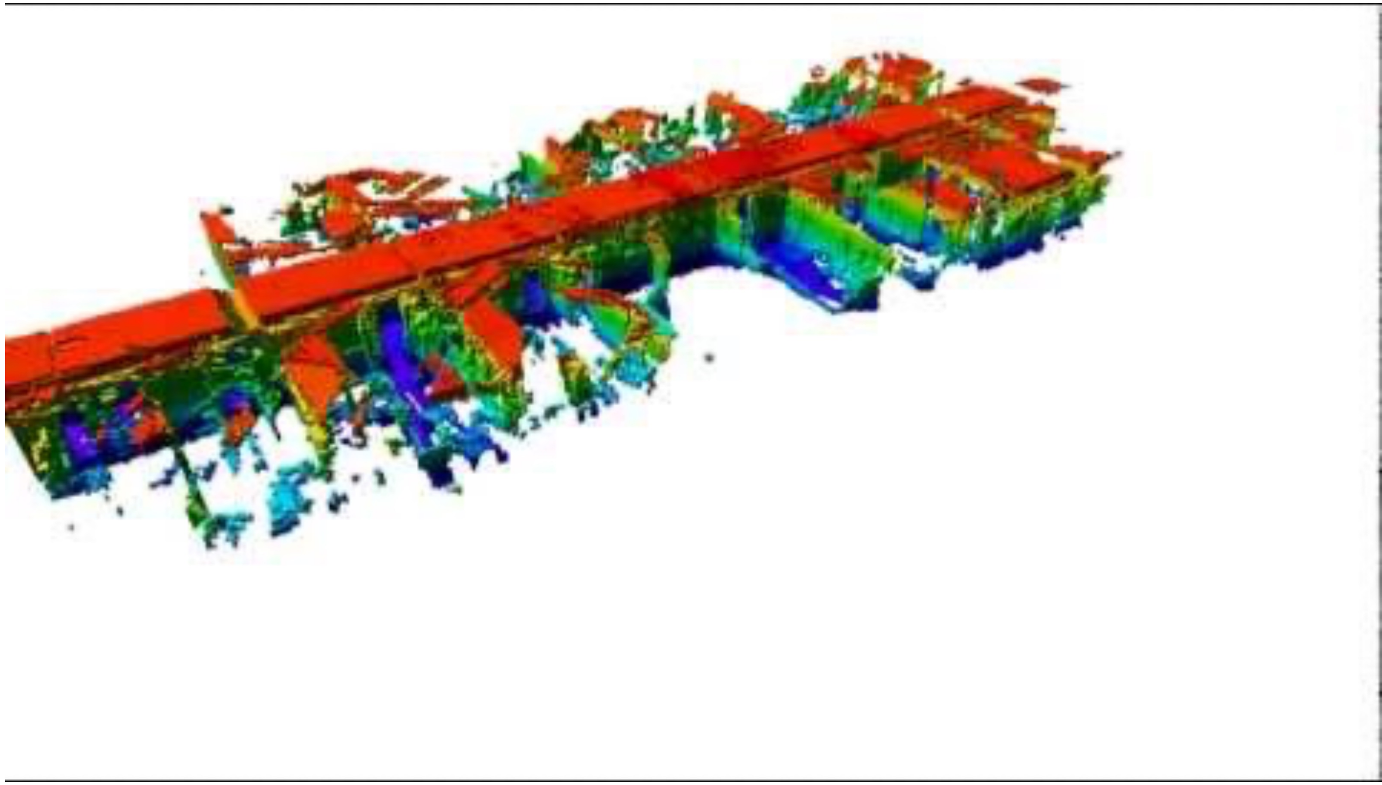
OCCUPANCY MAPS

Octomap

- Based on Octrees and Voxels
- store occupancy probabilities in inner nodes
- Free space is represented with fewer cells

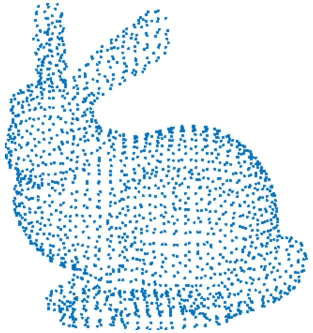
Supereight

- Adaptive resolution
- More efficient as lower levels of the tree are fixed



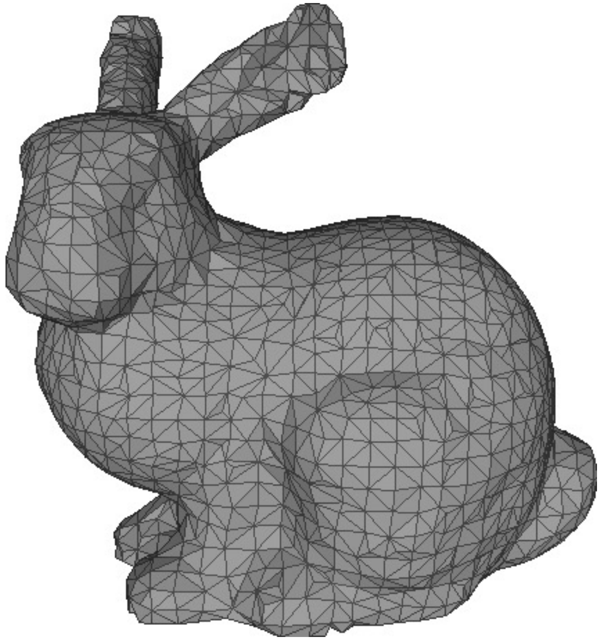
Hornung et al, 2013
Vespa et al, 2022

DENSE REPRESENTATIONS



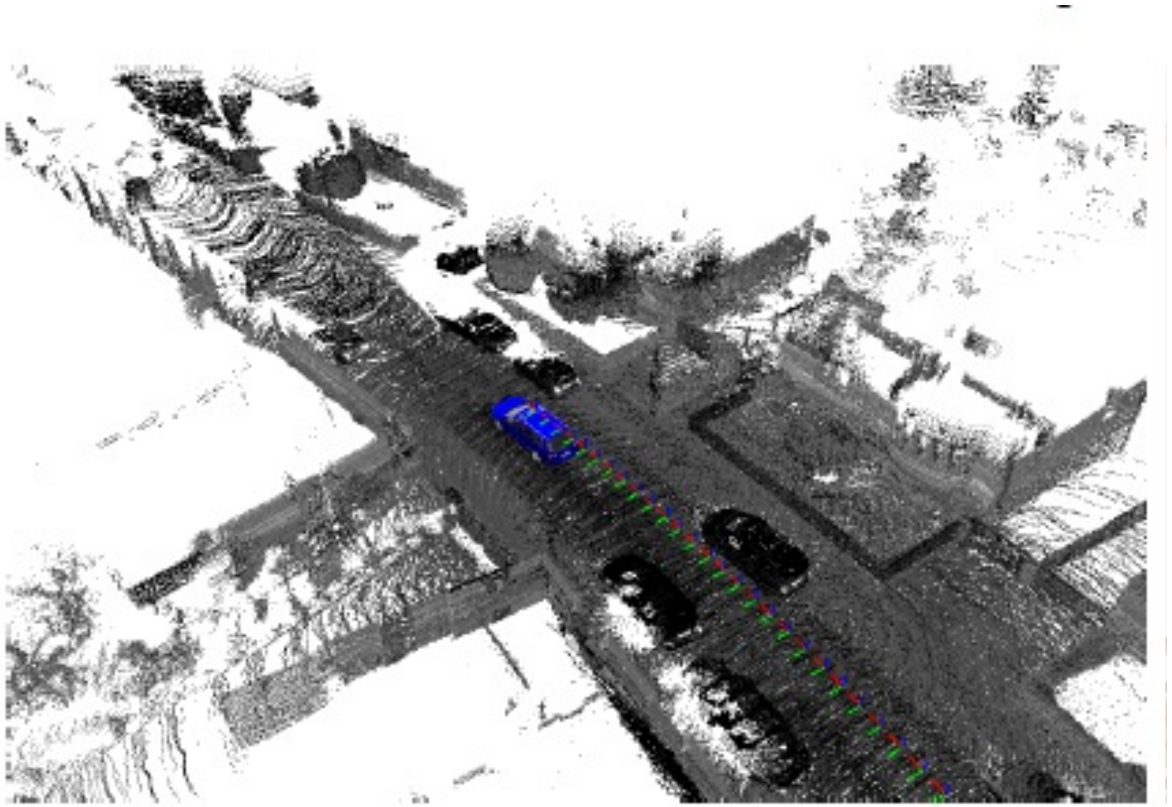
Estimated Quantities

- Occupancy
- Surface
- Distance fields /Implicit Surface



POINT CLOUDS

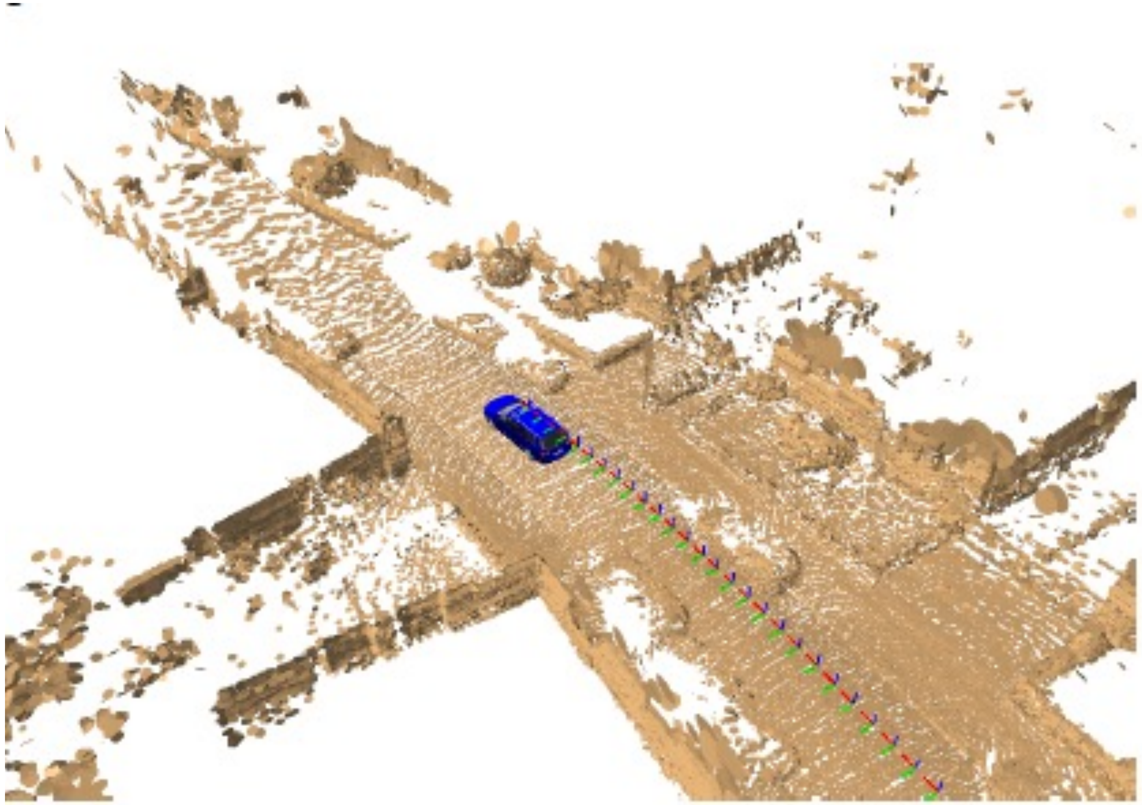
- Doesn’t scale to simply accumulate points
- Commonly store in voxel grids
- Only point clouds from keyframes are stored
- Although are samples from the surface do not inherently contain surface information
- Normals needs to be computed from noisy points



Reijgwart et al, 2024

SURFELS

- Circular or elliptic discs or ellipsoids "surface patches"
- Encode the normal of the surface in the direction of the sensor
- Model is point location, normal and radius
- Closely related to NDT
- Gaussian Splatting allows to integrate even texture



Reijgwart et al, 2024

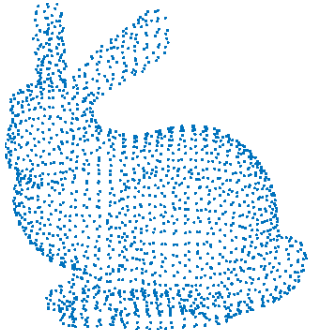
MESHES

- Polygon mesh is a collection of vertices, edges and faces that defines surfaces
- Faces are commonly triangular mesh
- Can be generated by Delaunay Triangulation or
- Marching Cubes from implicit surfaces (more commonly)

https://youtu.be/LvmBjMvmZKA

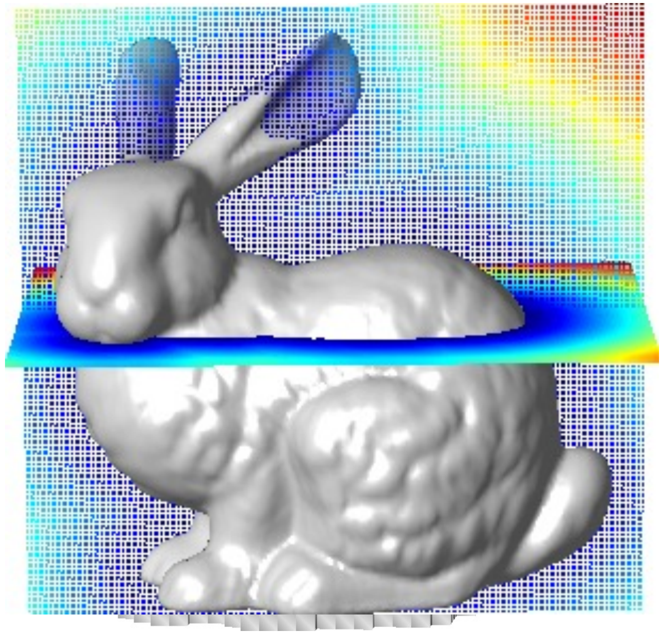
Texeria and Chli, 2016

DENSE REPRESENTATIONS



Estimated Quantities

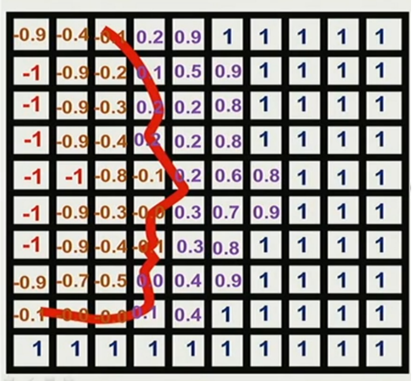
- Occupancy
- Surface
- Distance fields /Implicit Surface



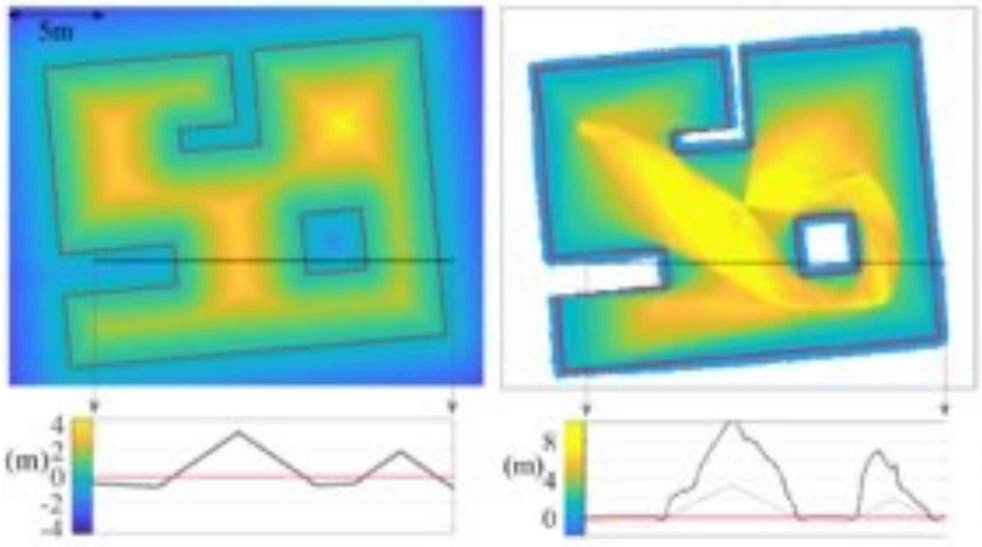
DISTANCE FIELD MAPS / IMPLICIT SURFACE

- The surface is implicitly represented
- The distance to the object can be calculated
- Store in grids (2D) or voxels (3D)
- Non-parametric model
- Depends on the algorithm it has been used with 3D arrays, hash maps, Octrees and VDB

$$f(\mathbf{x}) \begin{cases} > 0, & \text{if } \mathbf{x} \text{ is outside the surface} \\ = 0, & \text{if } \mathbf{x} \text{ is on the surface} \\ < 0, & \text{if } \mathbf{x} \text{ is inside the surface} \end{cases}$$



ESDF TSDF



TSDF

Kinectfusion

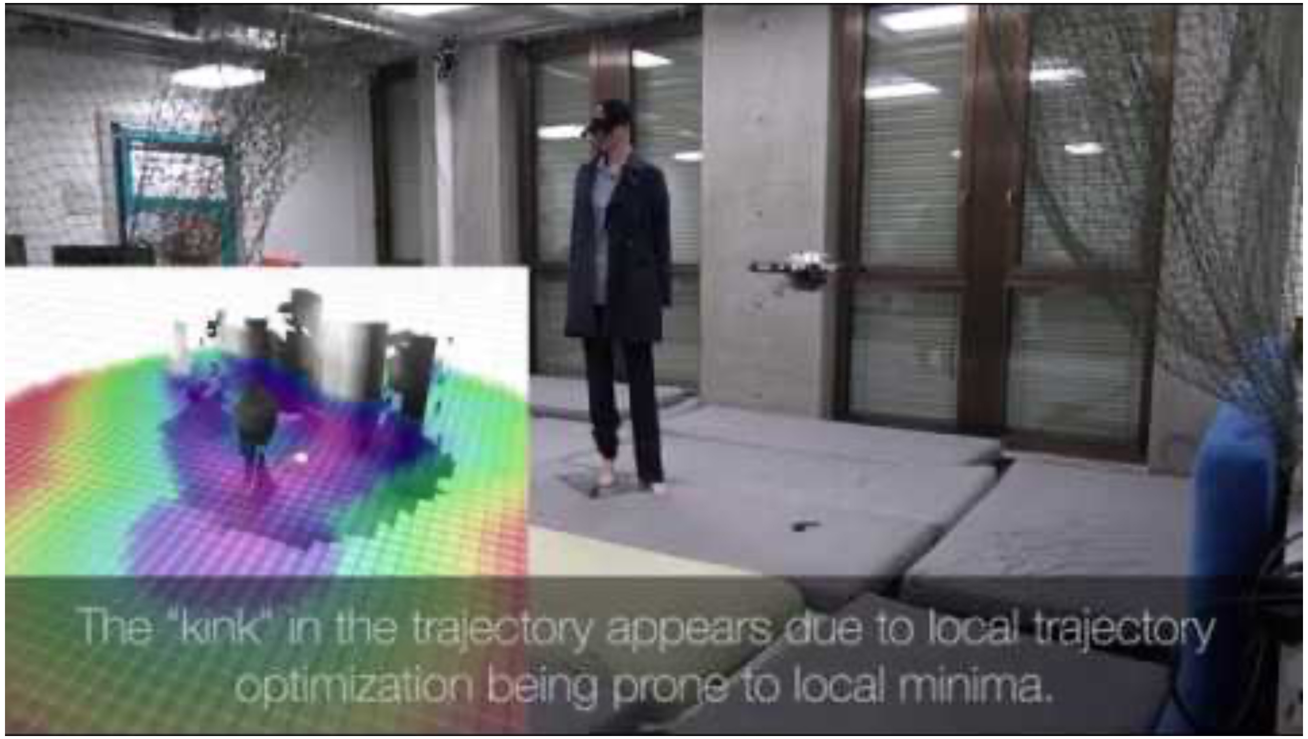
- Based on Truncated Signed Distance Functions
- TSDF fusion
- Voxel-based
- Data structure is a 3D array
- Small fixed size scenes
- projective distances

https://youtu.be/quGhaggn3cQ

ESDF

Voxblox

- Because TSDFs overestimate the Euclidean distance voxblox popularised incrementally building ESDFs through a wavefront propagation
- Voxblox fuses the sensor data using a TSDF



https://youtu.be/ZGvnGFnTVR8

Fiesta

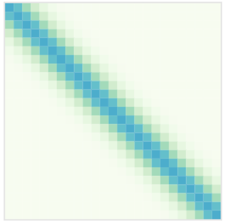
- Computes the ESDF from Occupancy

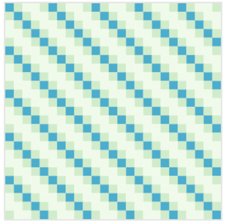
CONTINUOUS FUNCTIONS

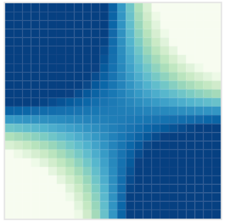
Gaussian Processes

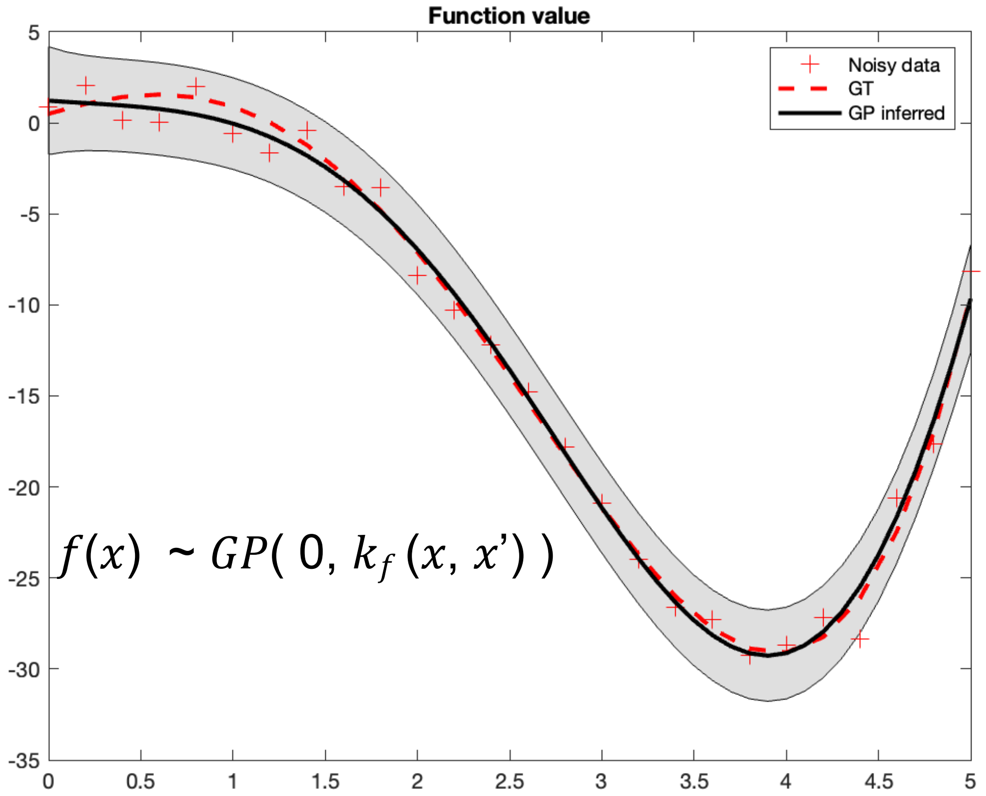
Generalisation of a Gaussian distribution over vector space to a function space

- Non-parametric
- Kernel method
- Reasons in Covariance
- Continuous functions can be linearly and nonlinearly operated









$$f(x) \sim GP(\, 0,\, k_f\,(x,\, x')\,)$$

CONTINUOUS OCCUPANCY MAPS

Gaussian Process Occupancy Maps

- sensor observations free or occupied as training data
- Regression of probability
- The cell's probability of occupancy is obtained by "squashing" regression outputs into occupancy probabilities using binary classification

- Hilbert maps
  - Approximates the Kernel with a Neural Network to make it more efficient

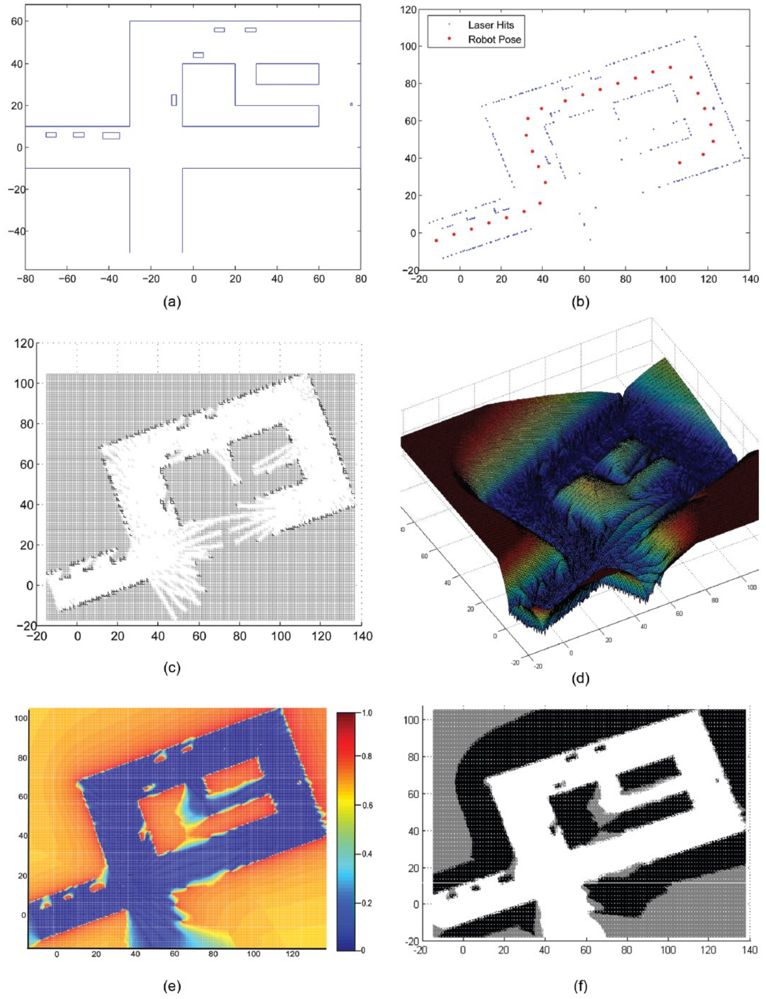


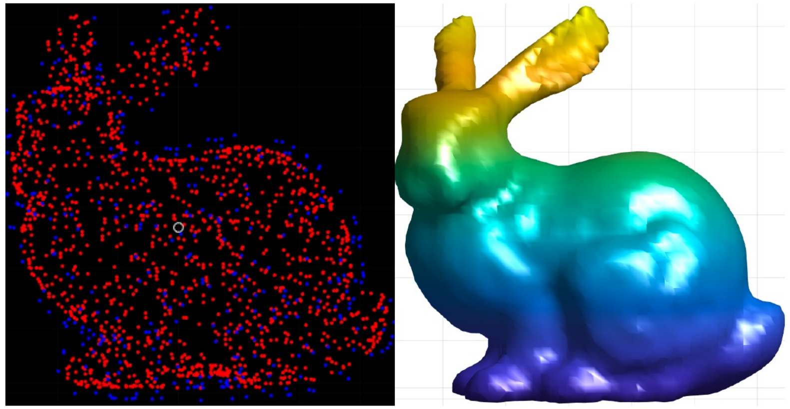
Fig. 6. Sequence of images illustrating the simulated results for Bayesian mapping using a Gaussian process (GP): (a) ground truth; (b) laser returns and robot pose; (c) occupancy grid; (d) predictive variance of GP; (e) probability of occupancy versus location using the GP approach; (f) classified GPOM. Black = Occupied ($p(O \mid \mathbf{x}) \geq 0.65$). White = Free space ($p(O \mid \mathbf{x}) \leq 0.35$). Grey = Unsure ($0.3 < p(O \mid \mathbf{x}) < 0.7$).

O'Callaghan and Ramos, 2012

CONTINUOUS IMPLICIT FIELDS

Implicit surface

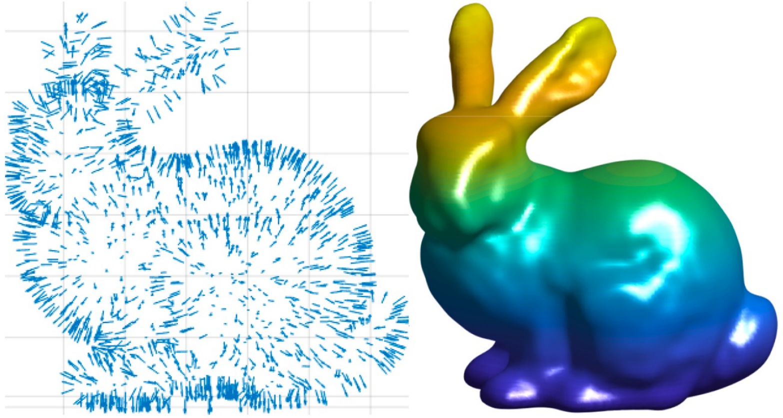
$$f(\mathbf{x}) \begin{cases} > 0, & \text{if } \mathbf{x} \text{ is outside the surface} \\ = 0, & \text{if } \mathbf{x} \text{ is on the surface} \\ < 0, & \text{if } \mathbf{x} \text{ is inside the surface} \end{cases}$$



Gaussian Process Implicit Surface (GPIS) with derivatives

$$\begin{bmatrix} f \\ \nabla f \end{bmatrix} \sim \mathcal{GP}(\mathbf{0}, \tilde{k}(\mathbf{x}, \mathbf{x}'))$$

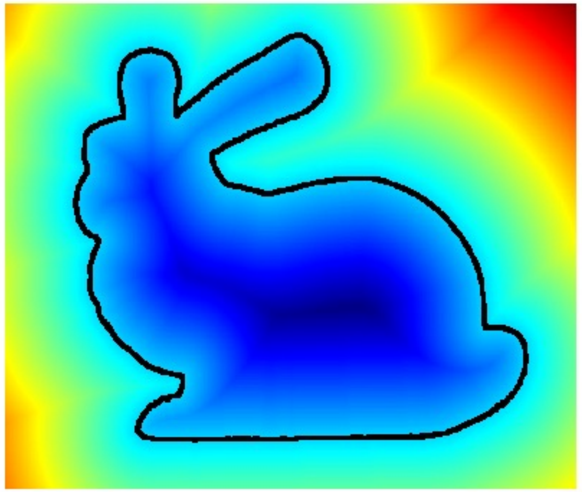
Models distance to the surface as a GP



CONTINUOUS EUCLIDEAN DISTANCE FIELD

Euclidean Distance Function

$$d(\mathbf{x}) = \min_{\mathbf{y} \in \partial S} d(\mathbf{x}, \mathbf{y})$$



Eikonal Equation $\quad |\nabla d| = 1 \quad \mathbf{x} \in \mathbb{R}^D$

$$d = 0 \quad \text{and} \quad \partial d / \partial \boldsymbol{n} = 1 \quad \text{on } \partial S$$

Nonlinear and hard to solve

The aim is to estimate $d(\mathbf{x})$ given sparse measurements $\mathbf{x}_i$, thereby reconstructing S

CONTINUOUS EUCLIDEAN DISTANCE FIELD – LOG-GPIS

Heat method to approximate a (linear) regularised Eikonal equation

Heat-based distance function

$$\frac{\partial v(\mathbf{x},t)}{\partial t} = \Delta v(\mathbf{x},t) \text{ with } \begin{cases} v(\mathbf{x},t) = 1 \text{ when } \mathbf{x} \in \partial S \\ v(\mathbf{x},0) = 0 \text{ when } \mathbf{x} \notin \partial S \end{cases}$$

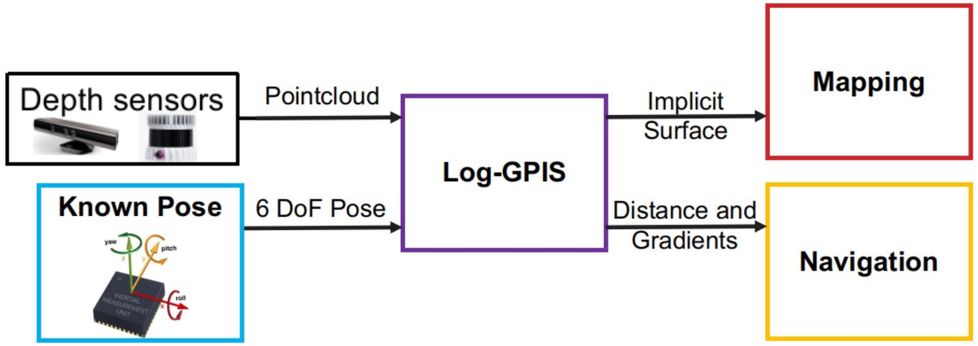
Log-GPIS -> models the heat equation $v(\mathbf{x})$ as a GPIS

- To get the app $d(\mathbf{x}) \approx -\sqrt{t} \ln v(\mathbf{x})$ we use

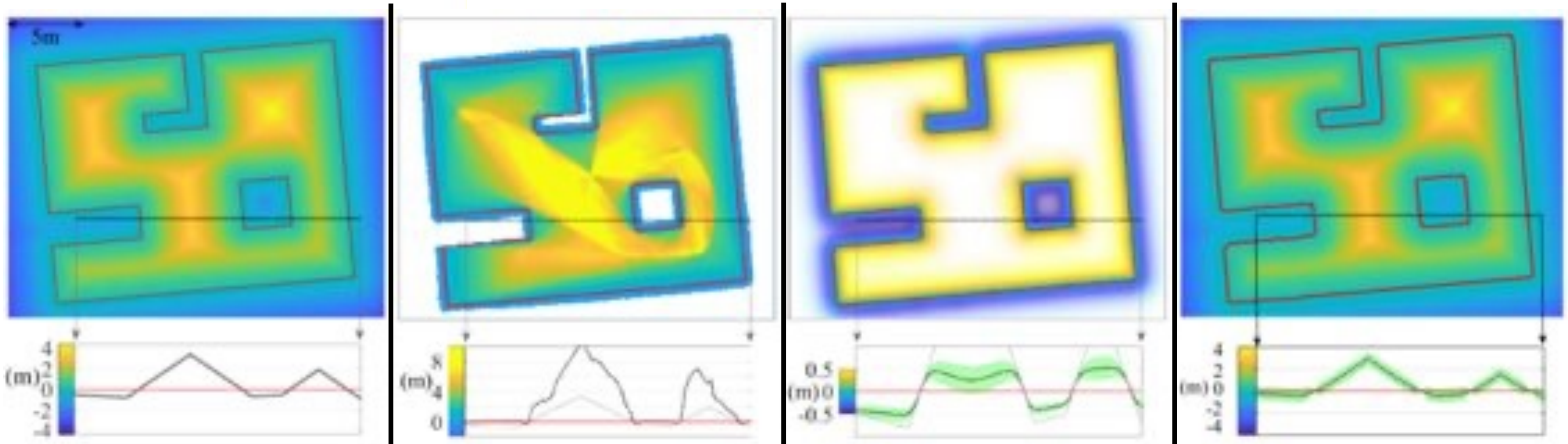
Log-GPIS

K. Crane et al, 2012.

LOG-GPIS



	Outputs of Log-GPIS	With Uncertainty
Surface	✓	✓
Surface Normals	✓	✓
Distance Field	✓	✓
Gradients of Distance Field	✓	✓

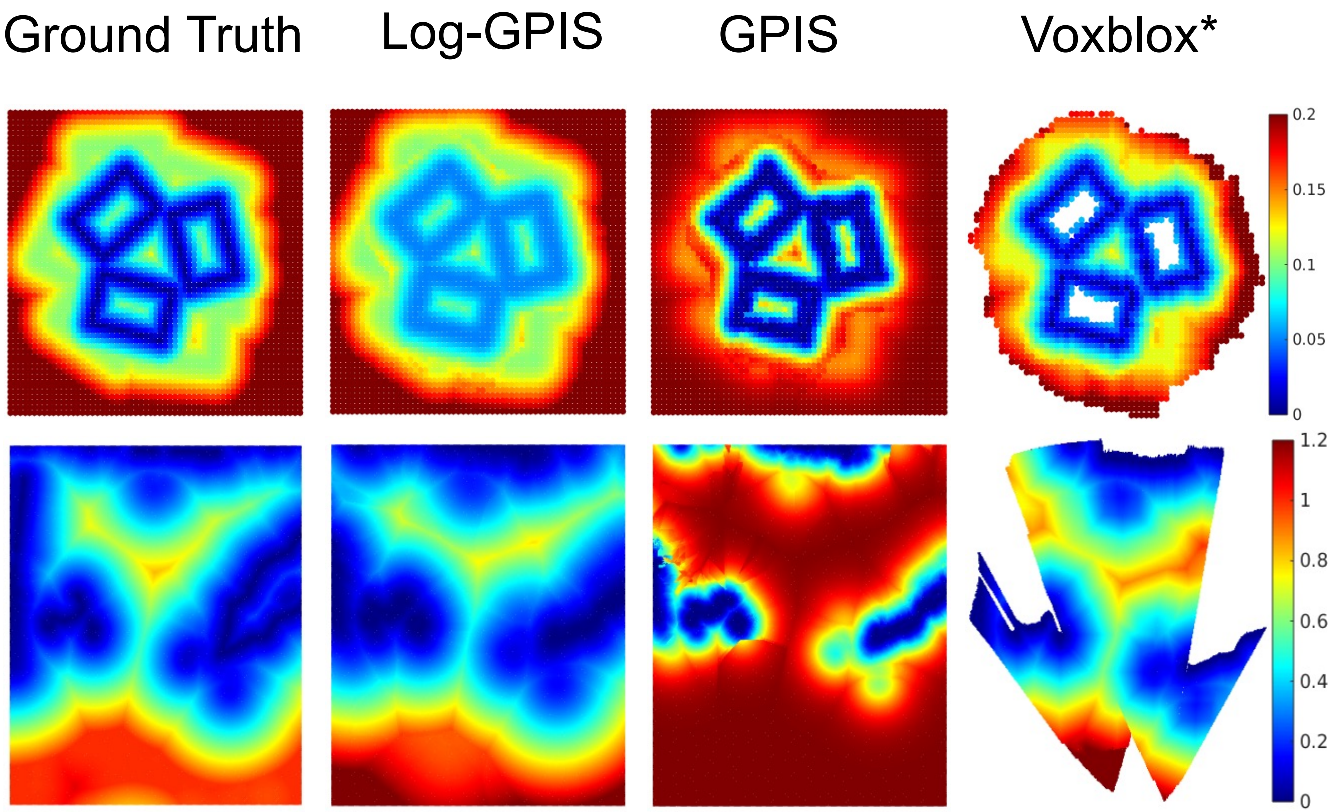


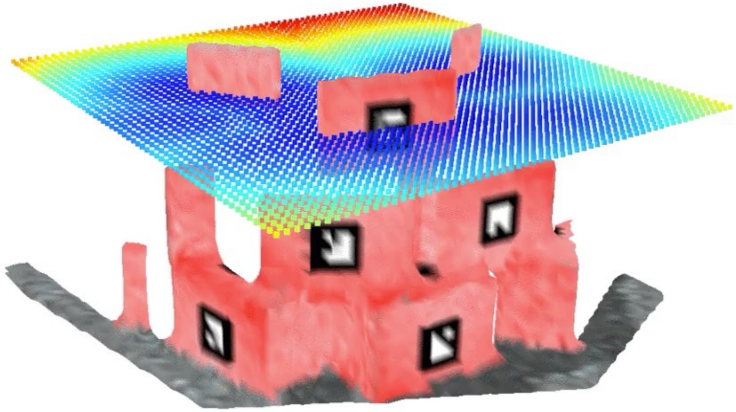
True ESDF TSDF GPIS* Log-GPIS

Lee et al 2019

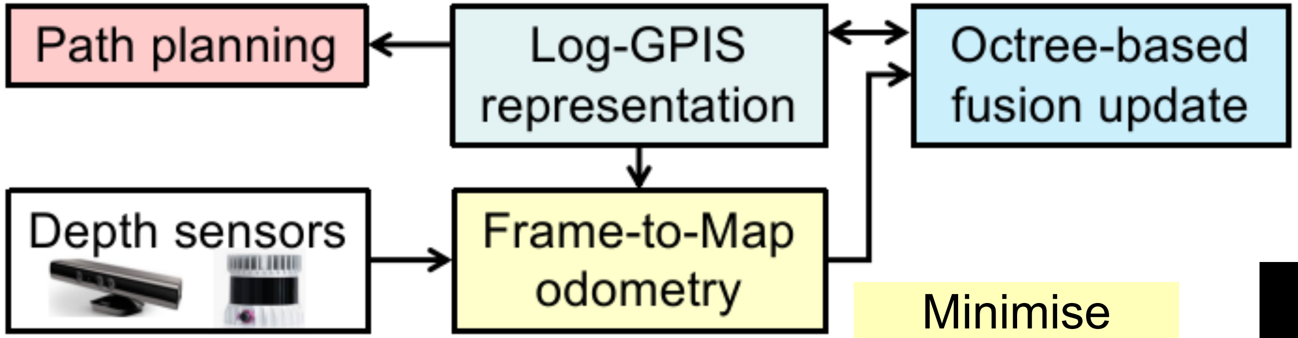
Wu et al RA-L 2021

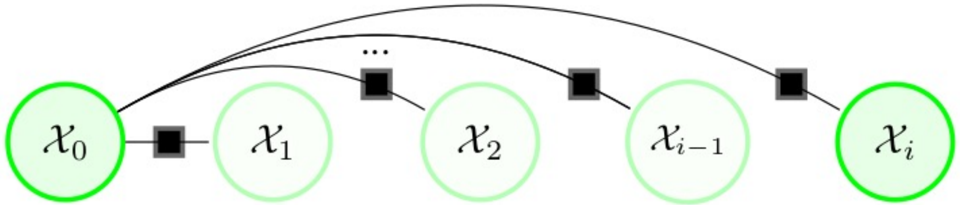
LOG-GPIS


Ground Truth
Log-GPIS
GPIS
Voxblox*
0.2
0.15
0.1
0.05
0
1.2
1
0.8
0.6
0.4
0.2
0

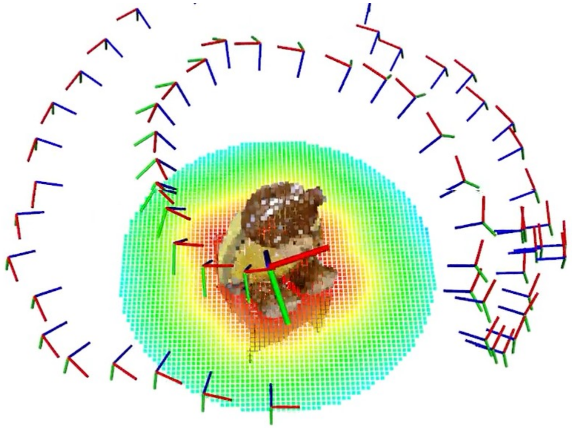


LOG-GPIS INCREMENTAL ODOMETRY AND MAPPING



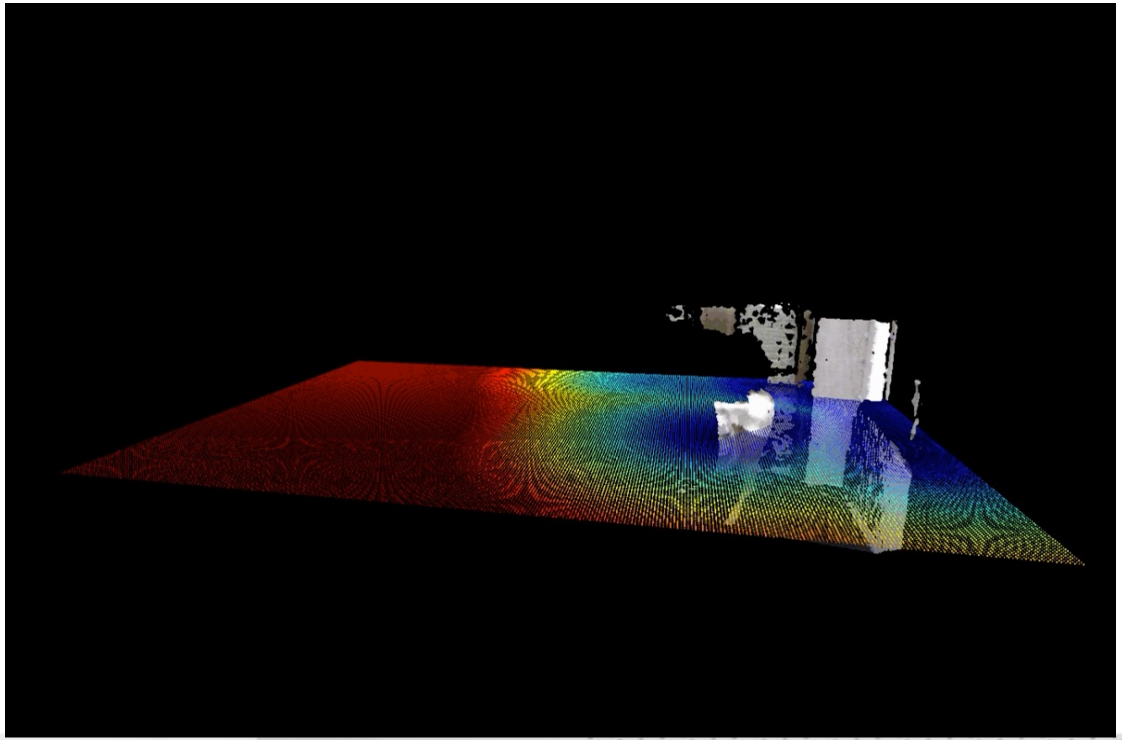


Minimise distance to the surface (GP query)

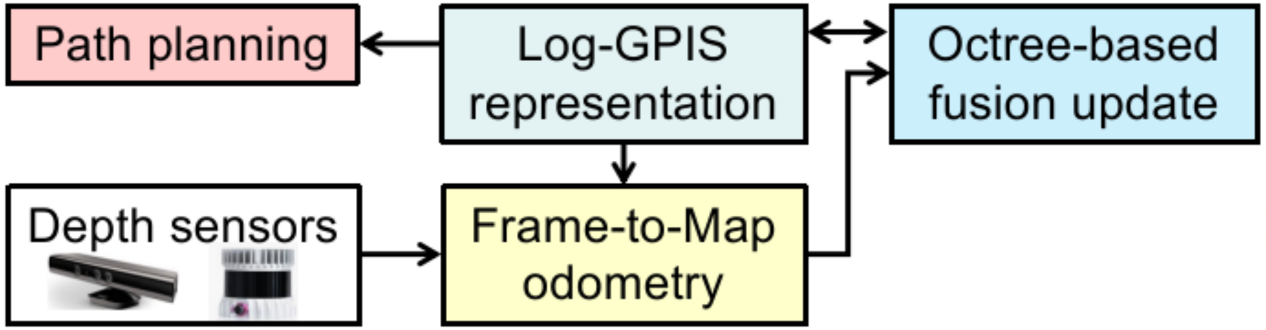


Reverting kernel

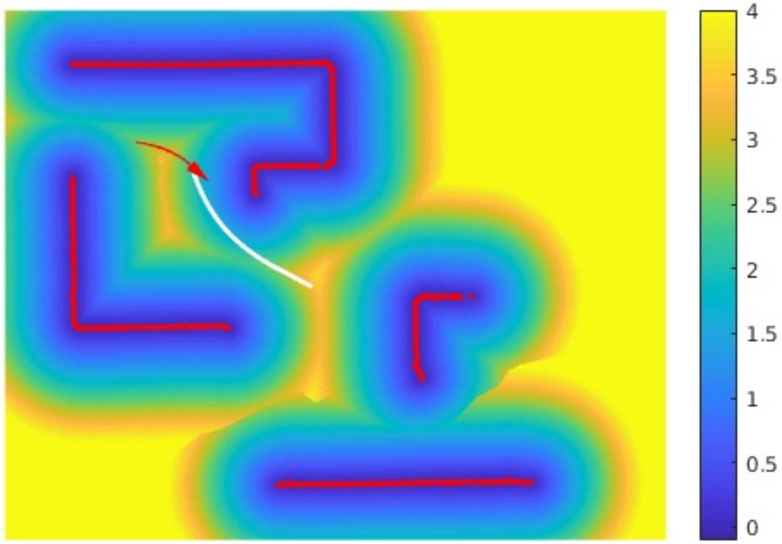
1X

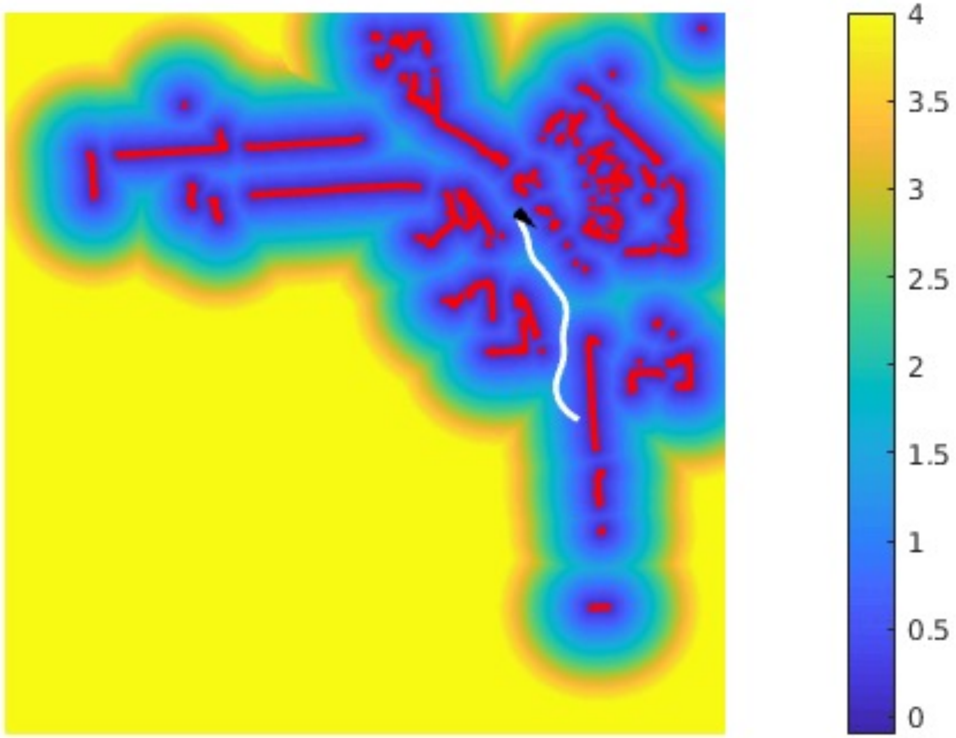


LOG-GPIS LOCALISATION, MAPPING AND PLANNING



Covariant Hamiltonian optimisation-based motion planner (CHOMP) for trajectory planning

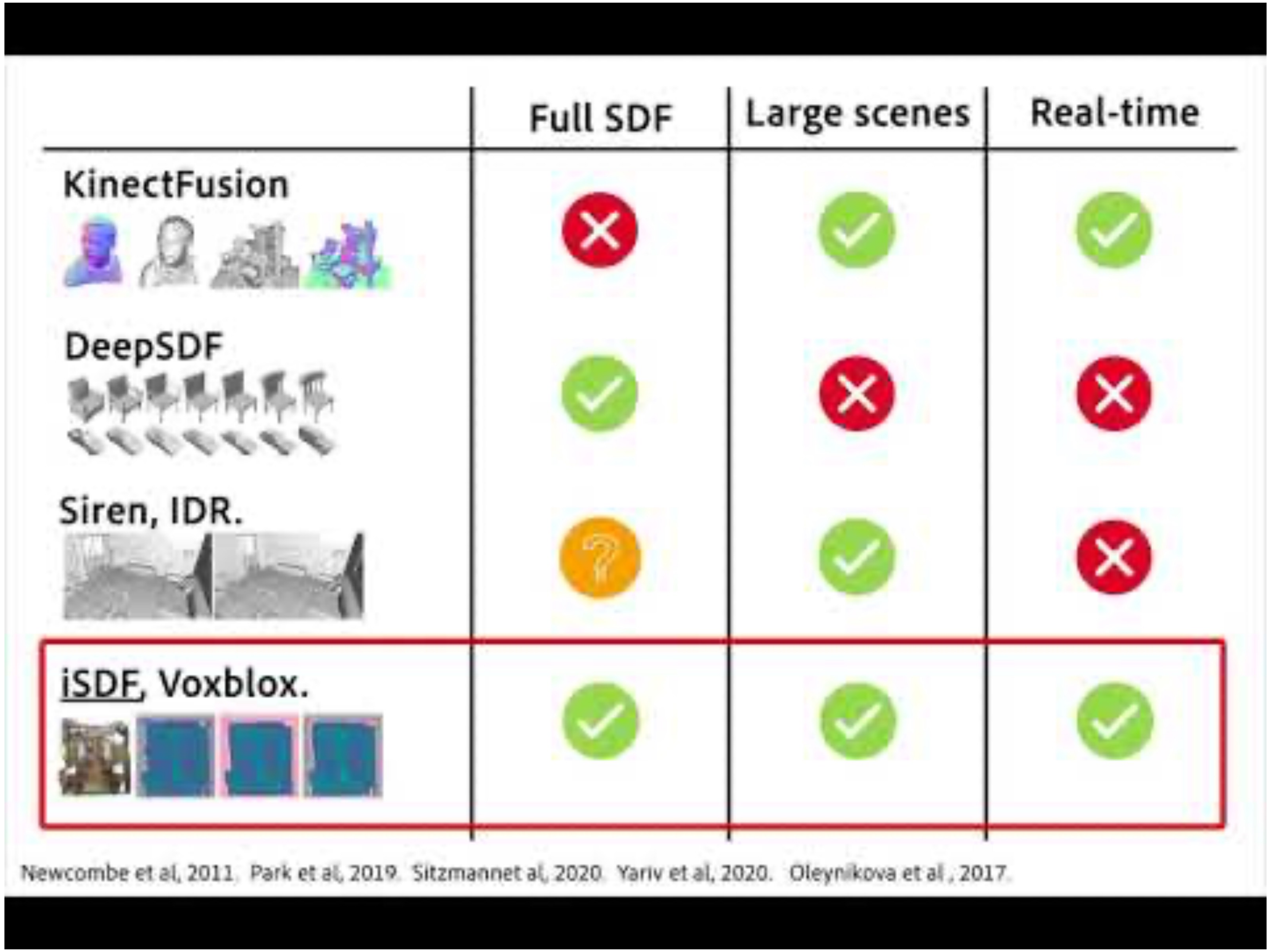




Wu et al IEEE TRO, 2023

DEEP LEARNING METHODS

- NERF-based



https://youtu.be/mAKGI1wBSic

Ortiz et al, 2022

DEEP LEARNING METHODS

- Gaussian Splatting –based

Gaussian Splatting SLAM

Spla-TAM

https://youtu.be/x604ghp9R_Q

Matsuki et al, 2023

Keetha et al, 2023