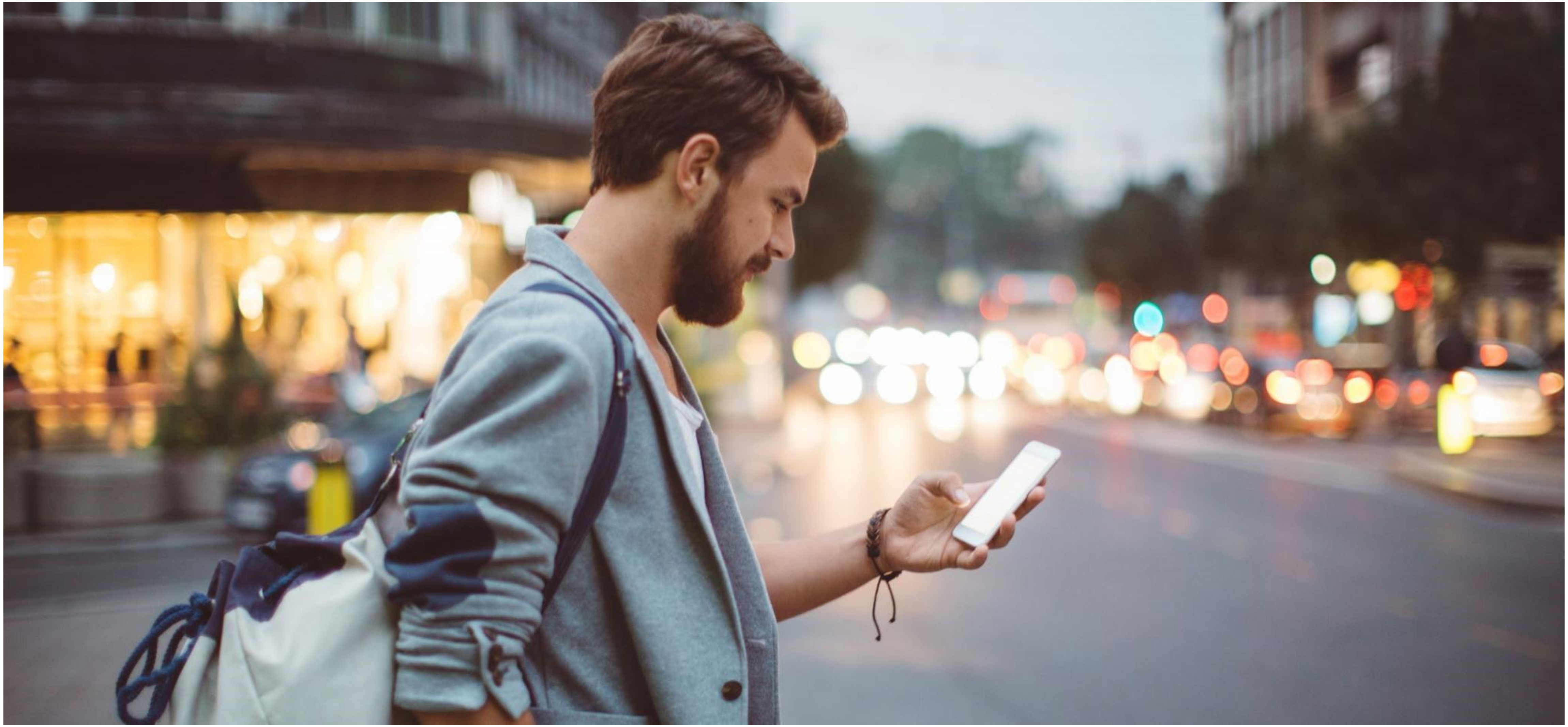
Augmented Reality for Indoor Positioning

A METHOD USING THE MICROSOFT HOLOLENS

TU Delft CGI

Laurens Oostwegel | July 2020



Indoor positioning

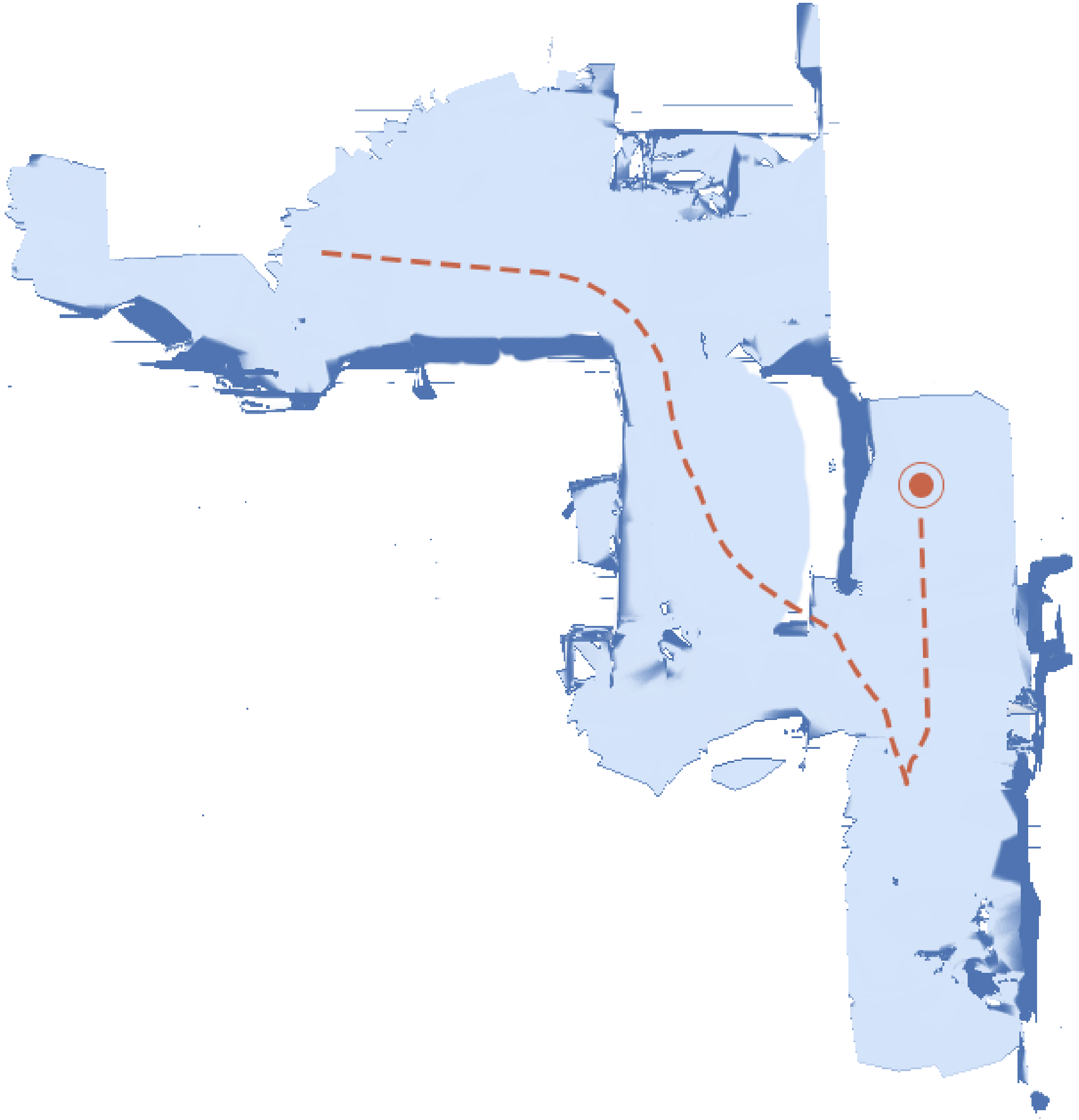
No large-scale solution

Infrastructure is needed

Bluetooth tags

Wi-Fi fingerprinting

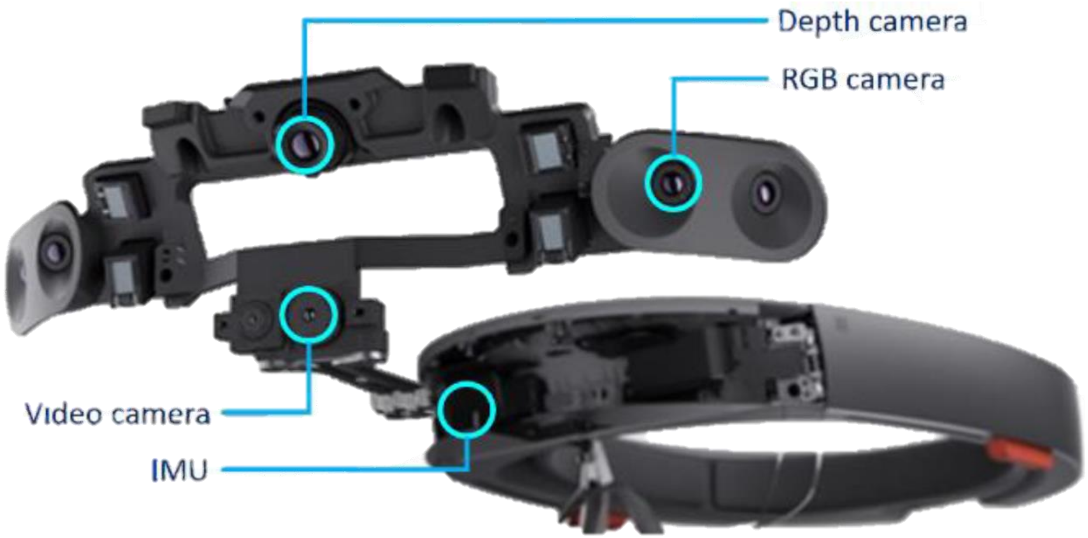
Augmente d reality



SLAM

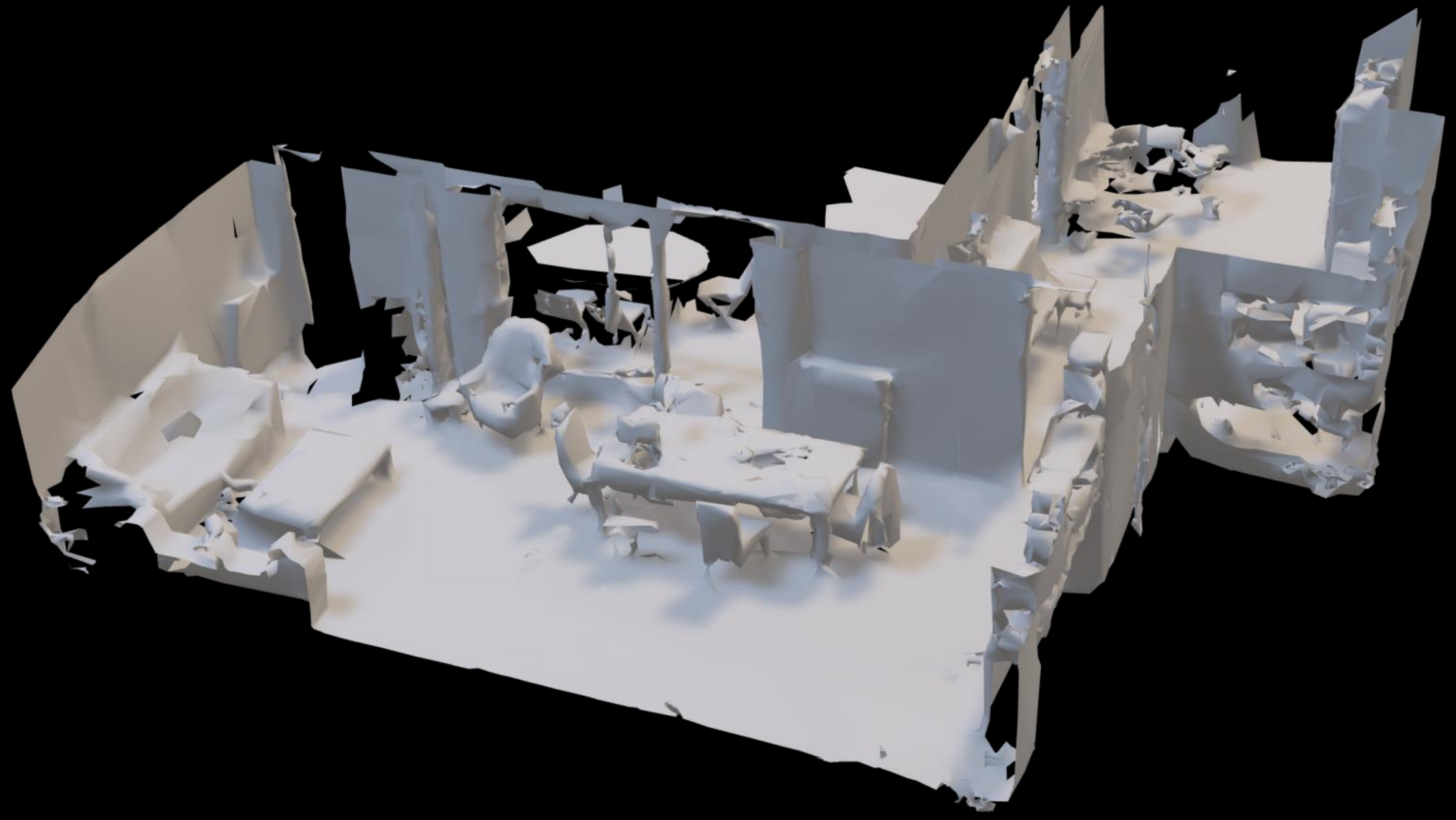
- Mapping

- Localization

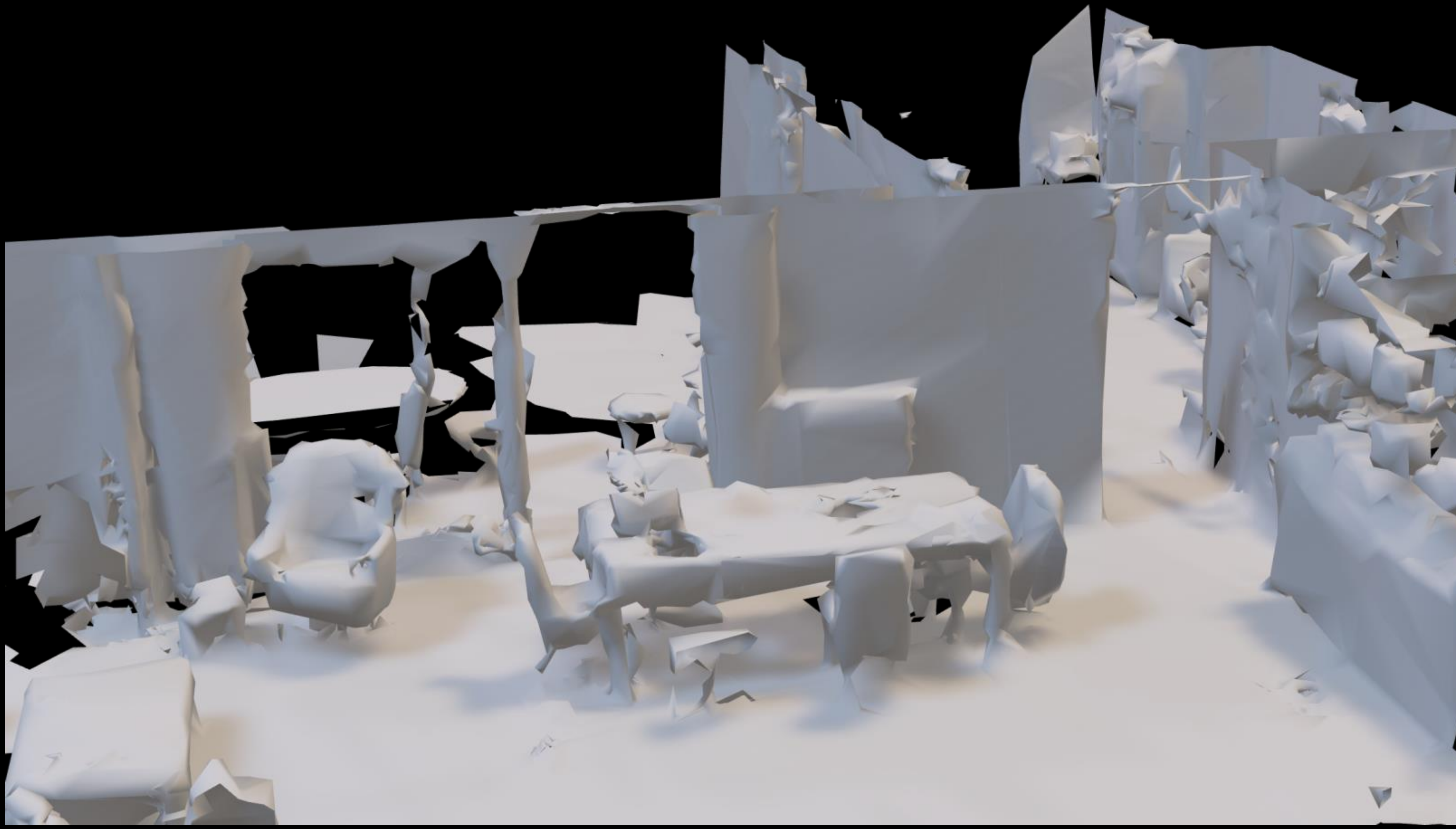


Microsoft Hololens

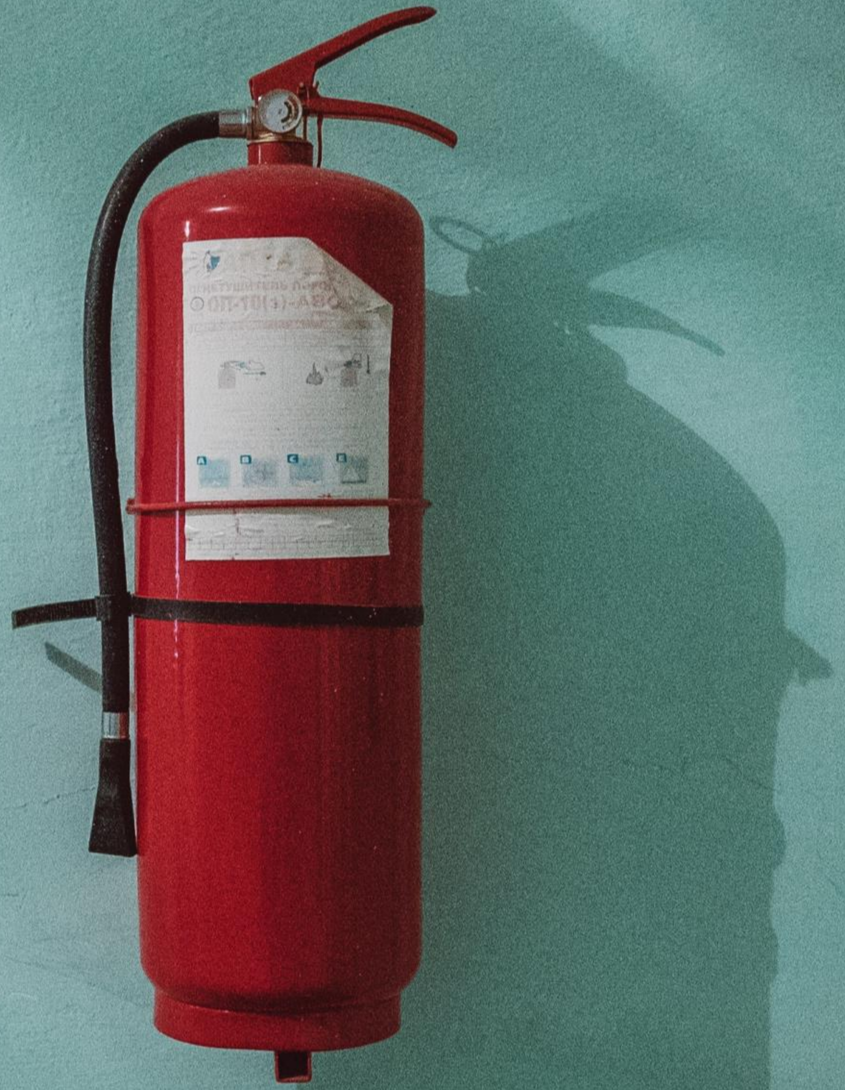
Hololens scan



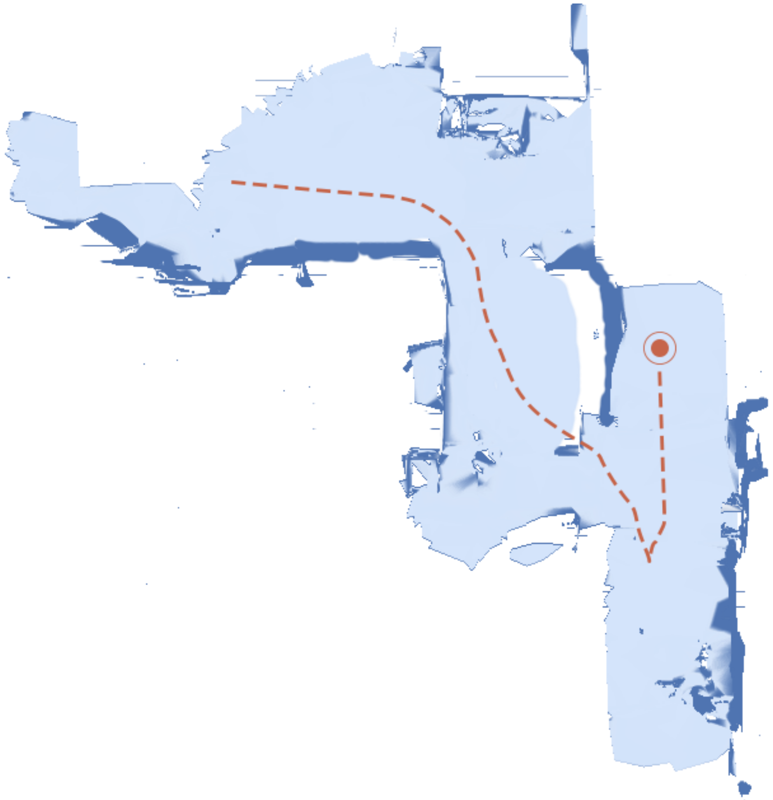
Hololens scan

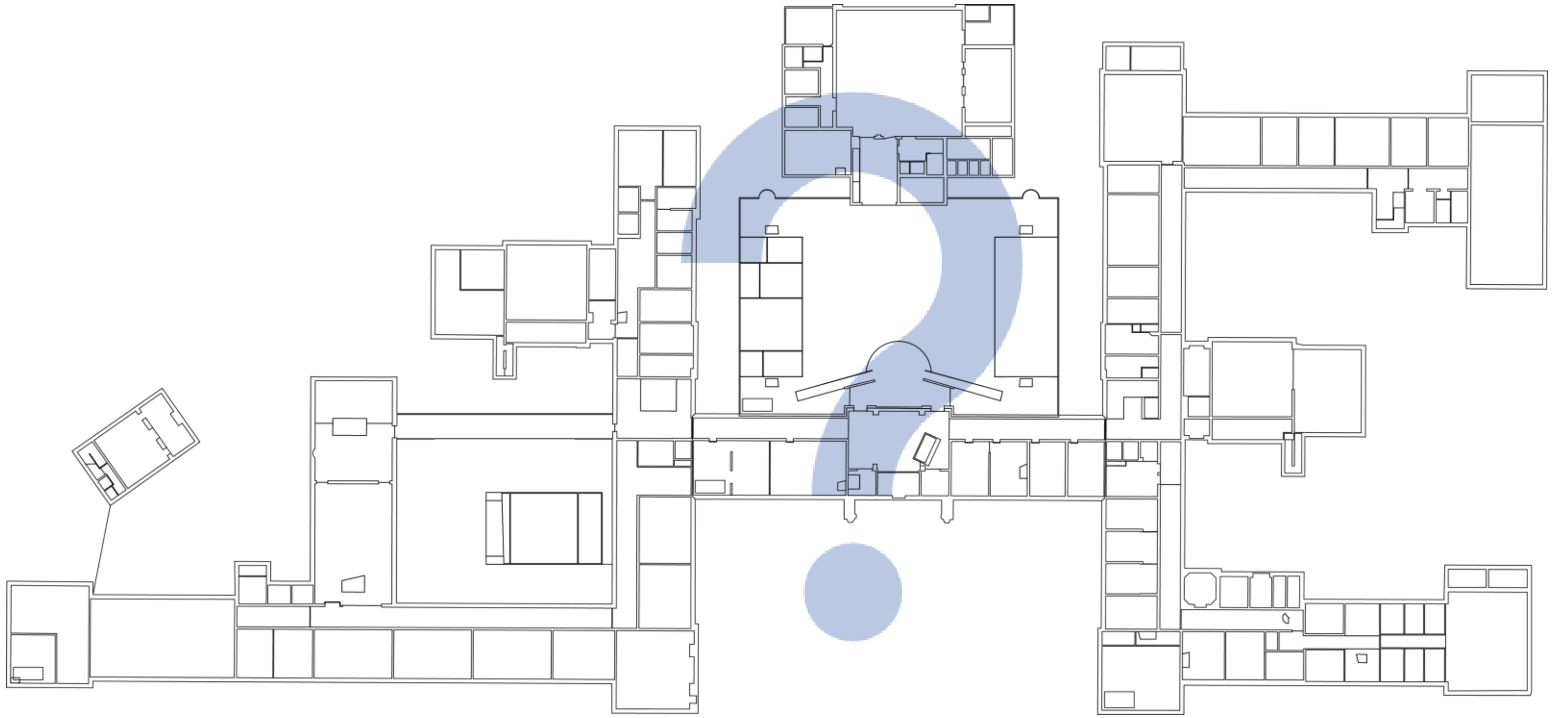


Emergency Response

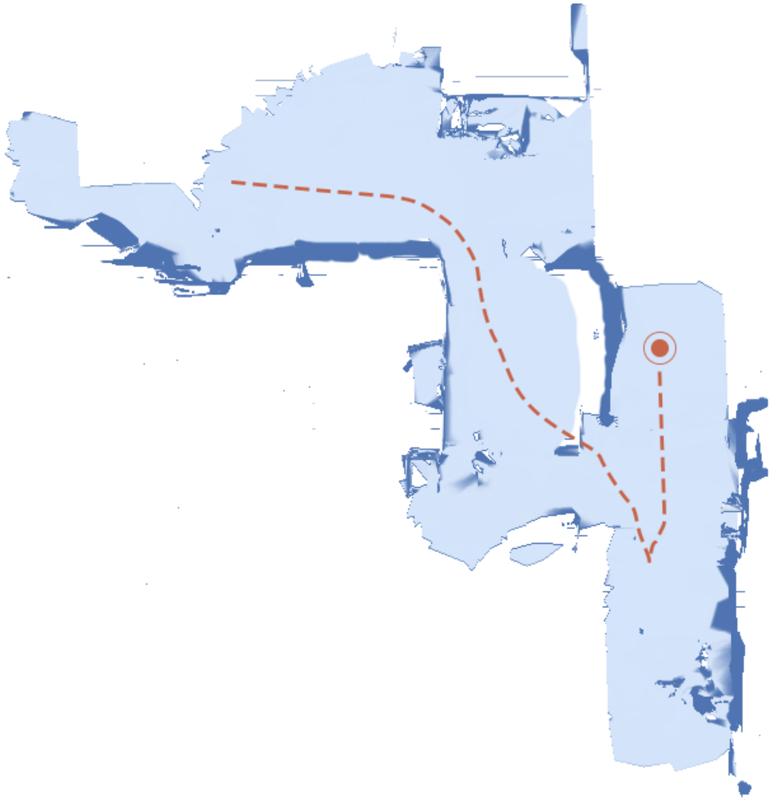


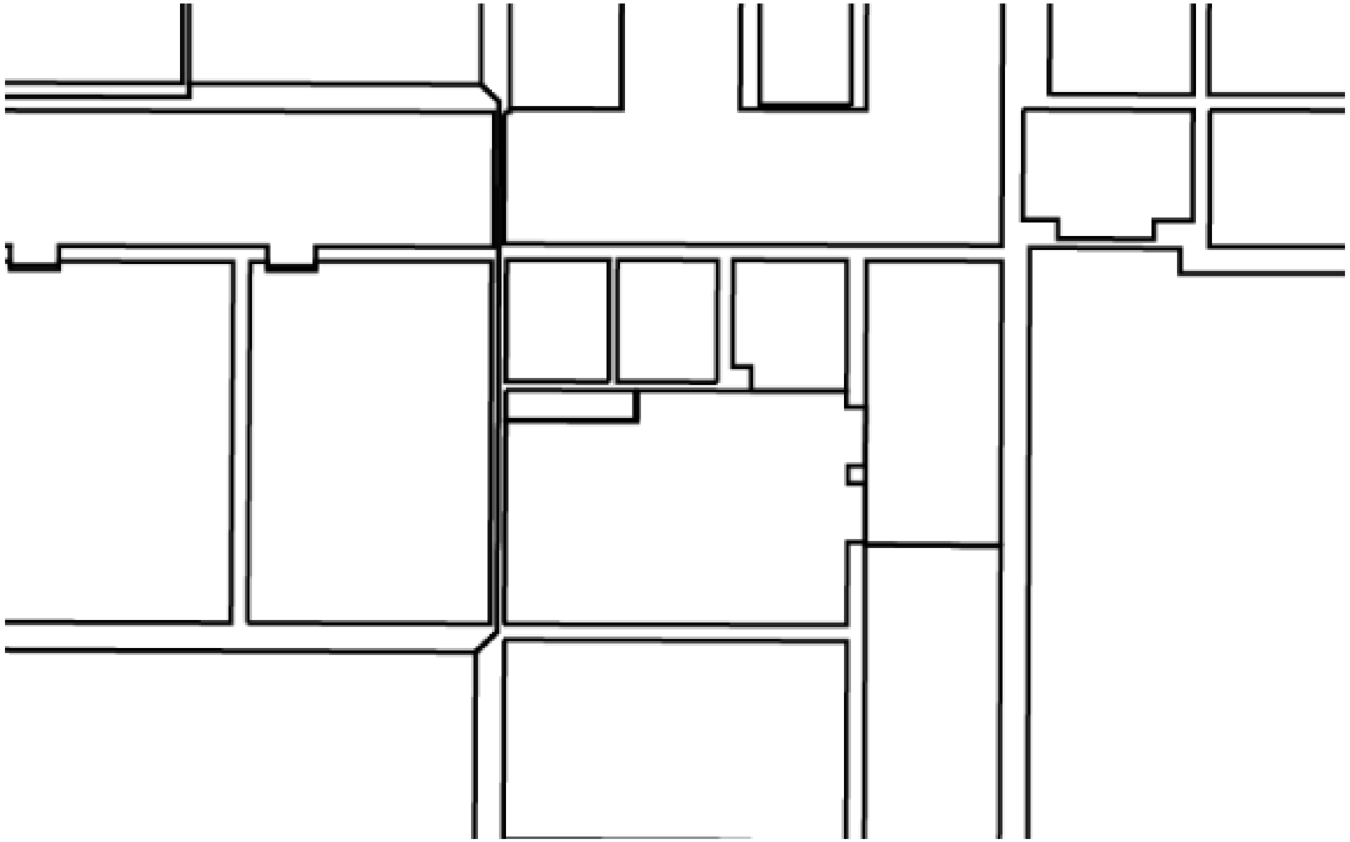
Problem statement (1) – Relation to floor plan



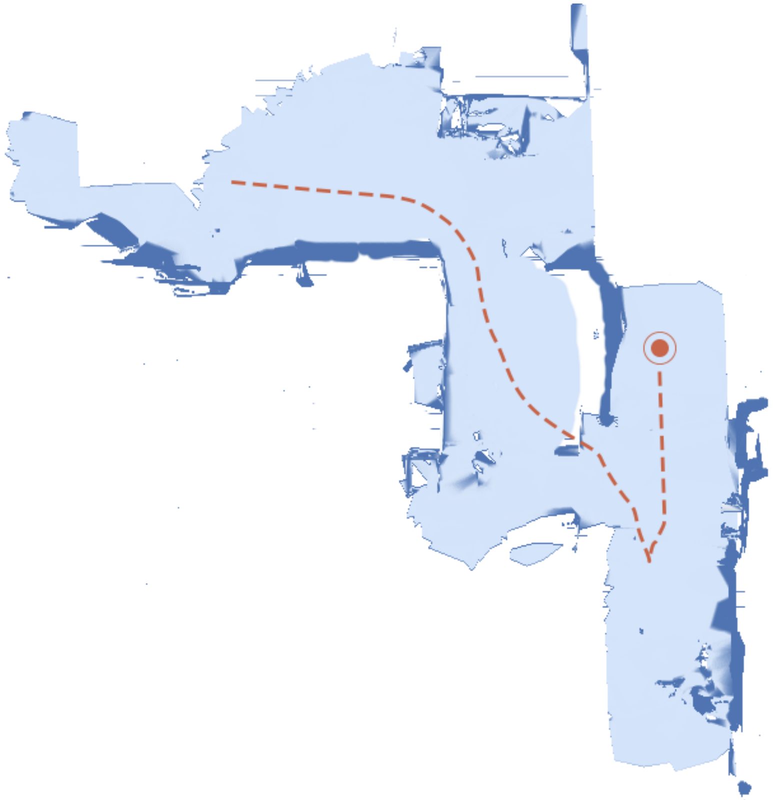


Problem statement (1) – Relation to floor plan





Problem statement (2) – Drift



Between 2-5 meter over 200 meter

Research question

"How can the Microsoft Hololens improve indoor positioning, using the on-the-fly produced mesh and an existing floor plan"

Research goal

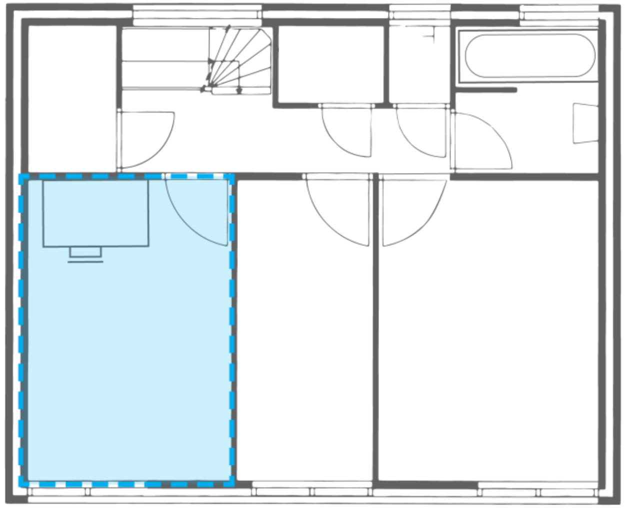
Develop indoor positioning application

Microsoft Hololens and floor plan

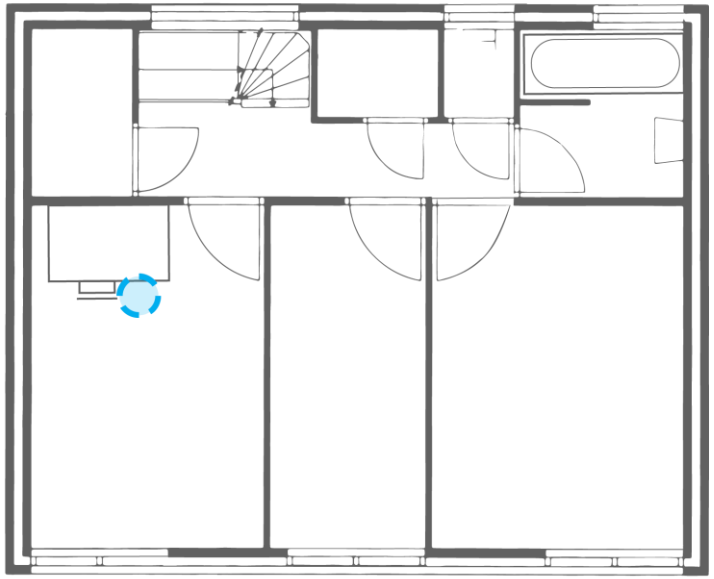
No infrastructure

Real-time

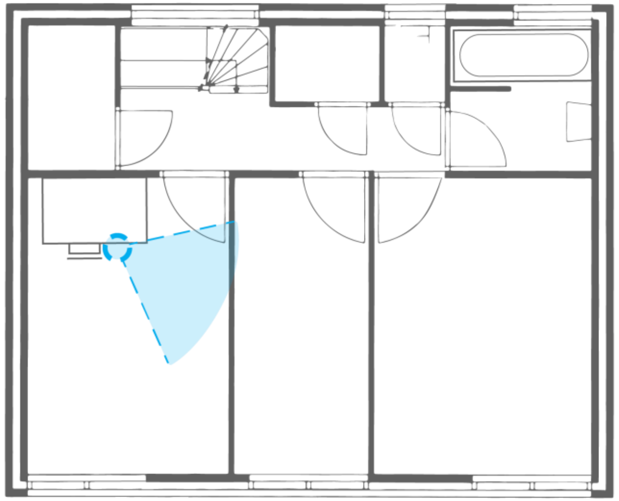
Location



Position

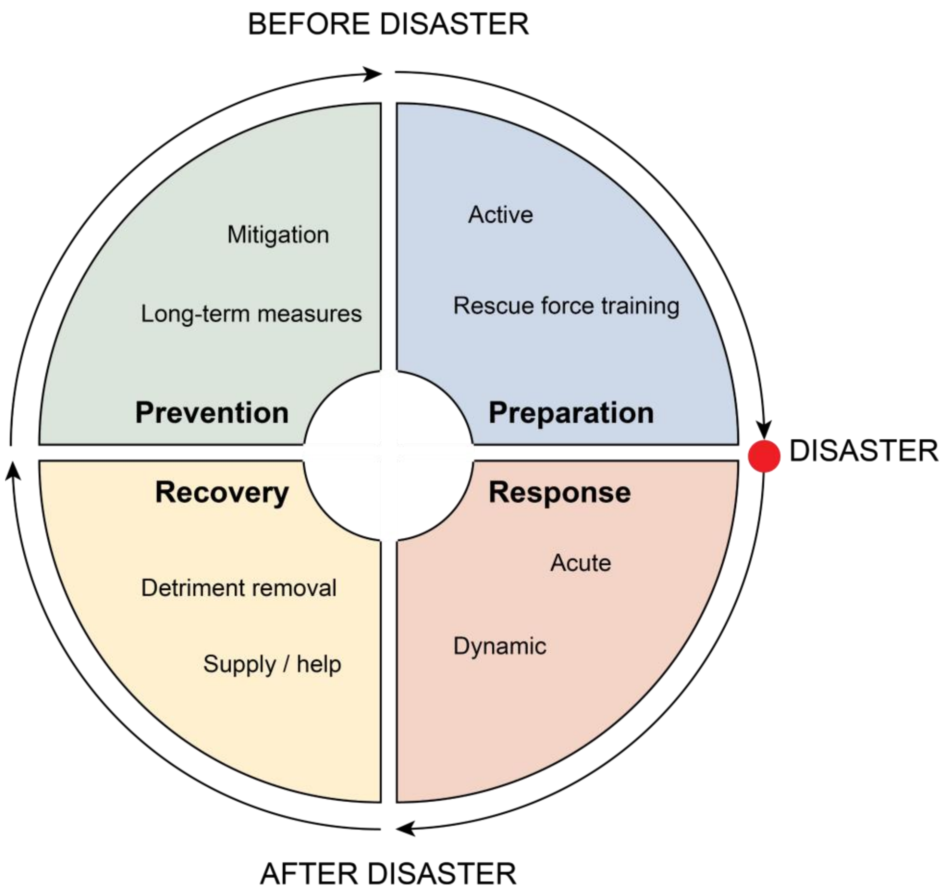


Pose



Emergency response

- Most extreme requirements
- Fast and efficient decision making
- Real-time and dynamic data



ACCURACY
<1M

CONSTANT
AVAILABILITY

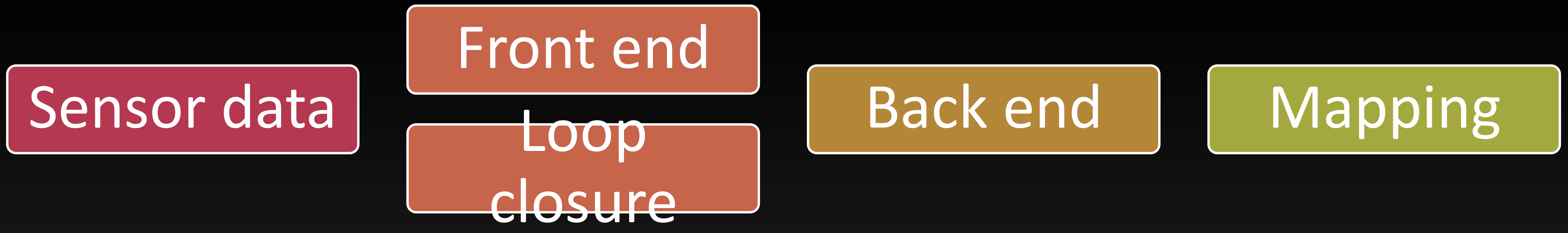
UNCERTAINTY
ESTIMATION

NO
PRE-INSTALLATION

Rantakokko et al. (2010)

Positioning requirements for ER

SLAM



SLAM

Sensor data

Front end

Loop closure

Back end

Mapping

Monocular
Binocular
RGB-D
Infrared

SLAM

Sensor data

Front end

Loop closure

Back end

Mapping

Feature method

Direct method

SLAM

Sensor data

Front end

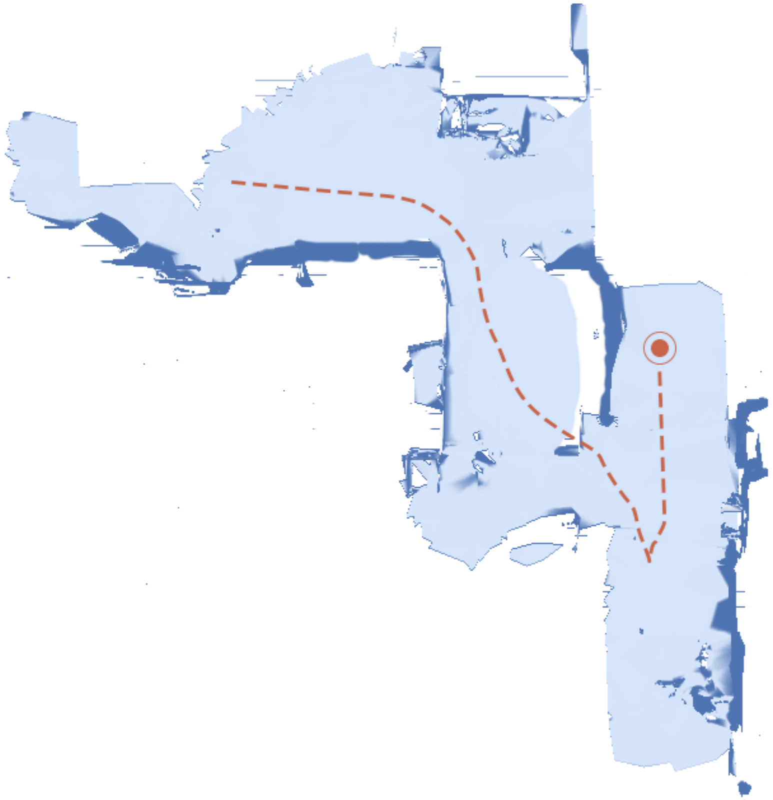
Loop closure

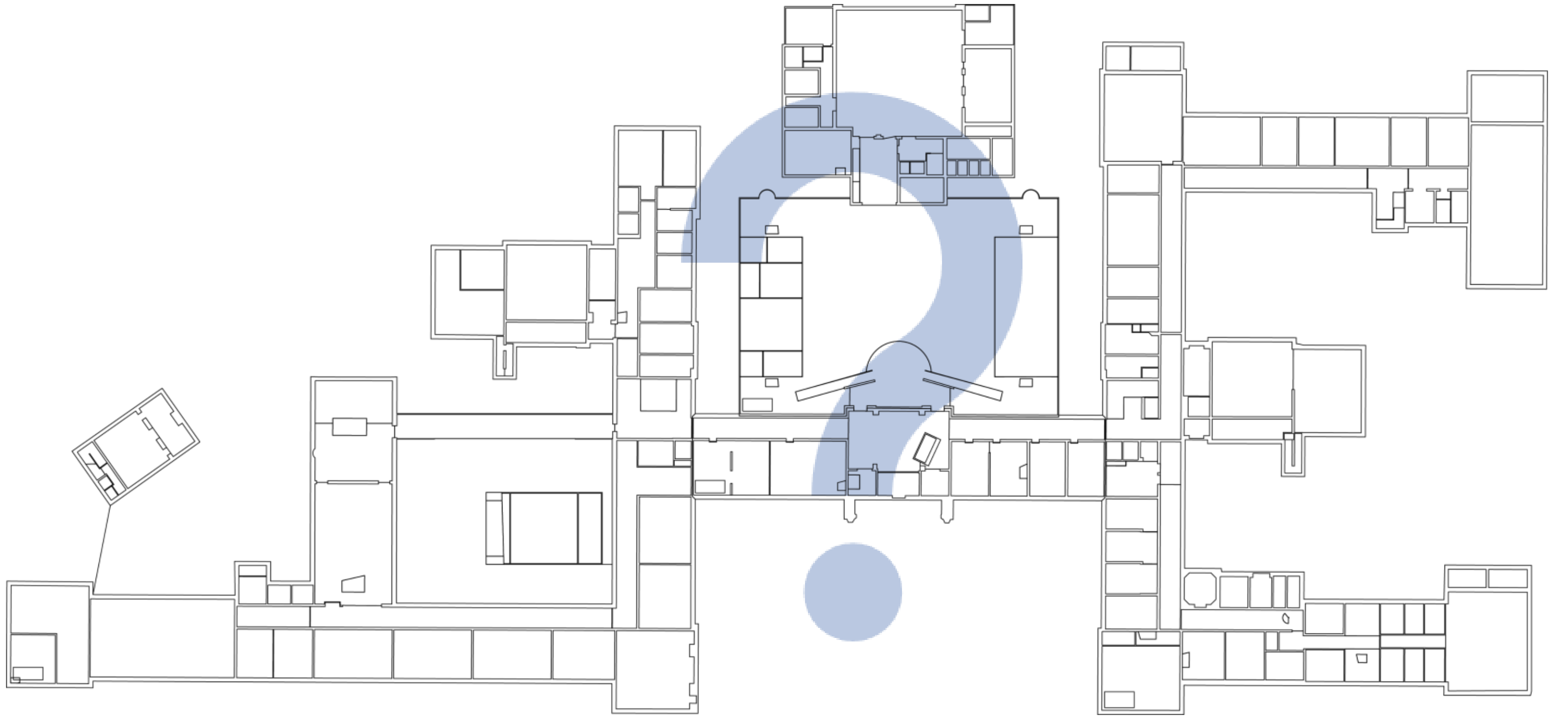
Back end

Mapping

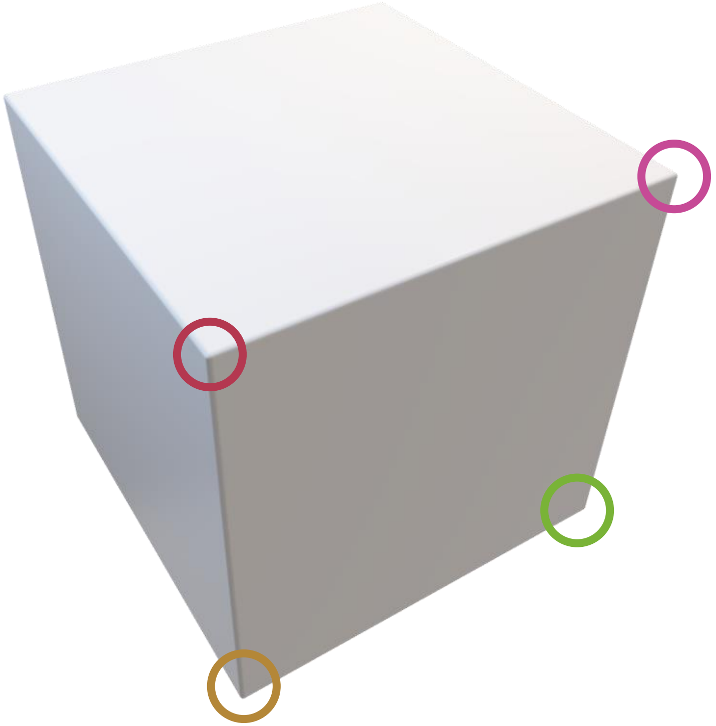
Posture optimization

Relation to floor plan

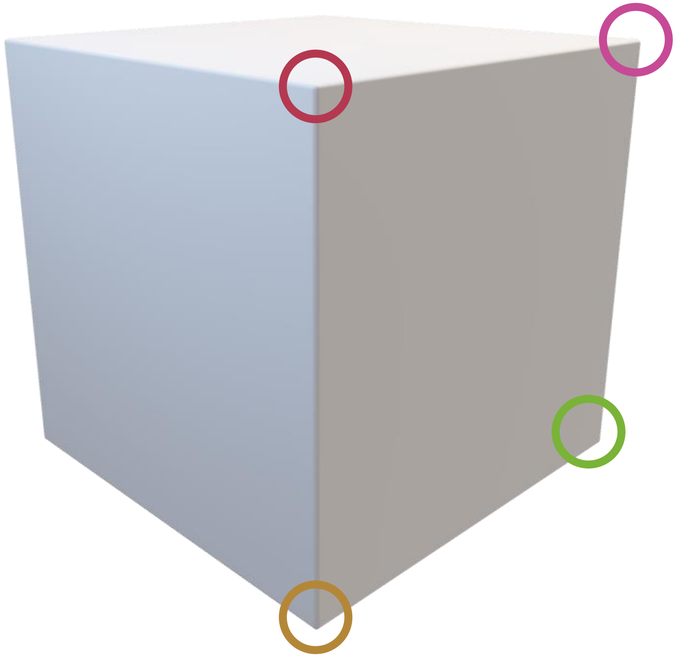




Spatial Matching

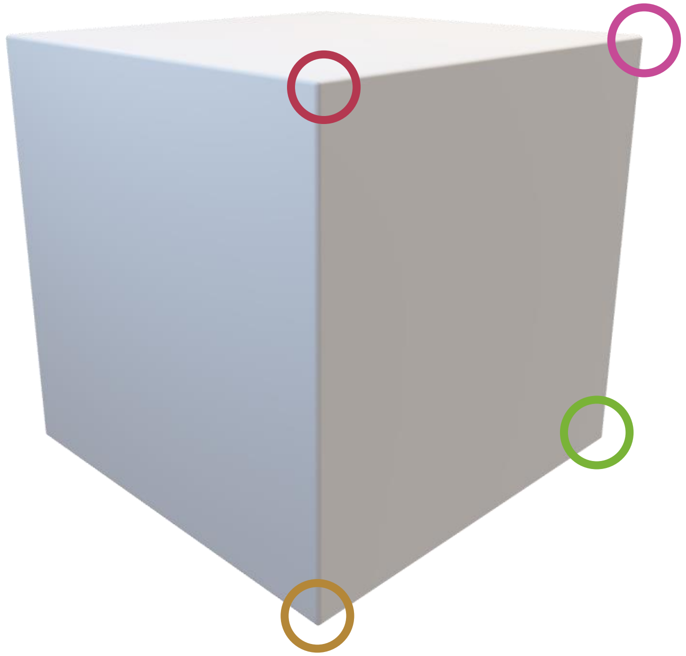


To be registered shape X



Reference shape Y

Spatial Matching

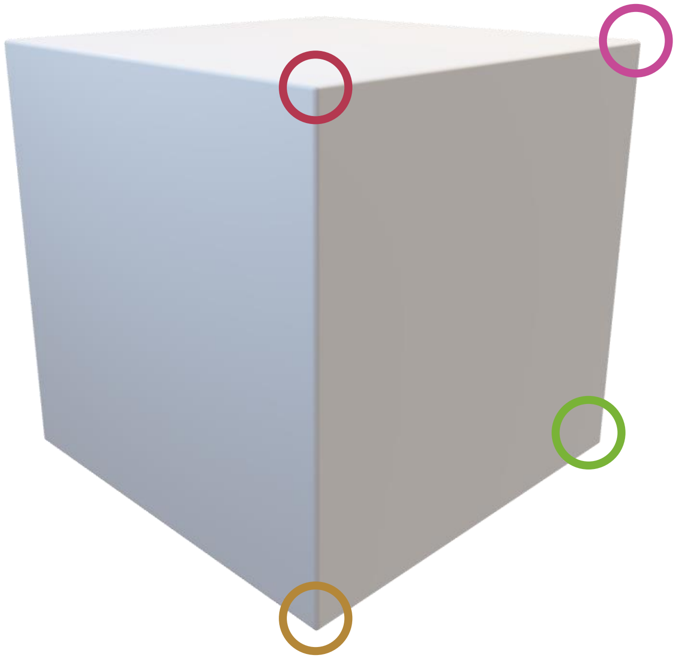


Registered shape X’

Transformation matrix M:

Rotation R
Translation T

X’ = RX + T



Reference shape Y

Spatial Matching

Iterative Closest Points (ICP)

- Most implemented
- Hololens uses ICP

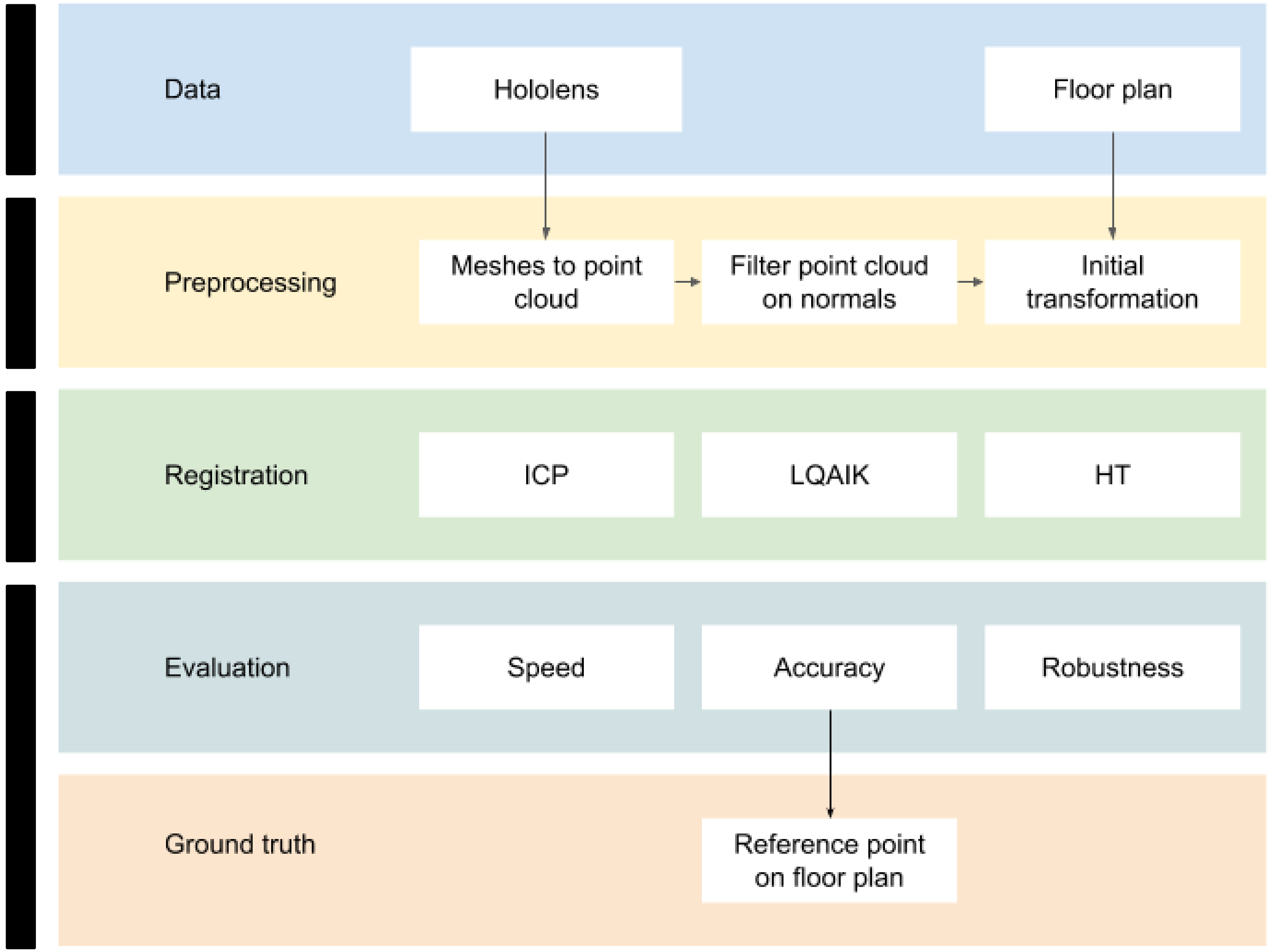
Instantaneous Kinematics (IK)

- Velocity vectors
- (Theoretically) faster convergence than ICP

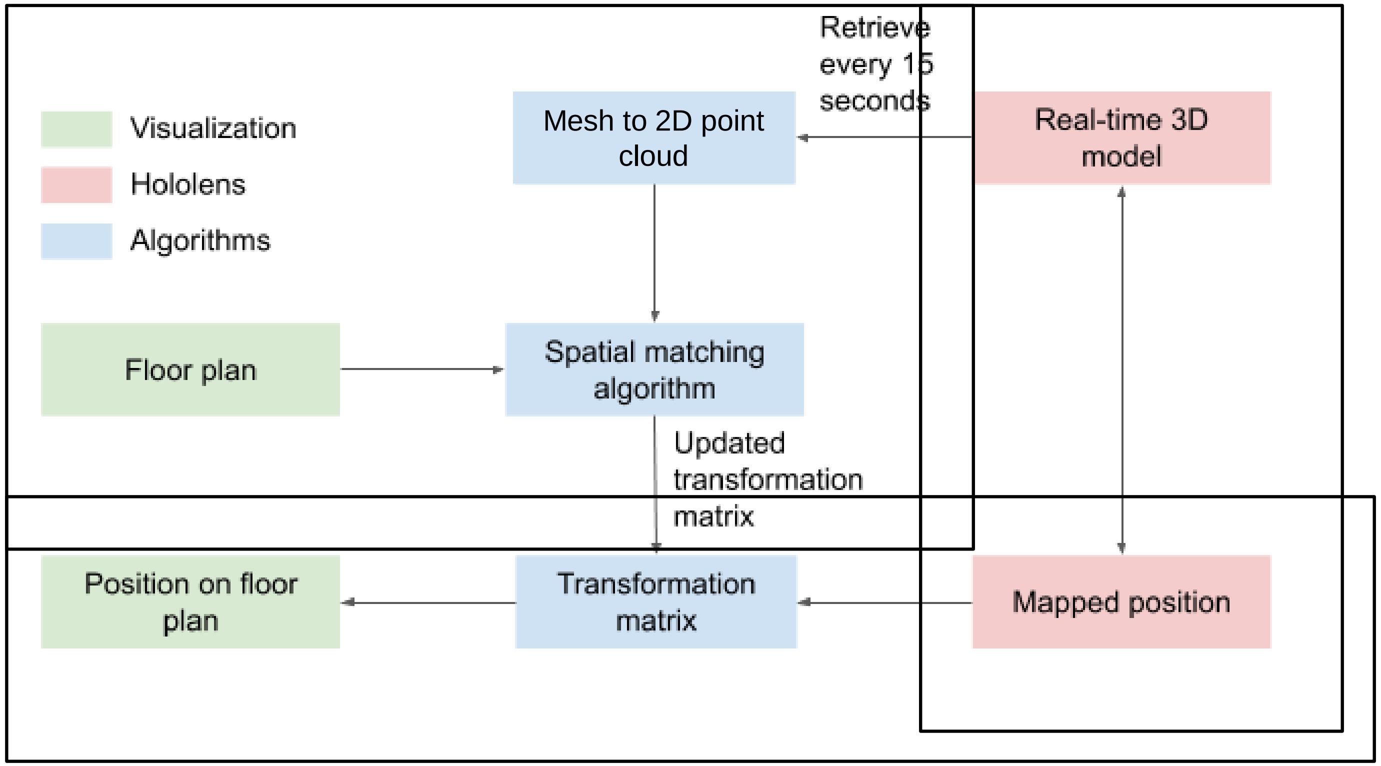
Hough Transform (HT)

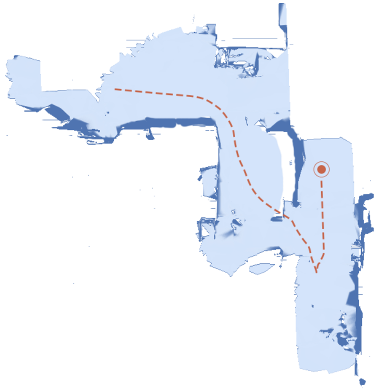
- Invariant rotation
- Combination 2D/3D data

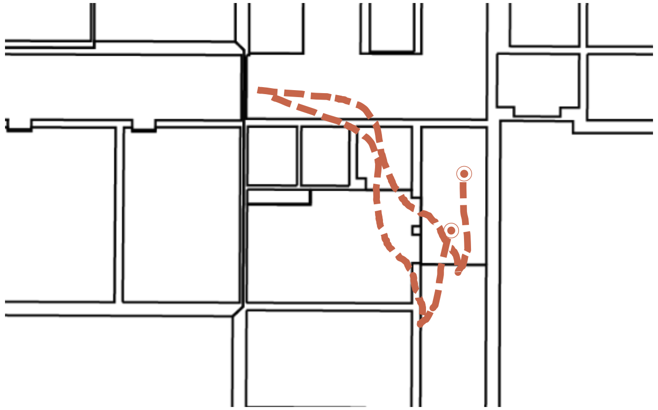
Research method



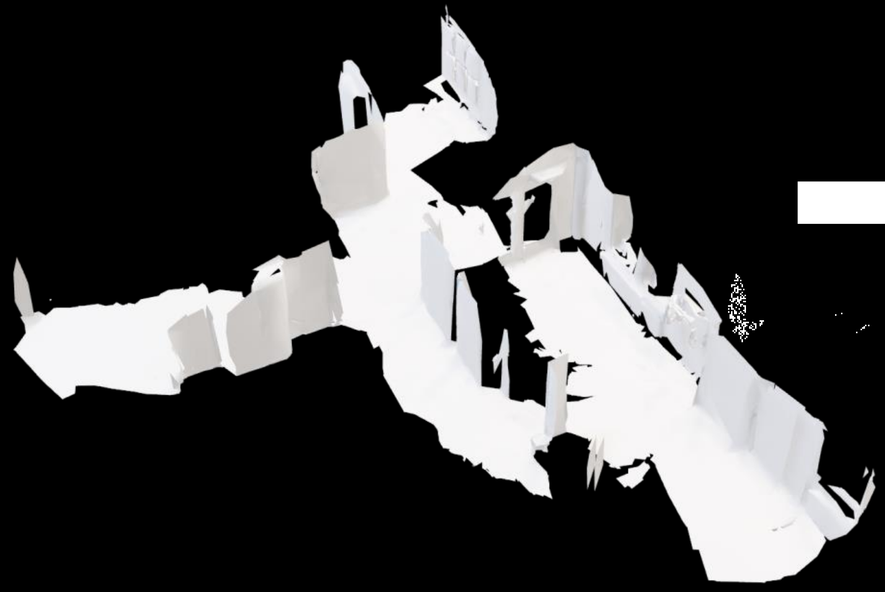
Workflow


Visualization
Hololens
Algorithms
Mesh to 2D point cloud
Retrieve every 15 seconds
Real-time 3D model
Floor plan
Spatial matching algorithm
Updated transformation matrix
Position on floor plan
Transformation matrix
Mapped position



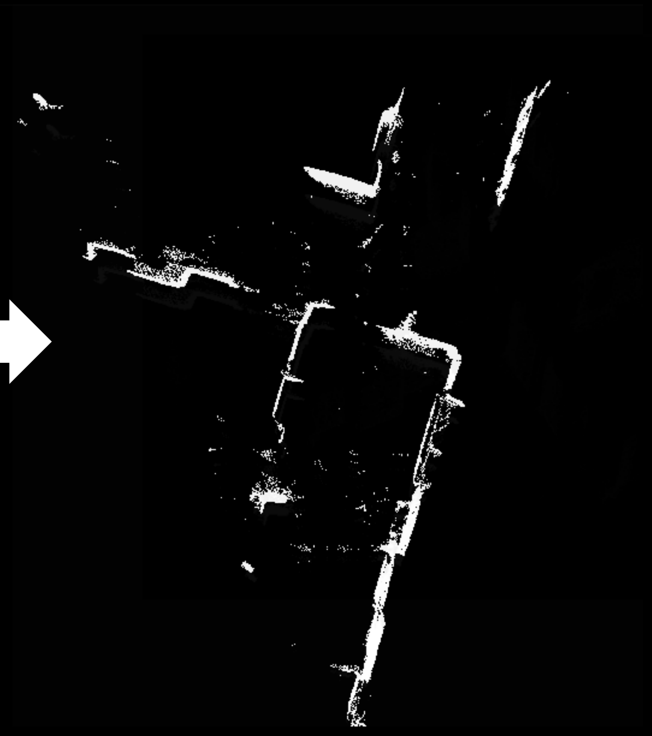


Mesh to point cloud



Mesh

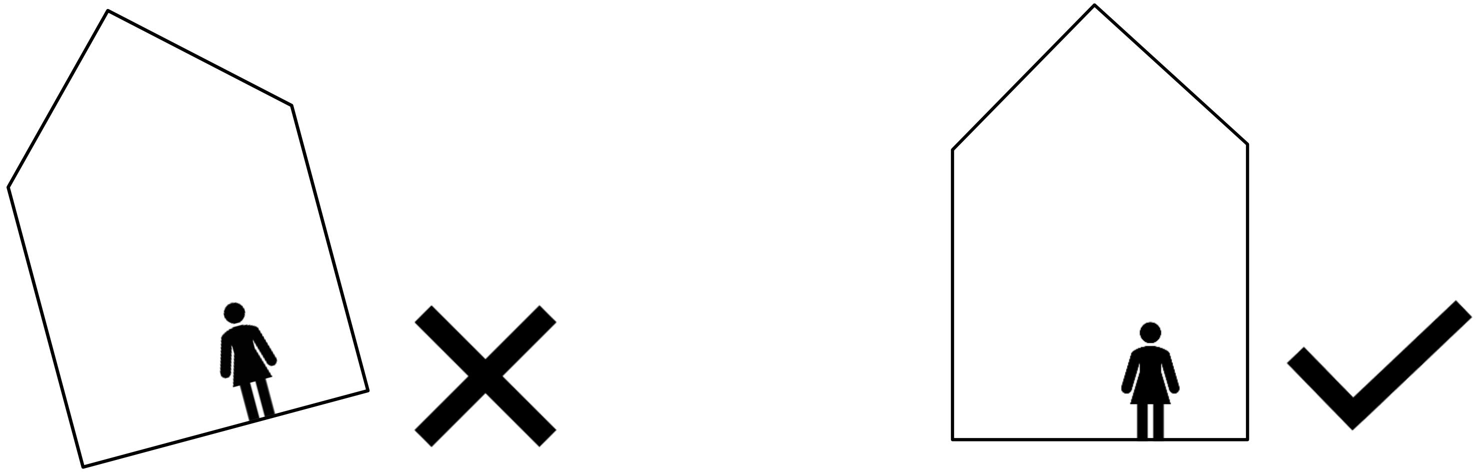
3D point cloud of walls

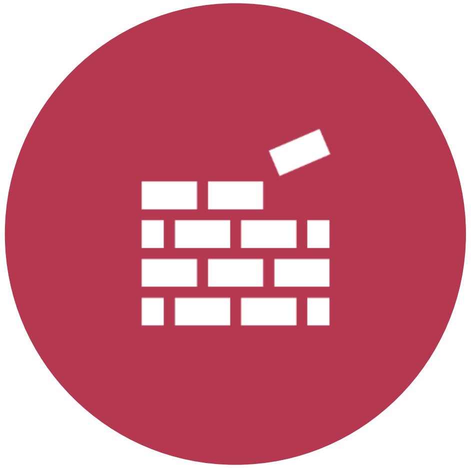


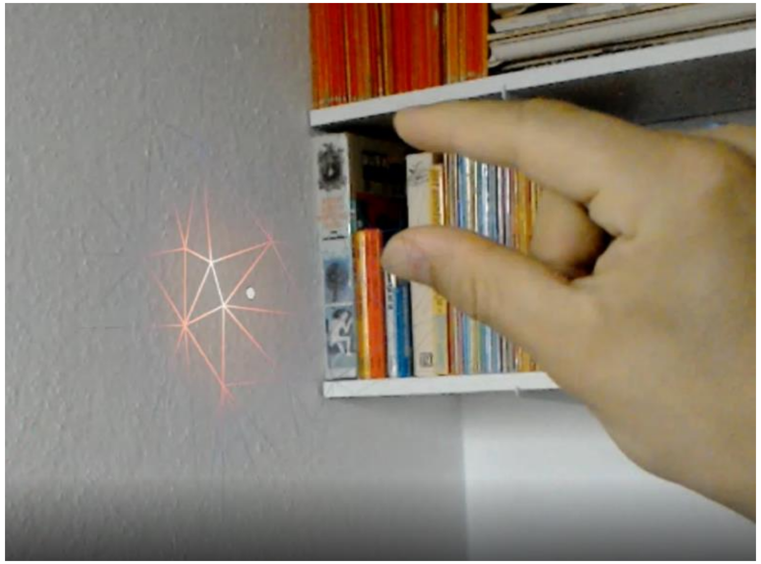
2D point cloud

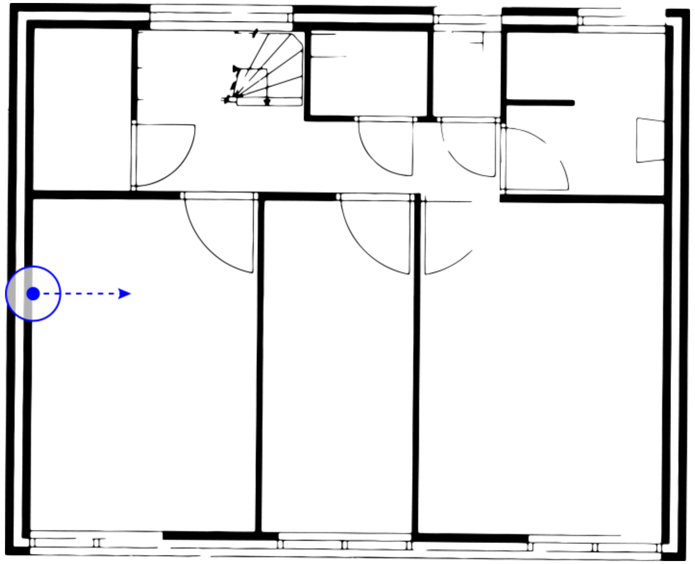
2D or 3D positioning

- Loss in dimension
- Complex buildings
- Correctly aligned horizontal plane









SELECTING A WALL

Scanned object

Floor plan

Initial transformation value

Iterative Closest Points (ICP)

Objective: Find a displacement vector q that minimizes the distances between displaced shape X and reference shape Y

$$q_t = \min_q d(\, q(X), Y)^2$$

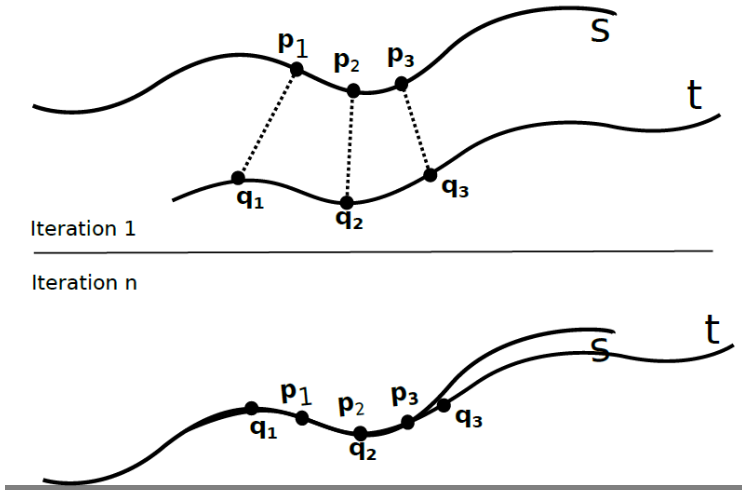
Find closest points:

$$d(x, Y) = \min_{y \in Y} |y - x|$$

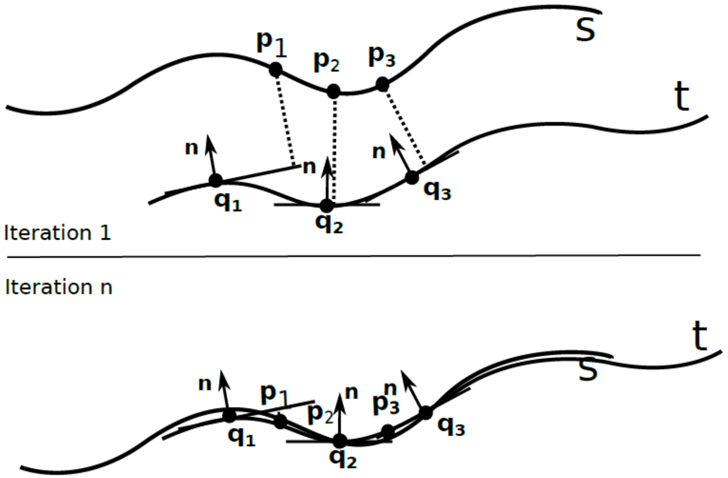
Least squares:

$$\min_q \frac{1}{N} \sum_{i=1}^{N} \|y_i - R(q_R)x_i - q_T\|^2$$

ICP optimizations



Point-to-point

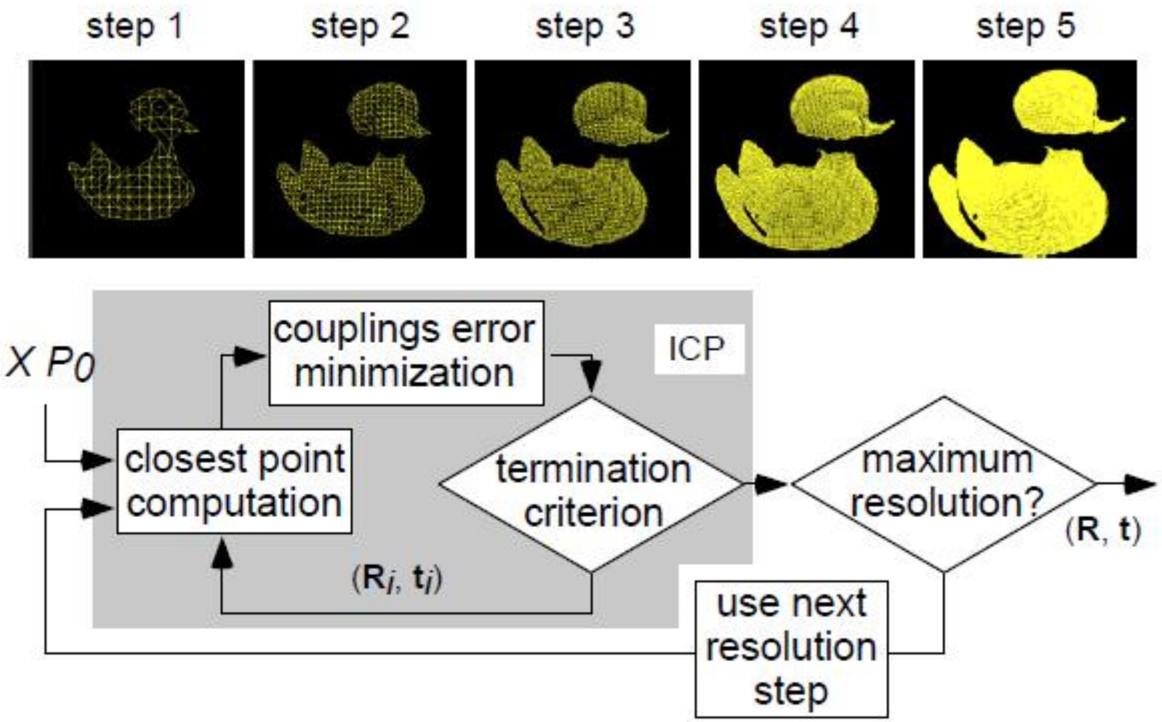


Point-to-plane / point-to-line

$$\min_{q} \frac{1}{N} \sum_{i=1}^{N} \|y_i - R(q_R)x_i - q_T\|^2$$

$$\min_{q} \frac{1}{N} \sum_{i=1}^{N} \left\| \left((R(q_R)x_i + q_T) - y_i \right) n_i \right\|^2$$

ICP optimizations



Multi-resolution (Jost & Hugli, 2003)

Instantaneous Kinematics (IK)

Velocity vector:

$$v(x_i) = \bar{\boldsymbol{c}} + \boldsymbol{c} \times x_i$$

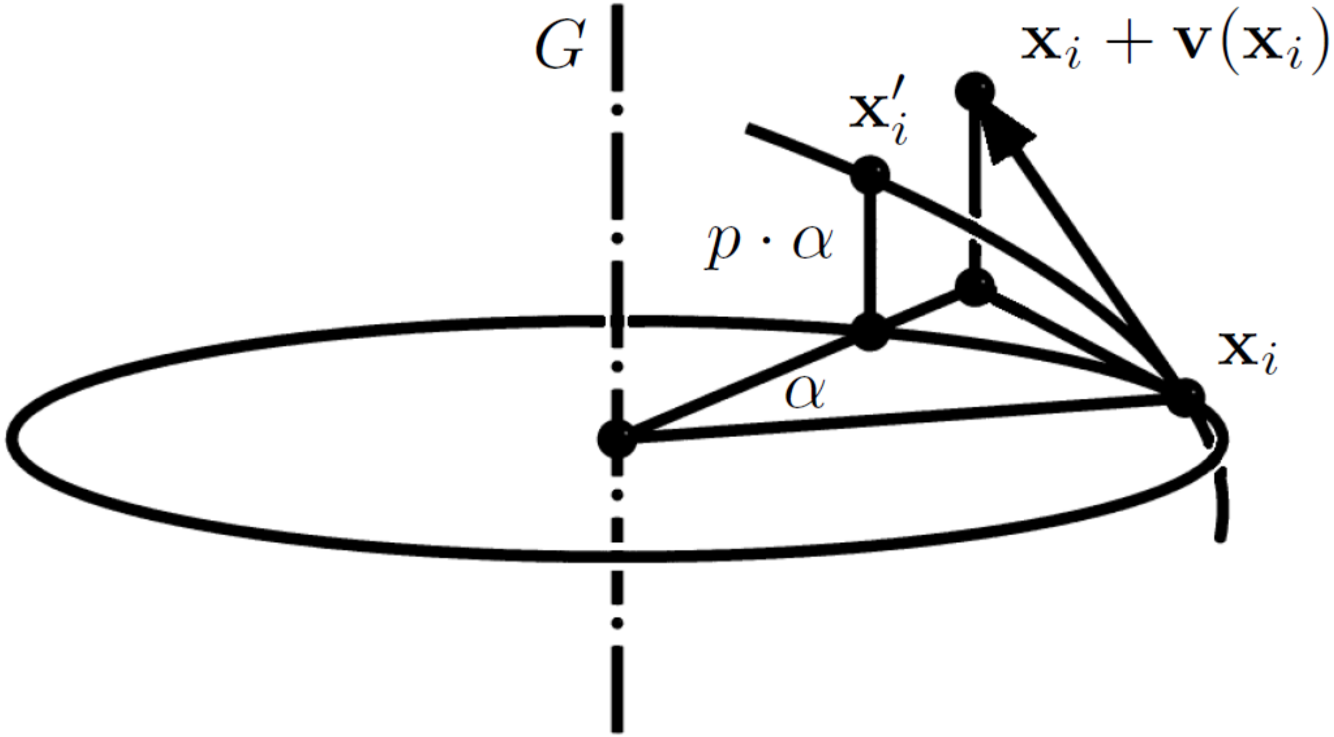
$\bar{\boldsymbol{c}}$: velocity vector at origin (≈translation)

$\boldsymbol{c}$: Darboux vector / vector of angular velocity (≈rotation)

Objective:

$$\min_{v(x)} \sum_{i=0}^{N} F(x_i + v(x_i), y_i)$$

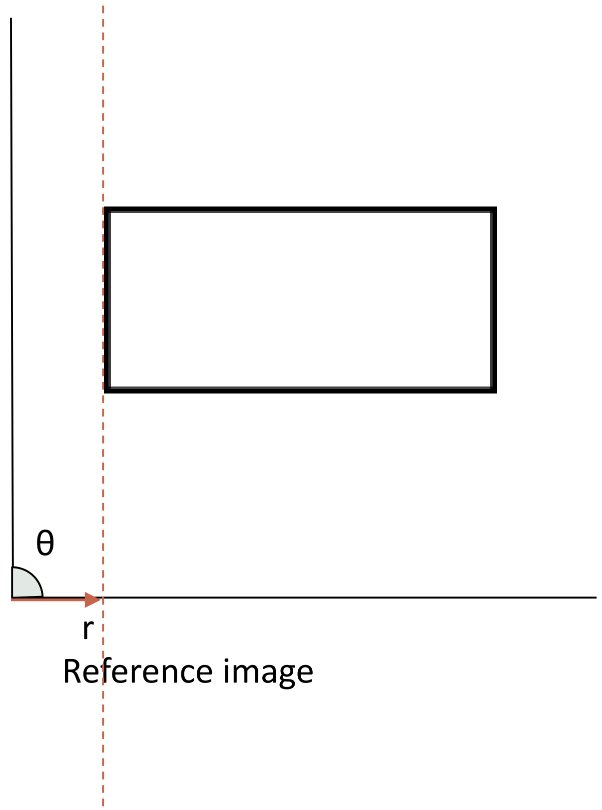
Solving rigidity constraint of IK

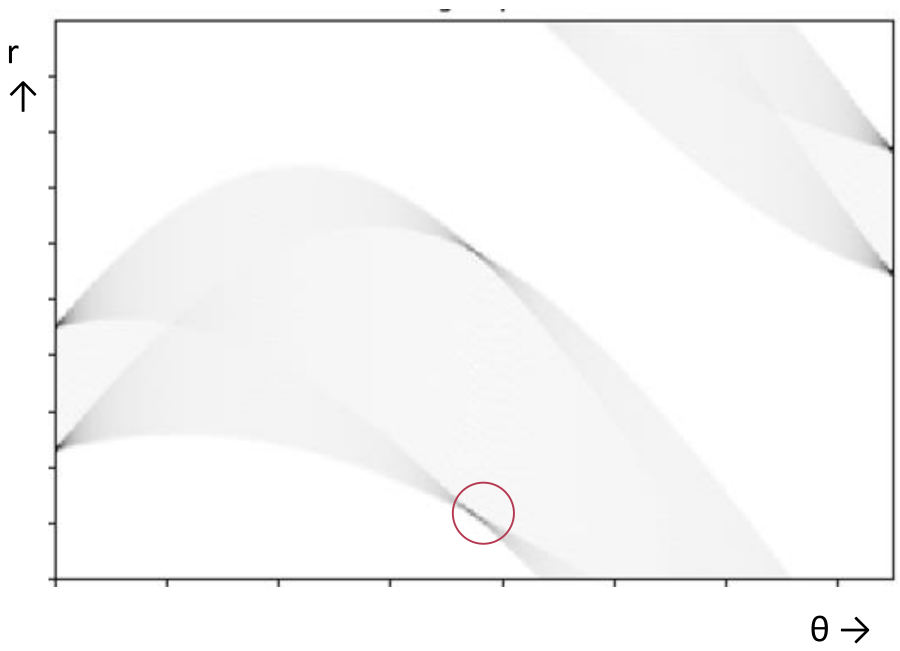


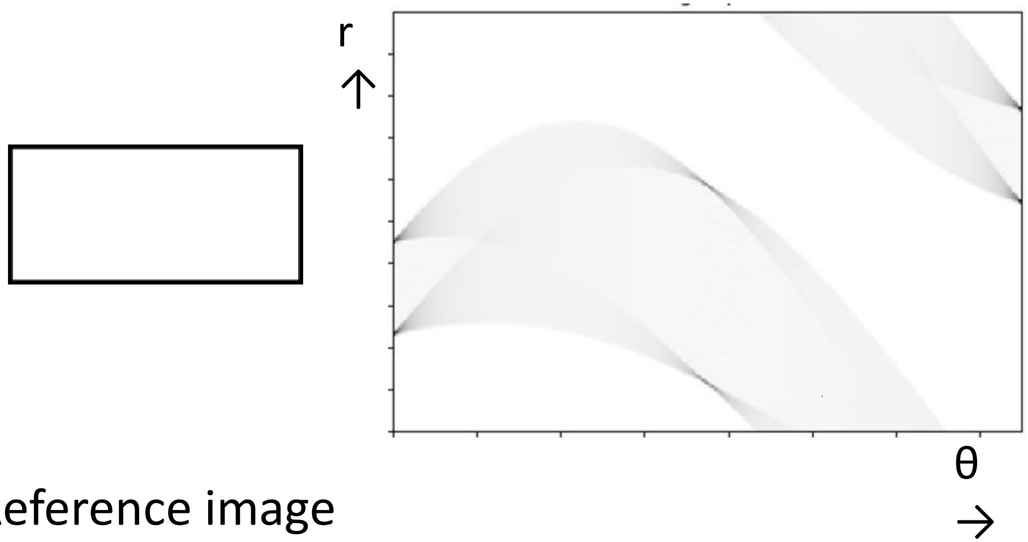
Hough Transform (HT)

Hough domain

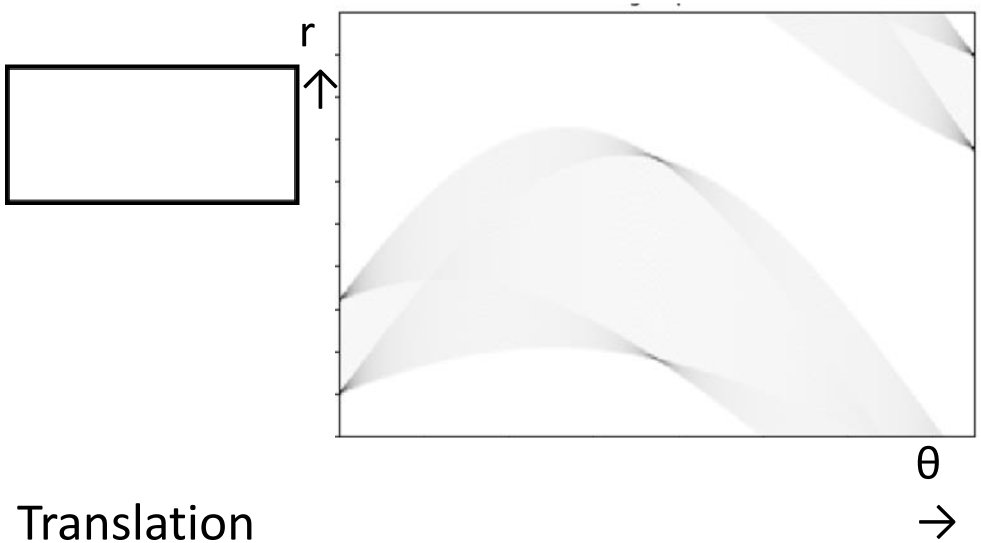
θ: angle
r: distance to origin







Reference image



Translation

Rotation is translation-invariant!

Plane/line finding using Hough Space

Finding the correct rotation

Finding the correct translation

Hough transform (HT)

Plane or line finding in HT

(x_{pt}, y_{pt}): point coordinates

(x_n, y_n): point normal vector

θ: $\tan^{-1}\frac{y_n}{x_n}$ (orientation)

r: $|x_{pt}\cos\theta + y_{pt}\sin\theta|$ (distance to origin)

Rotation in HT

Objective:

$$\max_{\gamma} \sum_{i} \sum_{\rho 1} H^{(floorplan)}(\theta_i, \rho_1) \sum_{\rho 2} H^{(pointcloud)}(\theta_i + \gamma, \rho_2)$$

In other words:

Find a rotation γ where the alignment between the Hough image of the reference shape $H^{(r)}$ and the rotated Hough image of the registered shape $H^{(q)}$ is highest

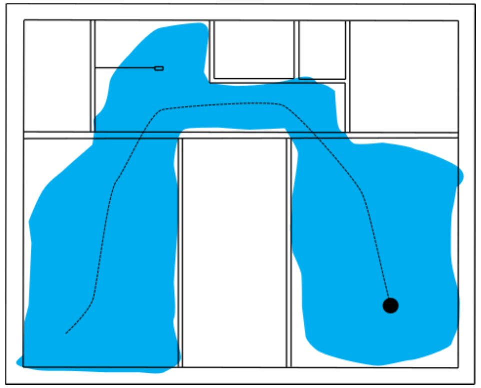
Translation in HT

Euclidean space

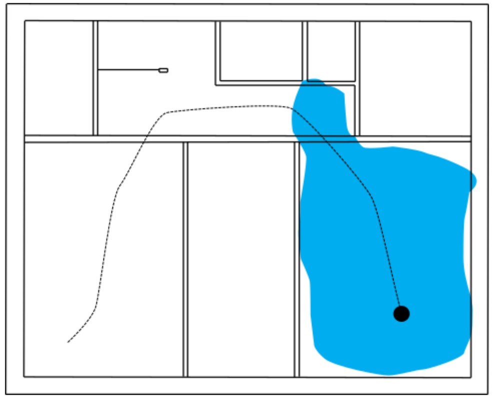
Using Iterative Closest Points with only translation:

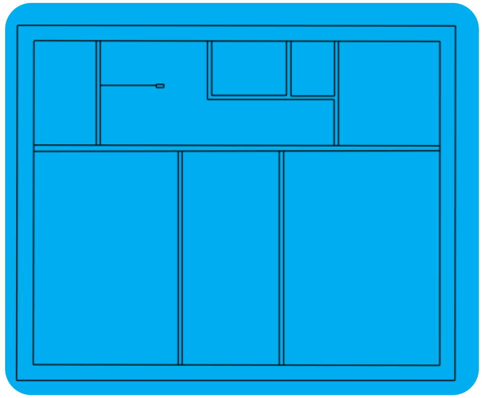
$$\frac{1}{N}\sum_{i} y_i - x_i$$

Configurations

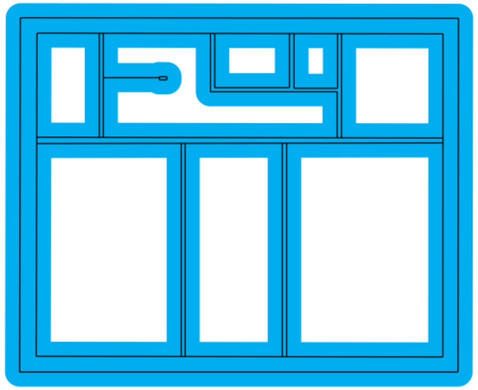


Use meshes that are scanned since start of the process OR Use meshes that are scanned since last registration





No buffer around walls OR 50cm buffer around walls



Evaluation criteria

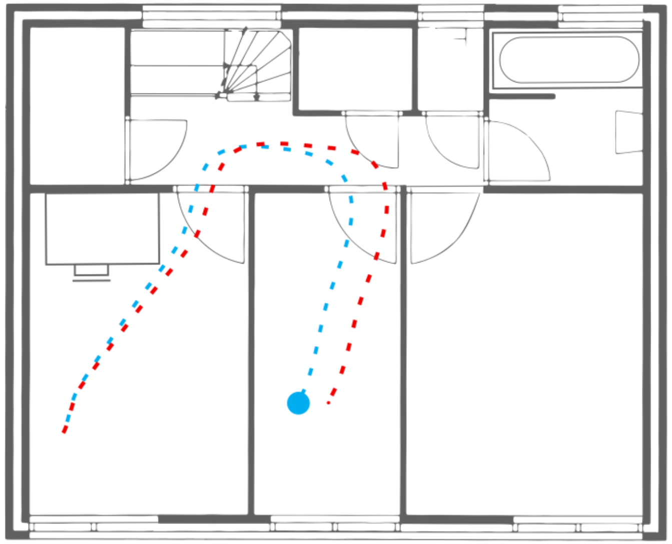
ACCURACY

COMPUTATION TIME

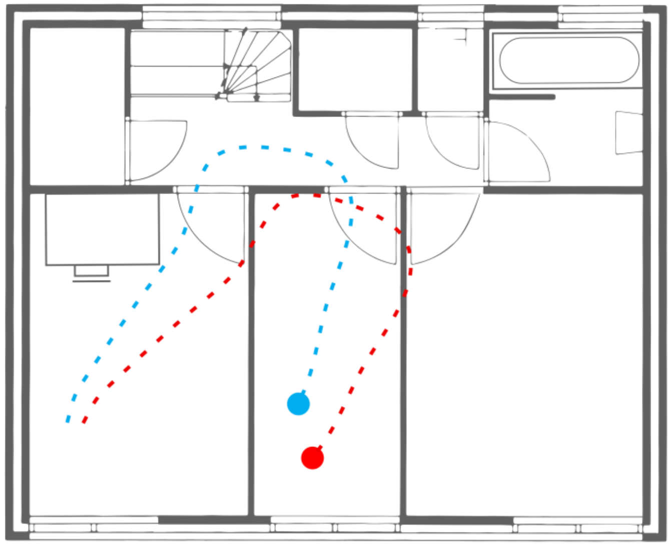
ROBUSTNESS

Accuracy

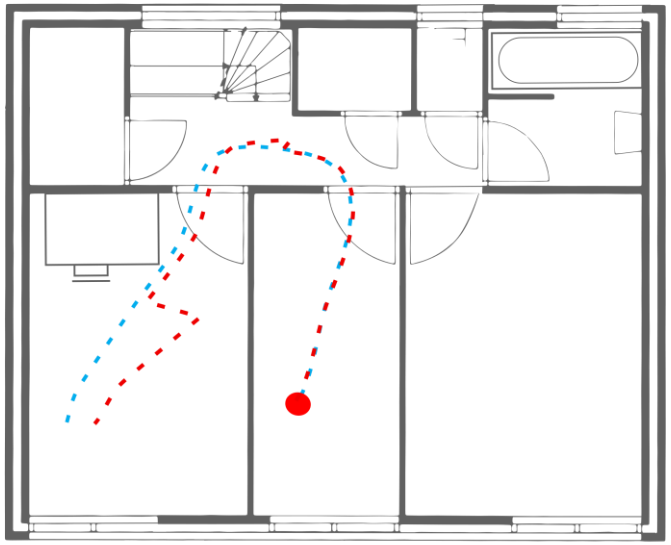
Drift



Initial transformation



Spatial matching techniques



Accuracy <1m

Computation time

< 15 seconds

Well scalable:

- 100 points
- 1000 points
- 10000 points
- 100000 points
- Different sizes of the scanned mesh

Robustness

Behaviour of algorithms on special cases:

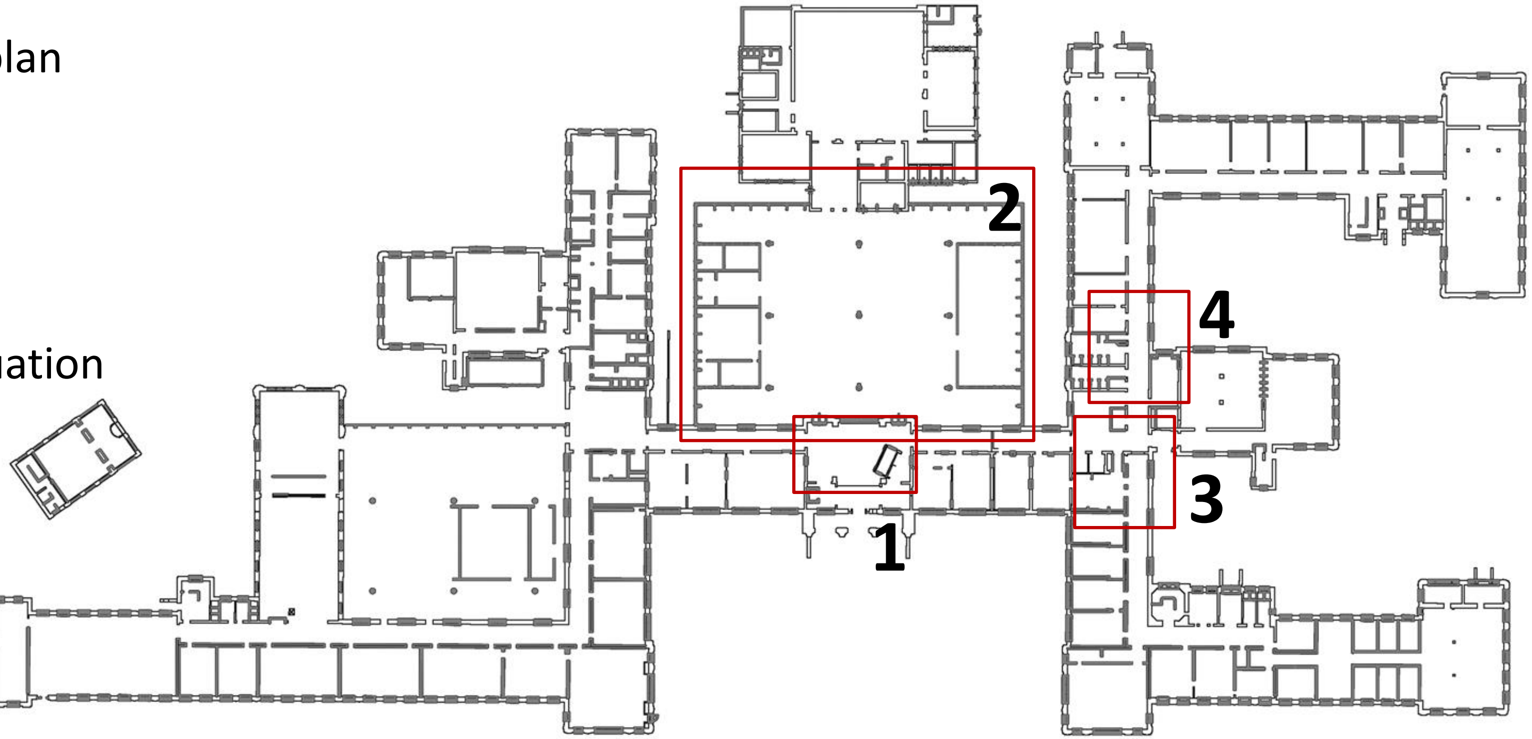
- Doors

- Walls that do not exist in the floor plan

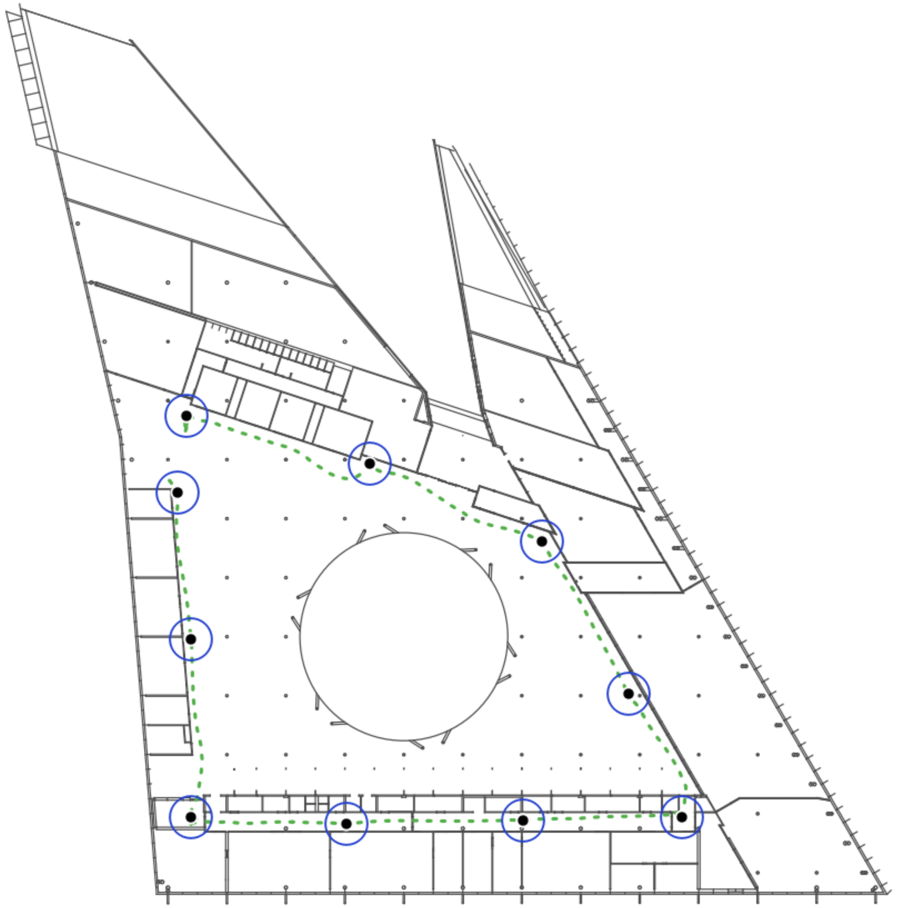
- Walls that exist in the floor plan, but not in the present situation

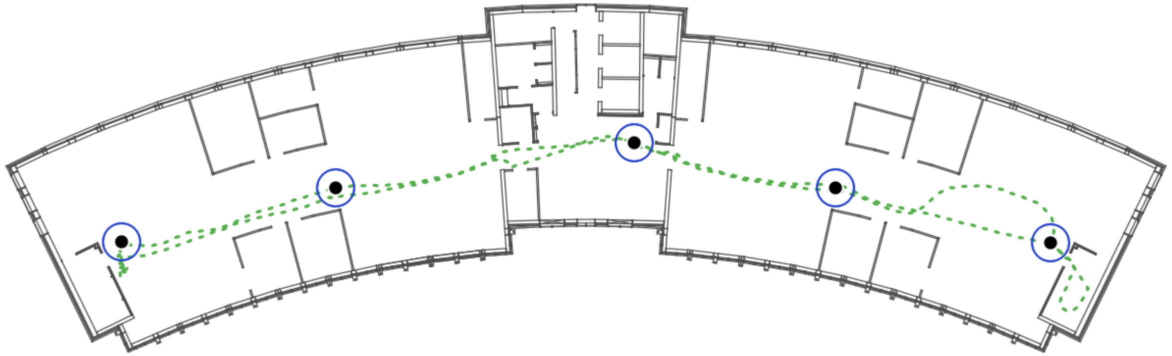
- Large spaces (>20x20 meter halls)

1	‘Wall’ that does not exist in the floor plan
2	>20x20 meter hall
3	Wall that does not exist in present situation
4	Door



TU Delft Architecture building

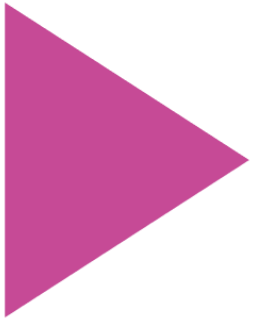




TU Delft Library building | CGI Rotterdam office

Indoor positioning application

Real-time



Simulation

Functionalities

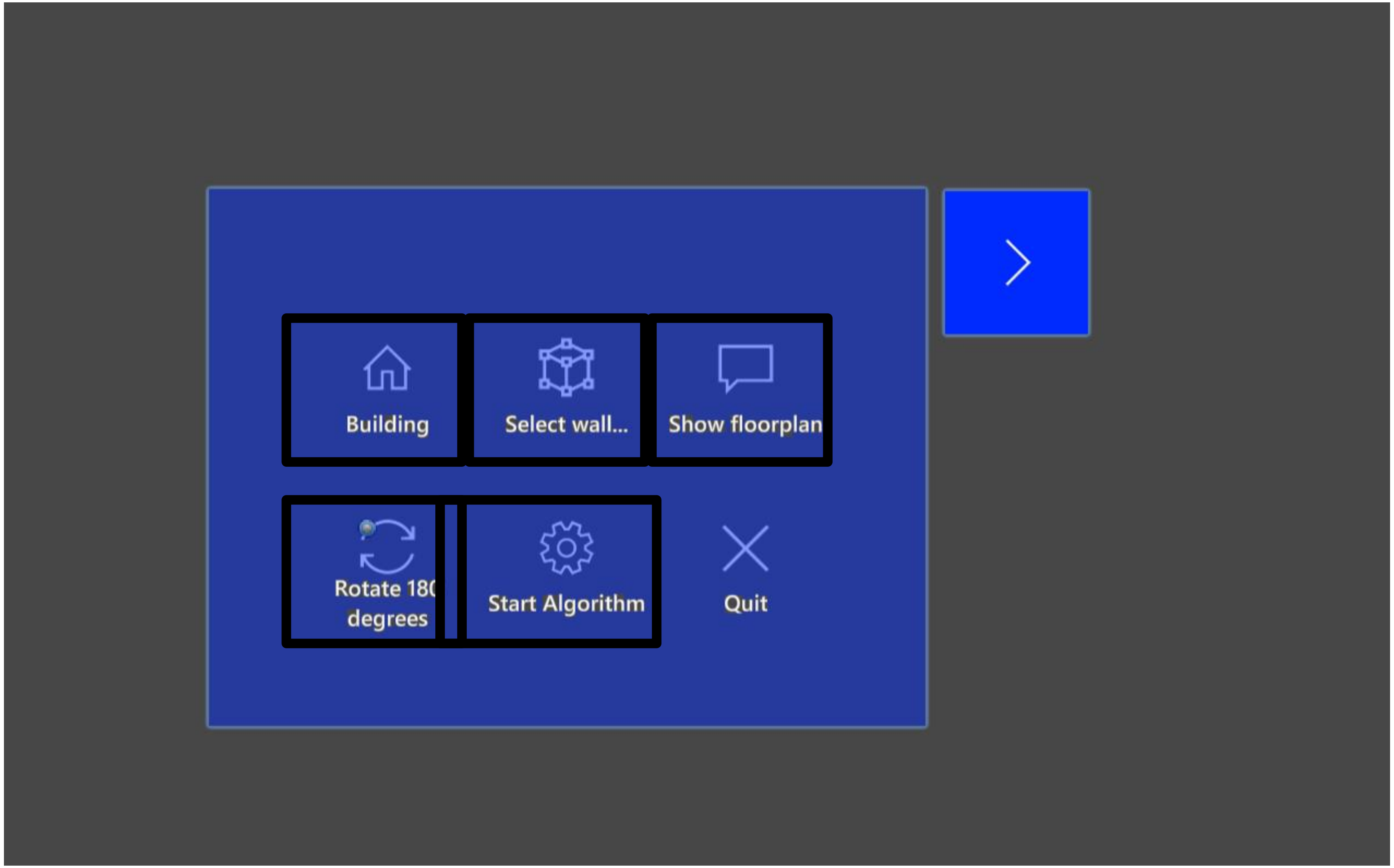
PROCESS INPUT OF HOLOLENS

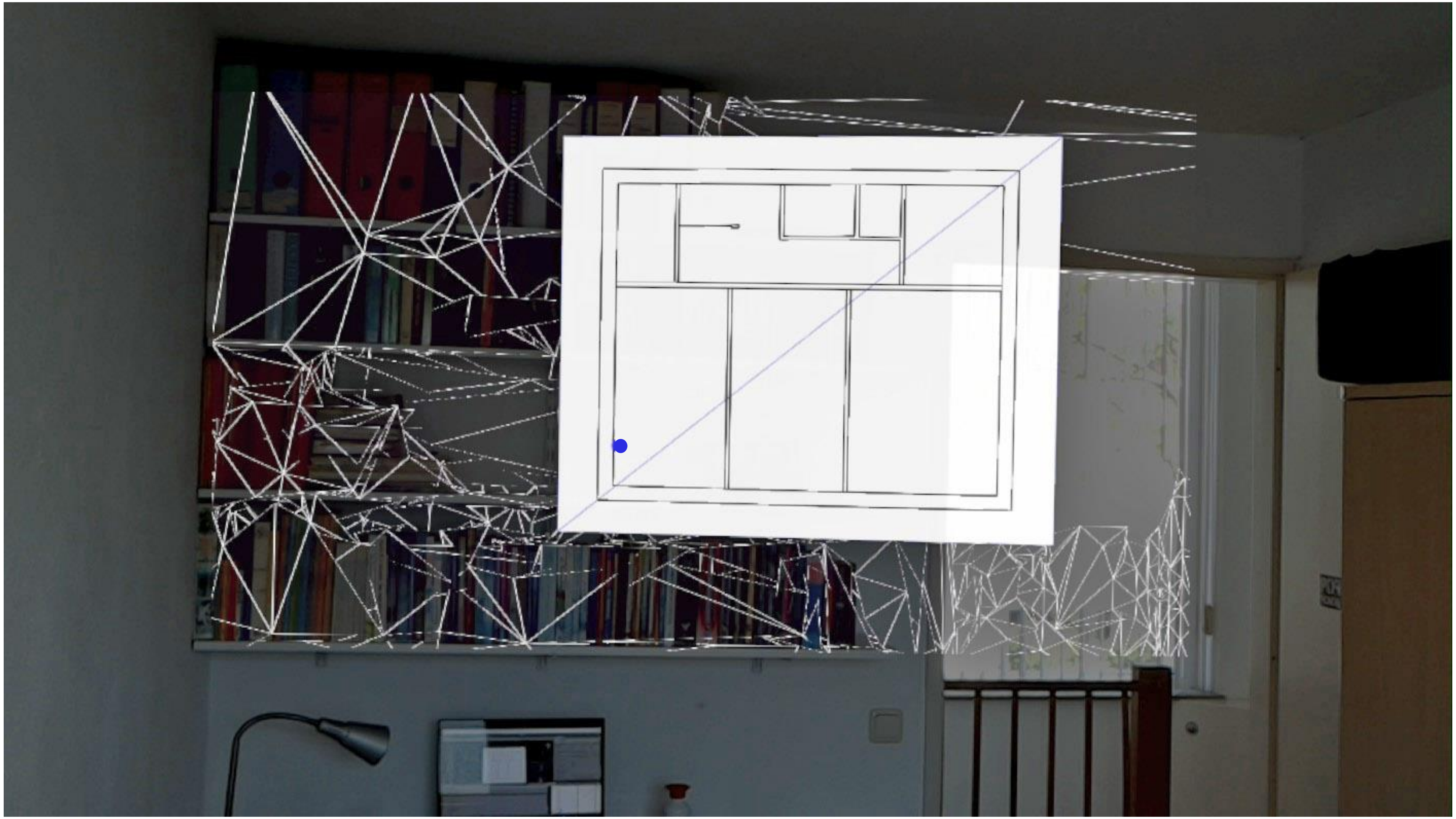
USER INTERFACE

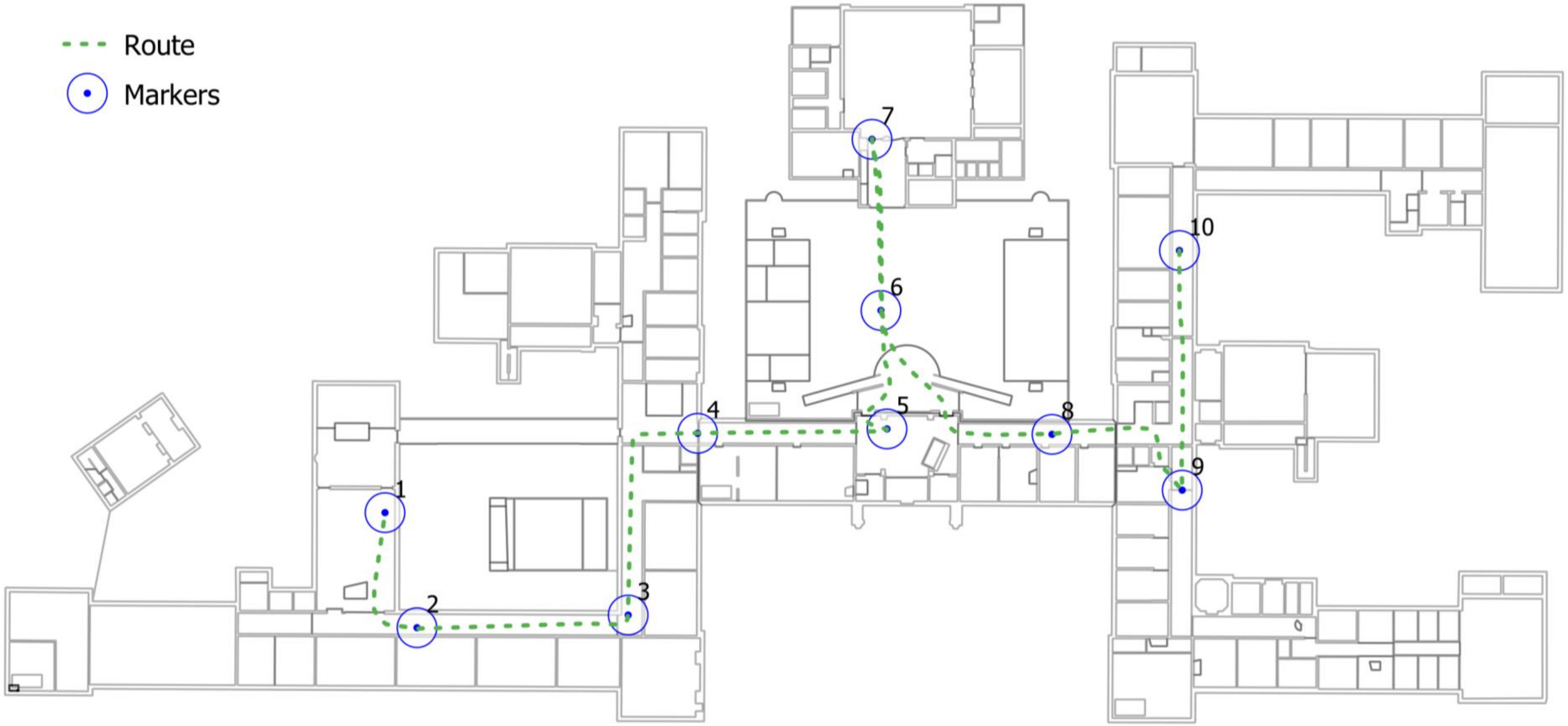
SHOW POSITION ON FLOOR PLAN

ADJUST POSITION WHEN USER MOVES

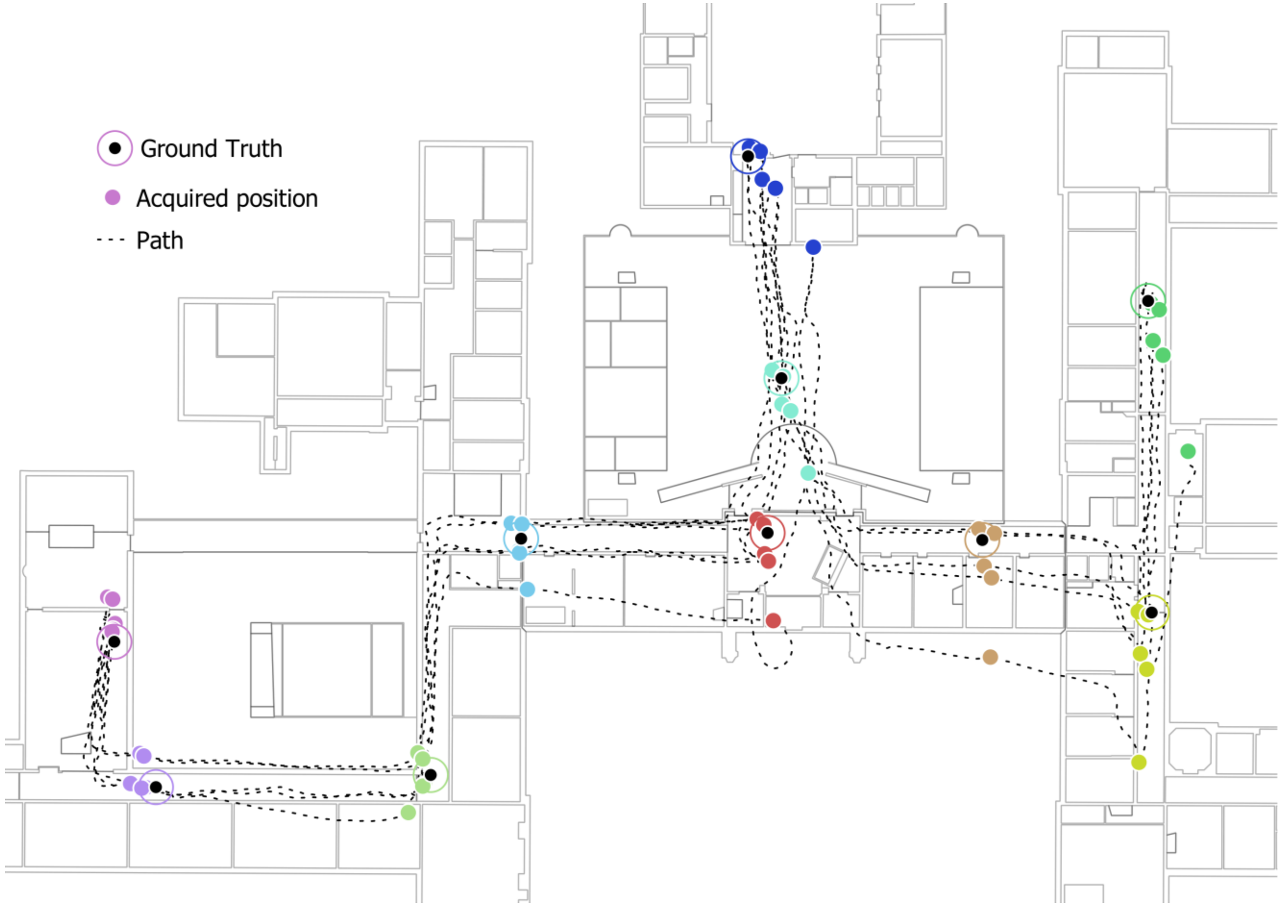
CORRECT ERRORS IN POSITION

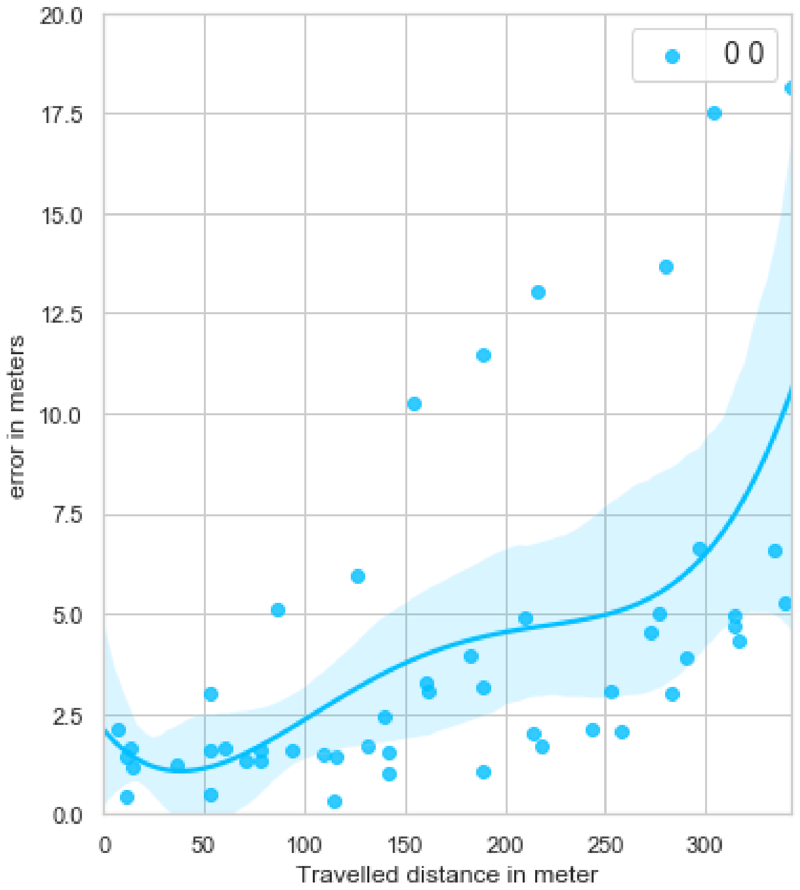

Building
Select wall...
Show floorplan
Rotate 180 degrees
Start Algorithm
Quit



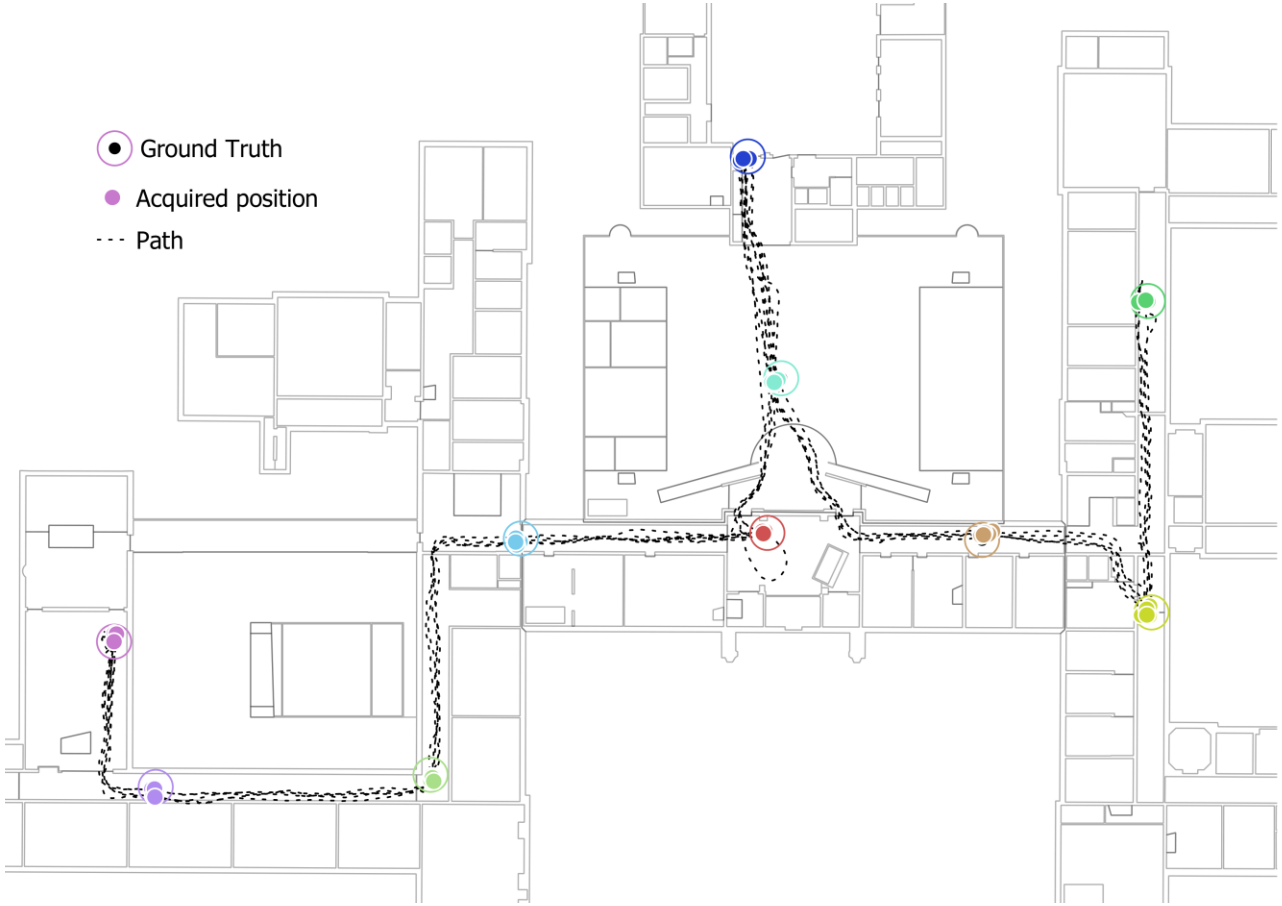

Route
Markers
1
2
3
4
5
6
7
8
9
10

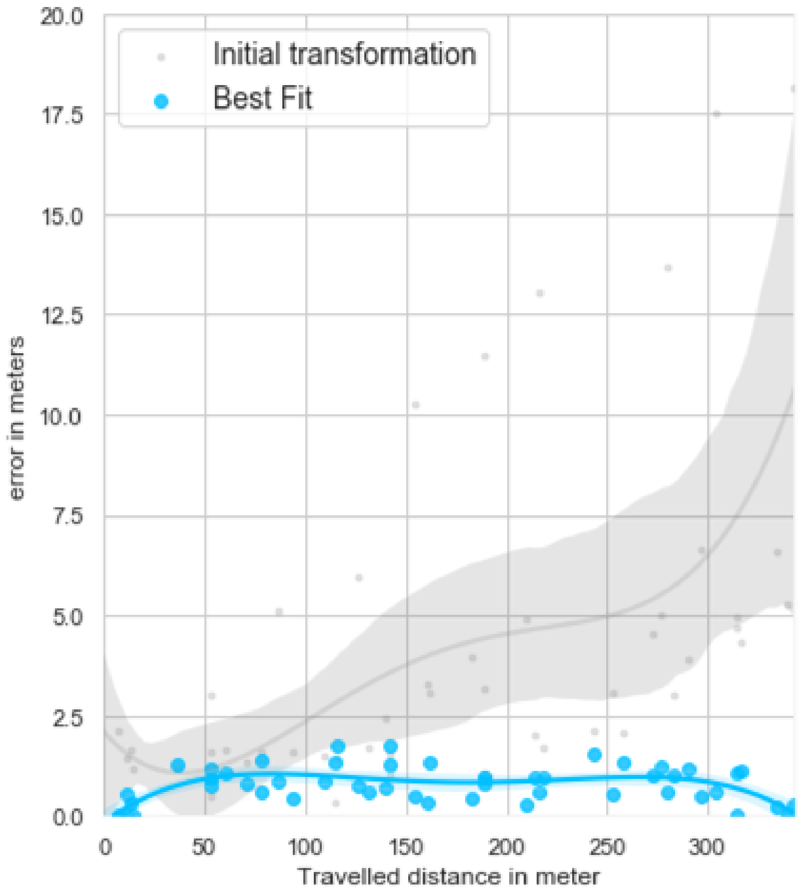
Errors after initial transformation





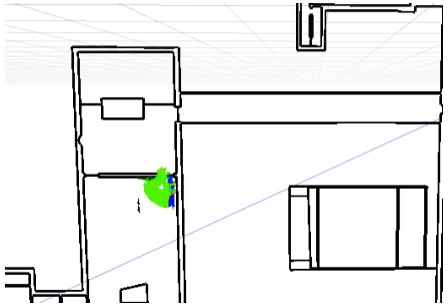
Errors of best fit using SVD (MH drift)





Benchmark

Small mesh



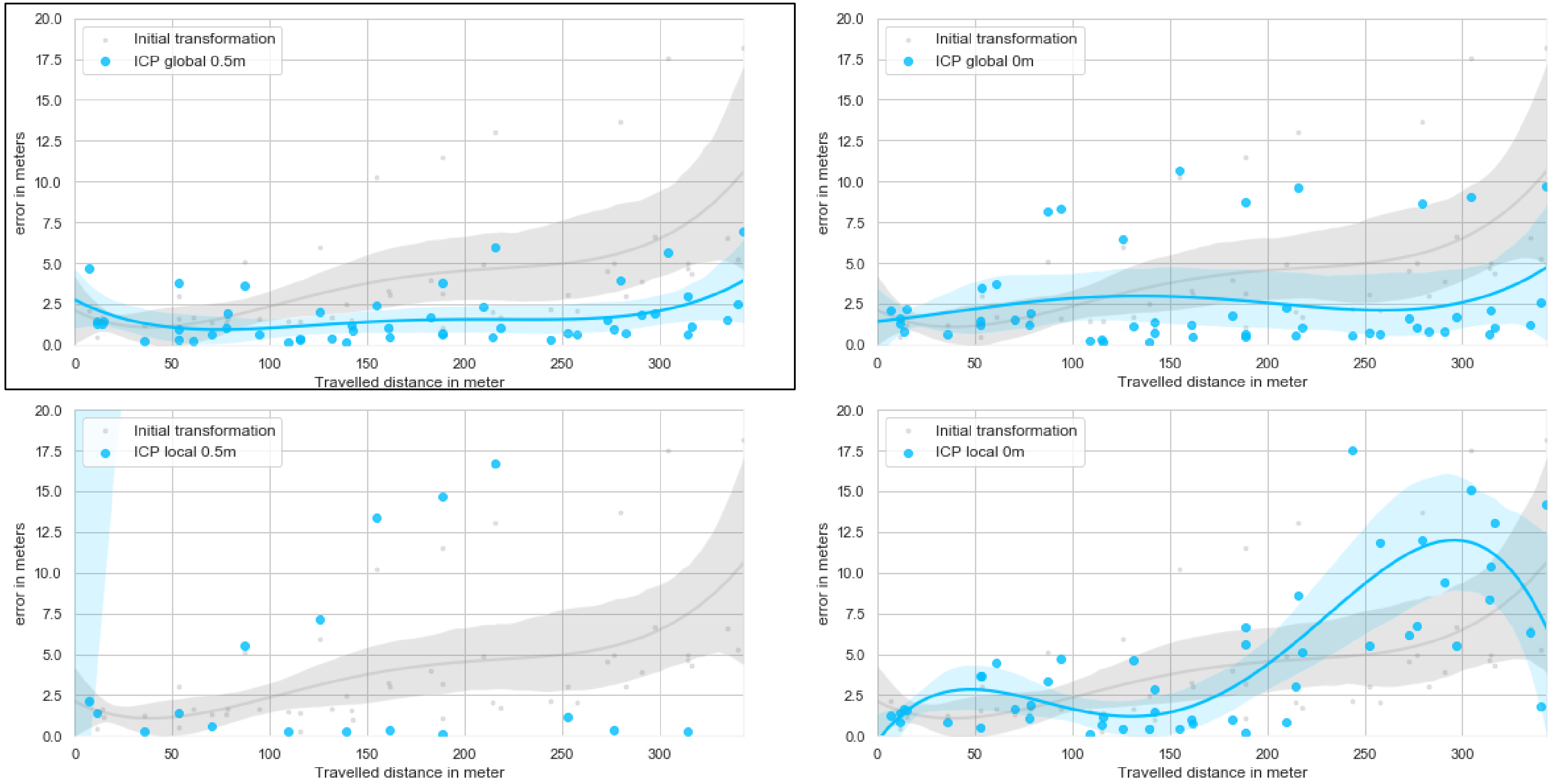
Medium mesh

Large mesh

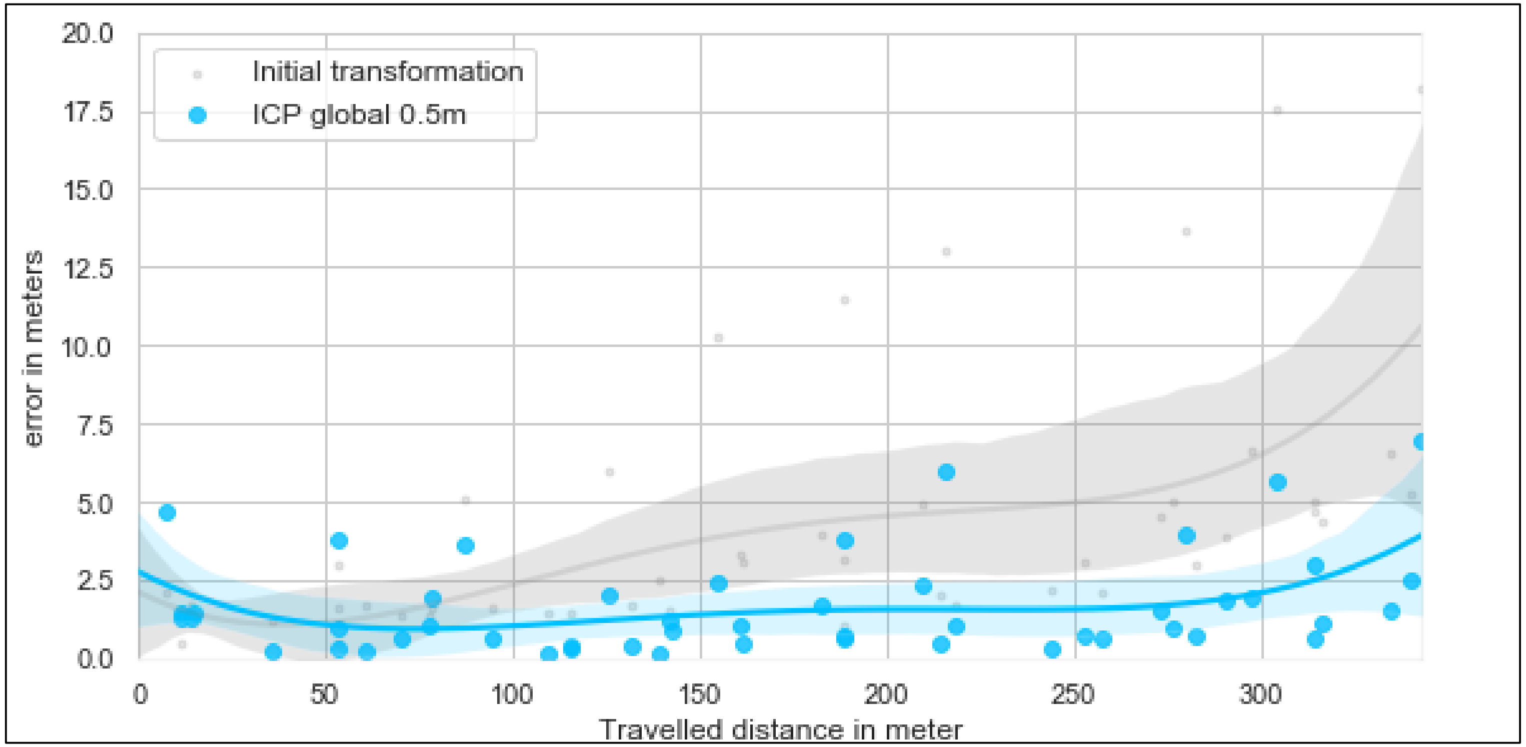
Computation times in seconds per algorithm per mesh with N points

N points	ICP			Instantaneous Kinematics			Hough Transform		
	Small	Medium	Large	Small	Medium	Large	Small	Medium	Large
100	1.24	1.3	9.79	4.3	2.8	2.26	0.51	1.3	3.37
1000	2.611	4.8	9.63	3.55	3.9	4.81	3.166	4.6	5.54
10000	14.96	17.25	36.75	16.55	11.9	22.47	24.8	26.15	29.33
100000	126.33	214.5	408.46	171.72	166.28	202.63	244.1	264	281.7

Results using ICP



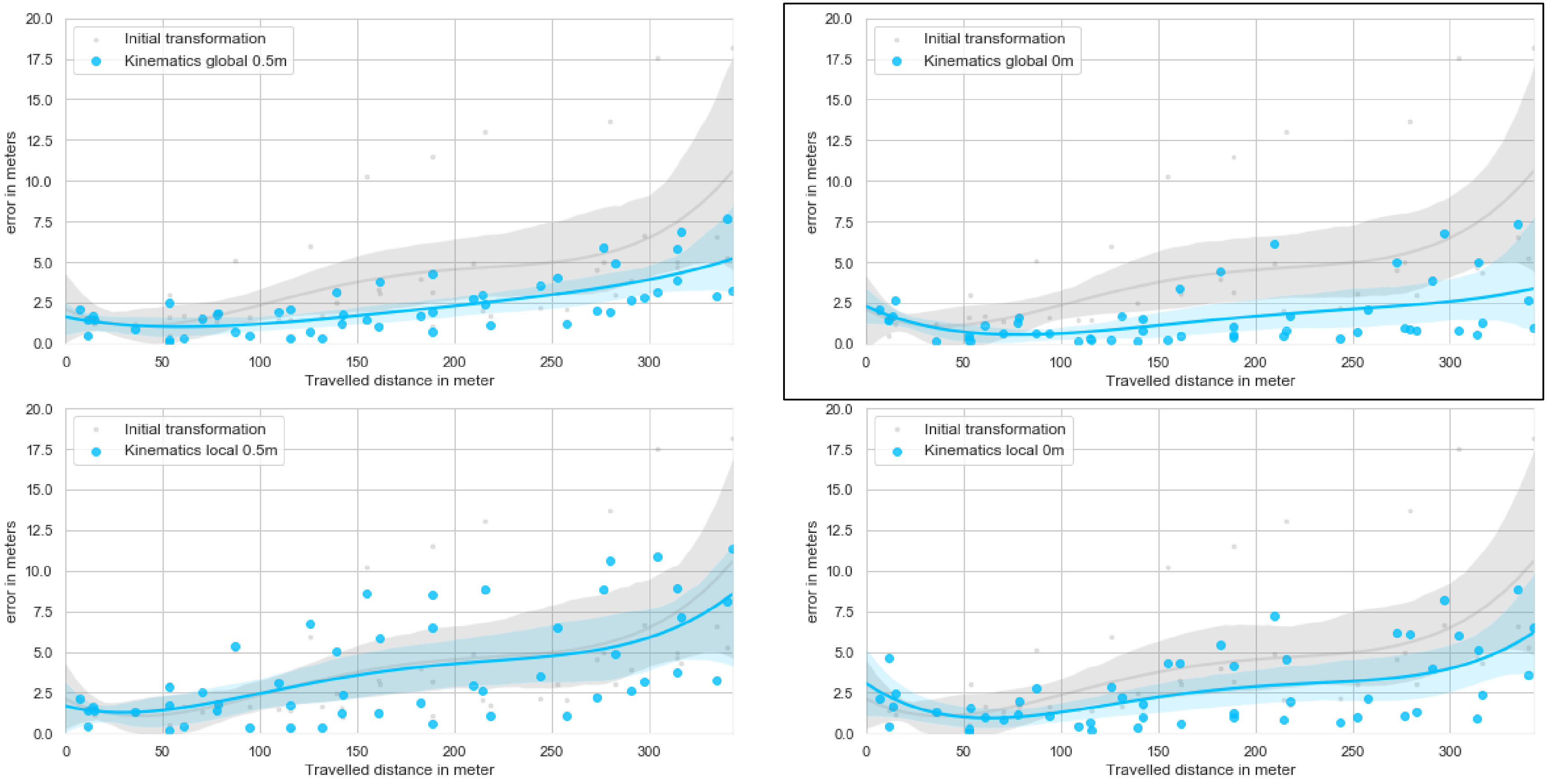
Results using ICP



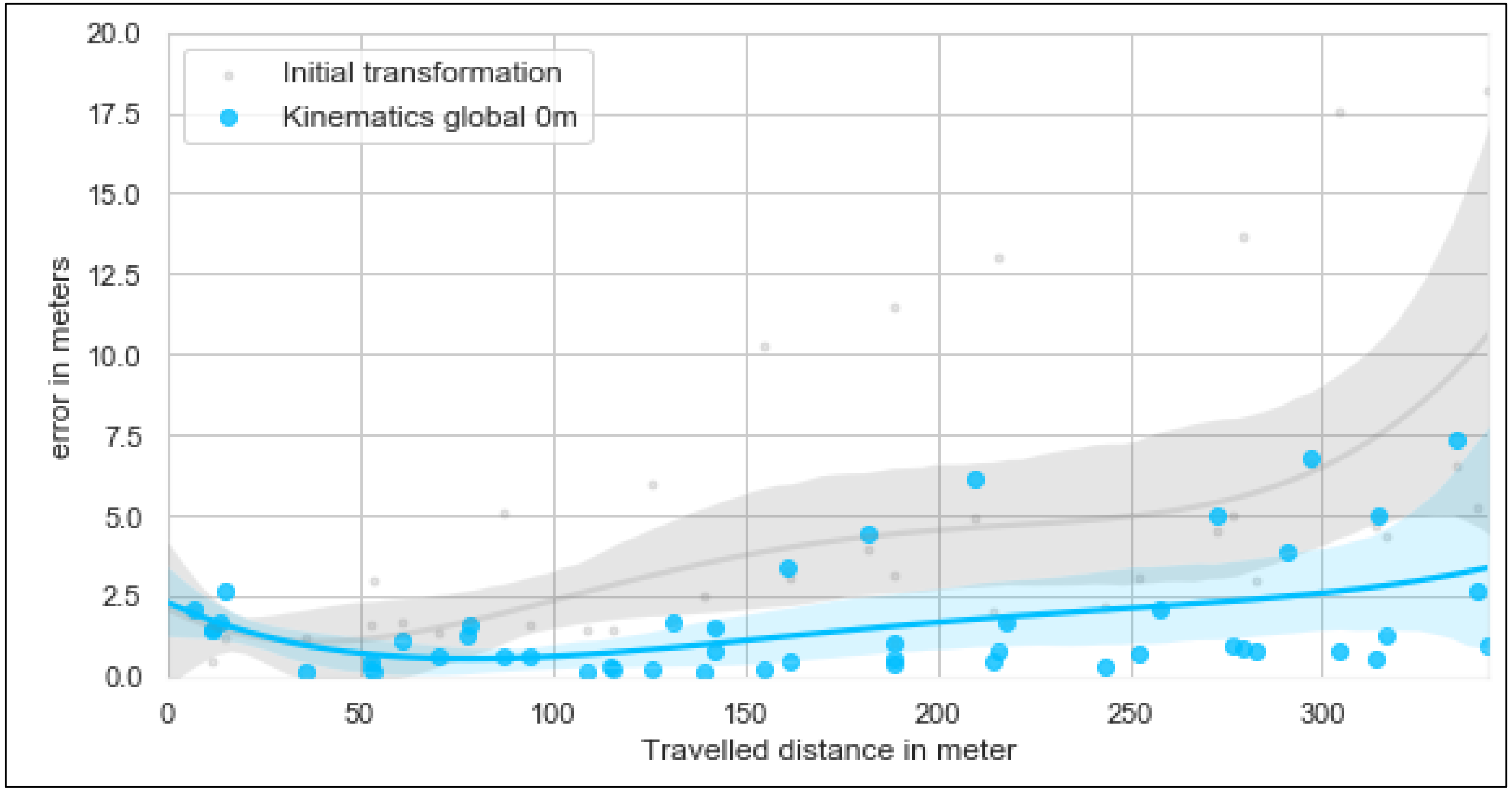
- Use of all meshes
- With use of 50cm buffer
- Max error of 7m

Computation times:
18s (sd=18) per registration

Results using Instantaneous Kinematics



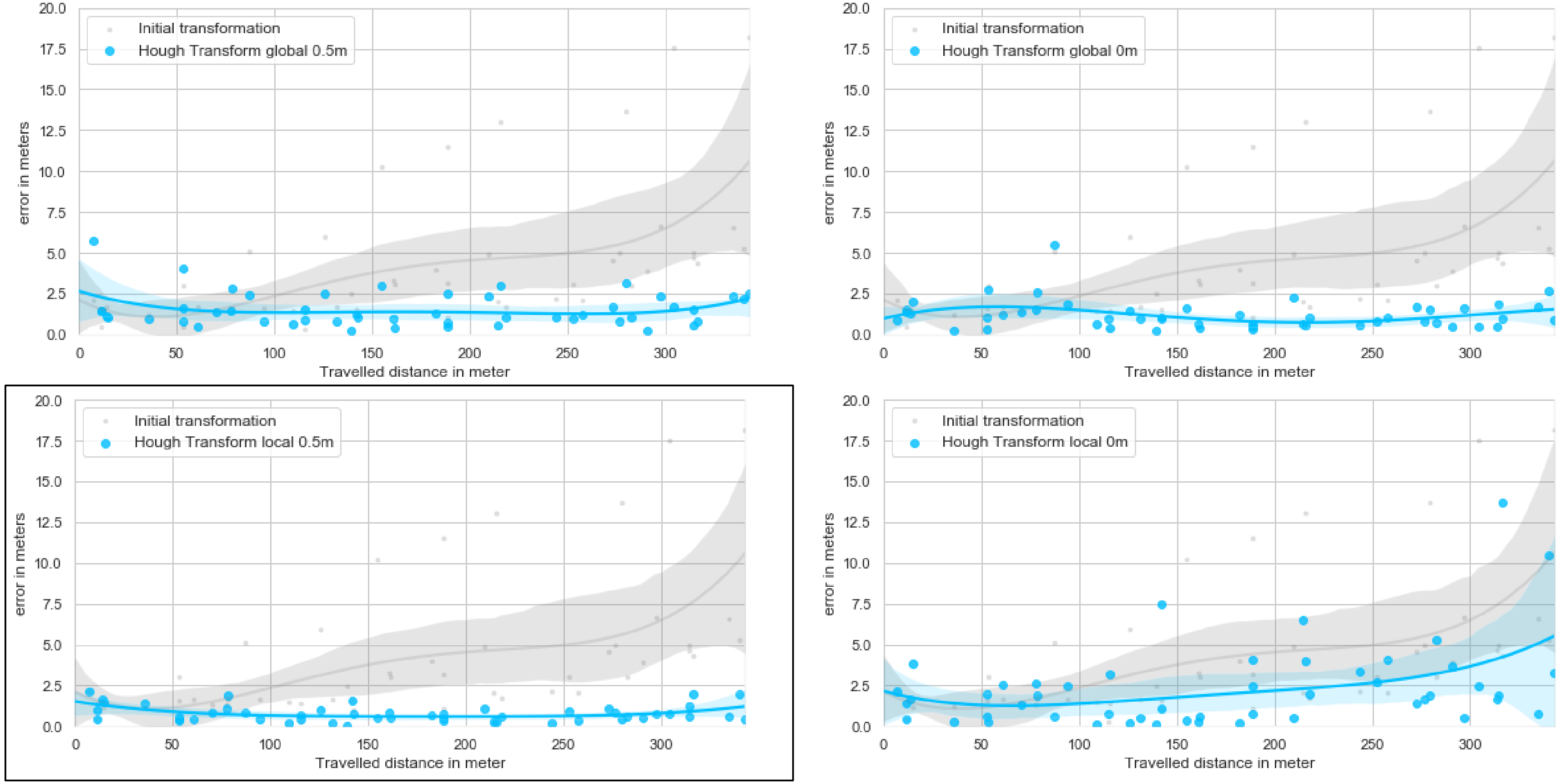
Results using Instantaneous Kinematics



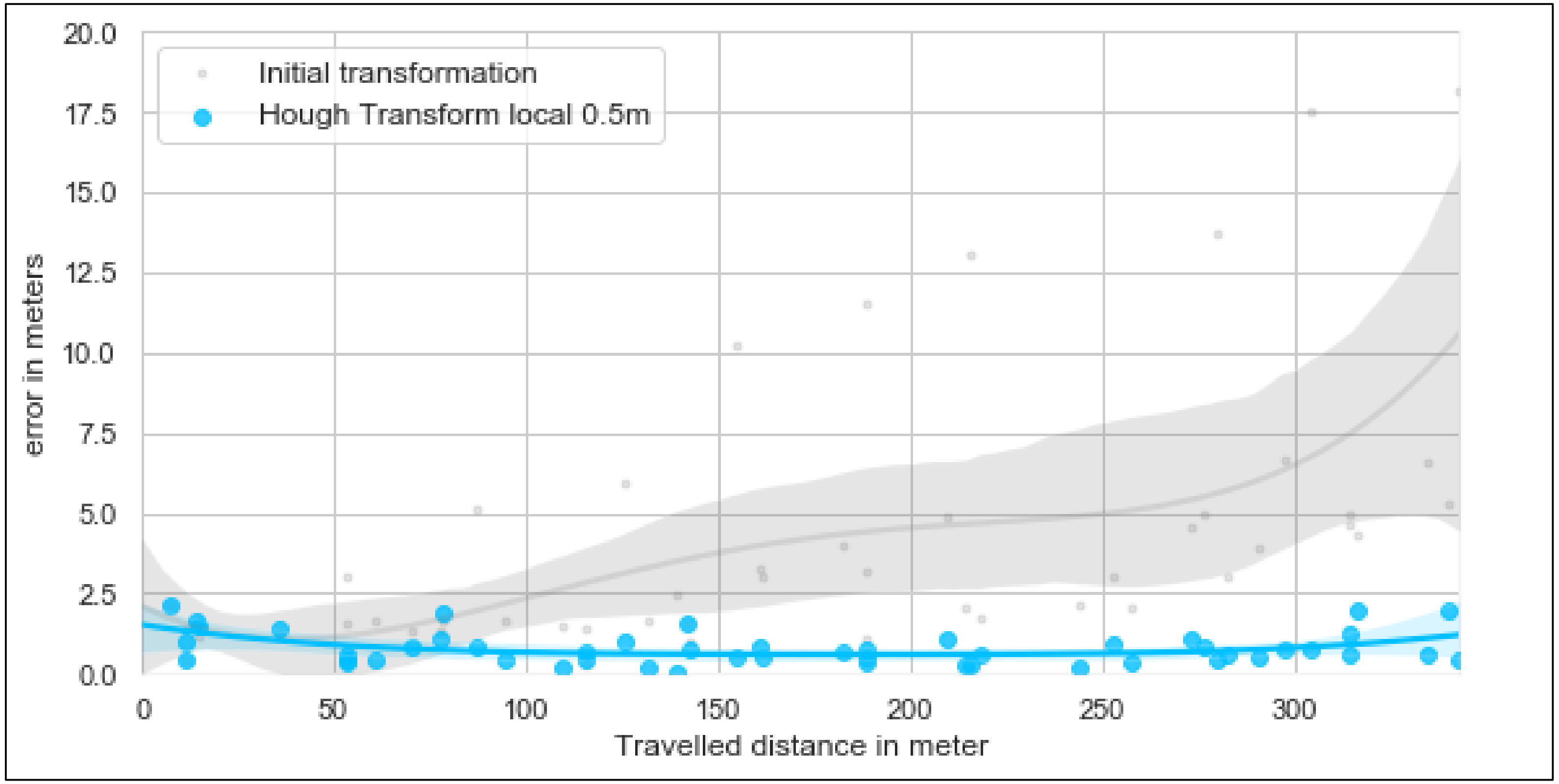
- Use of all meshes
- no buffer
- Average error of 1.6m
- Max error of 7.3m

Computation times:
31s (sd=12.5) per registration

Results using Hough Transform



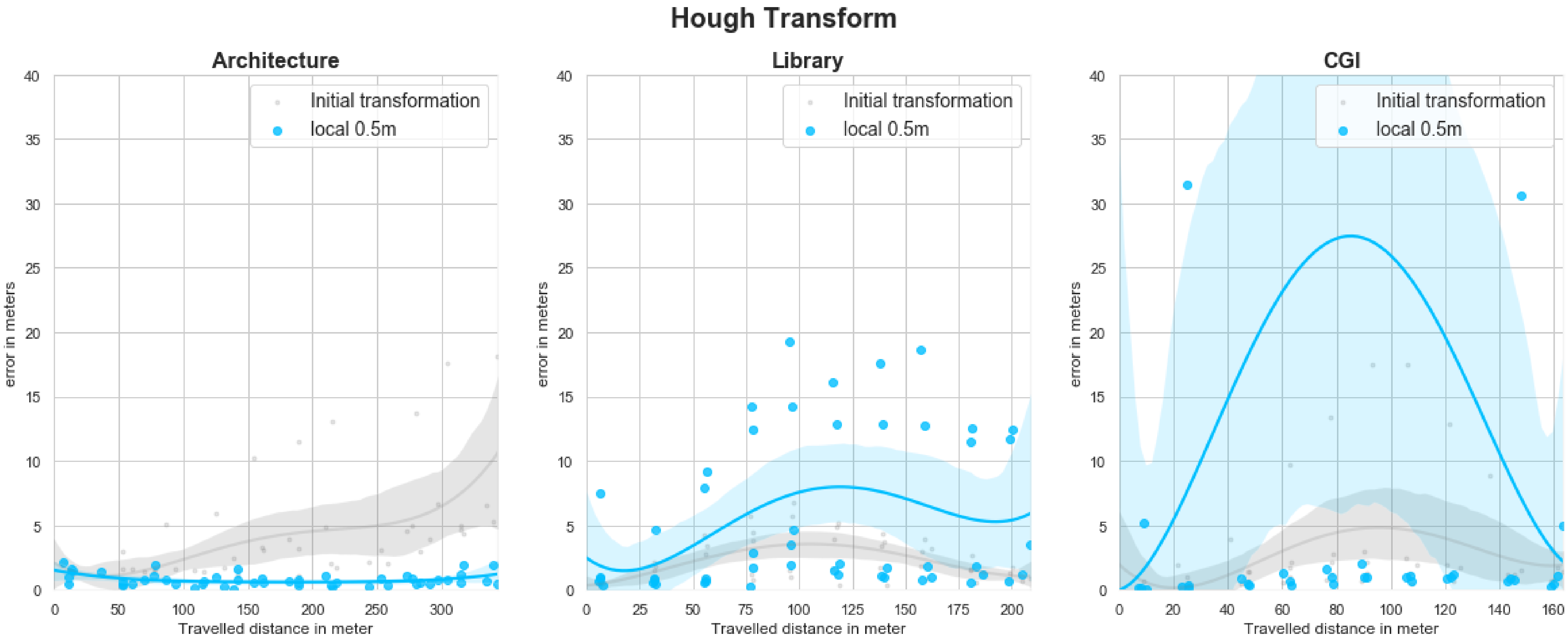
Results using Hough Transform

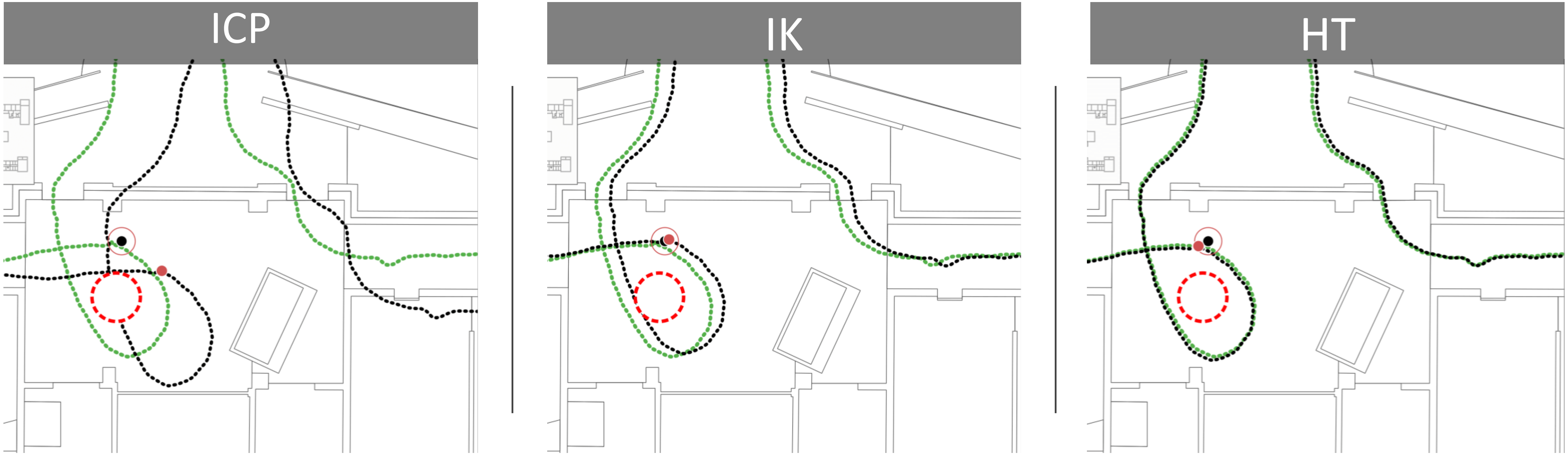


- Use of meshes in proximity
- 50cm buffer
- Average error of 80cm
- Max error of 2.13m

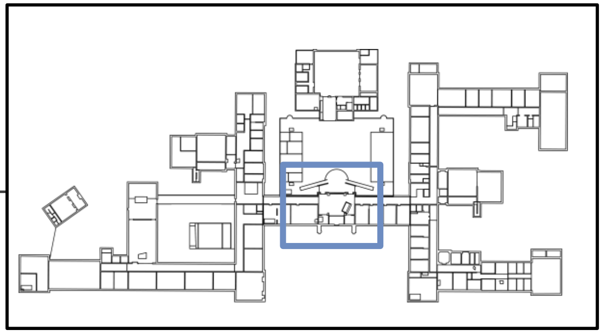
Computation times:
6.74s (sd=3.8) per registration

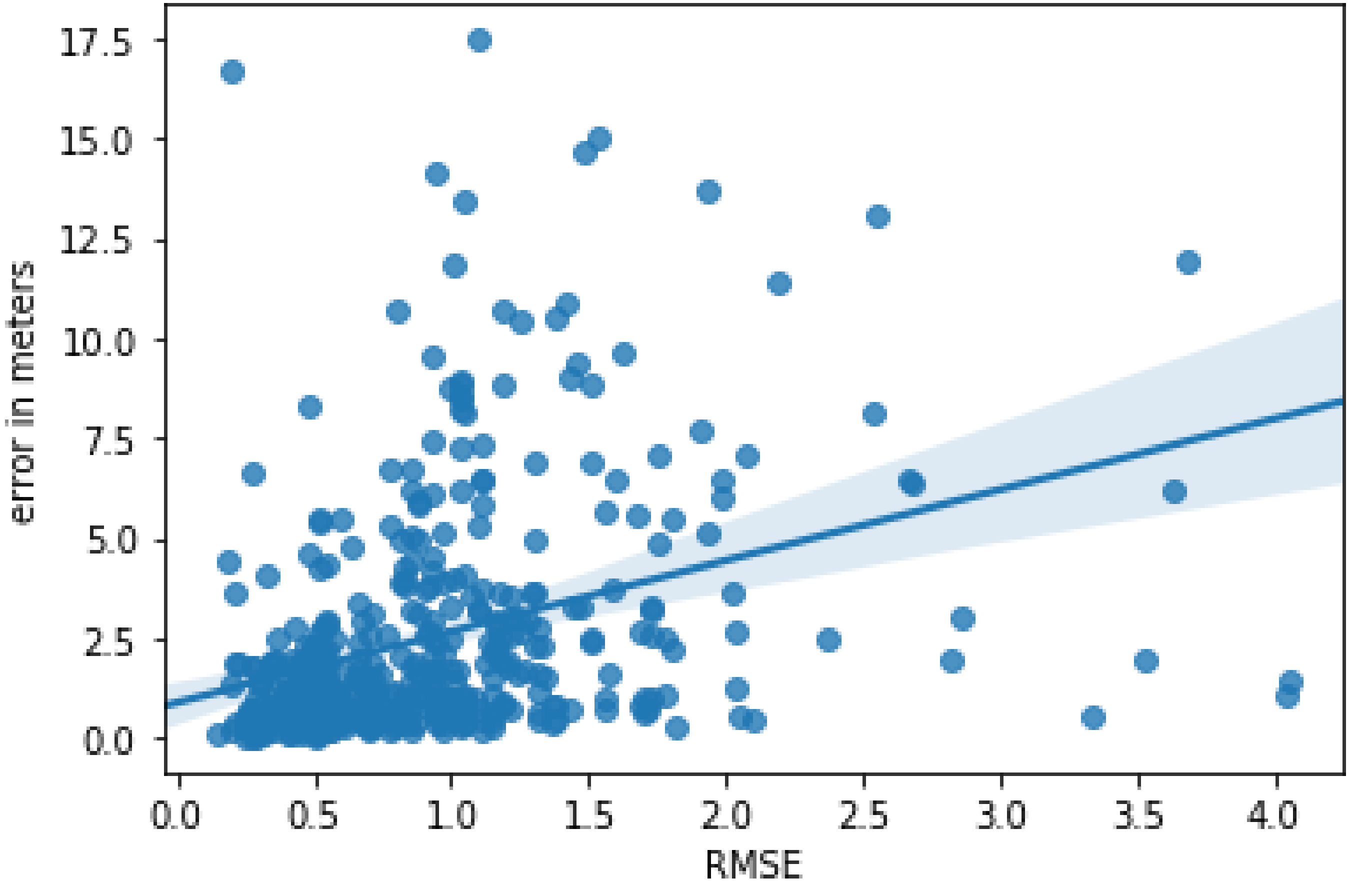
Validation tests





Robustness – walls in scanned mesh





RMSE as estimation for accuracy

Significant correlation (p<0.001)

Only 12% explanation of variance

Conclusions

(1) **Iterative Closest Points**, **Instantaneous Kinematics** and **Hough Transform** have been selected as feasible algorithms.

(2) **Hough Transform** gave most accurate results and is fastest.

(3) 80% of experiments <5 meter, 15 second computation time met.

(4) Bad scan quality and artefacts are likely to affect accuracy most.

Research question

"How can the Microsoft Hololens improve indoor positioning, using the on-the-fly produced mesh and an existing floor plan"

Requirements for positioning in ER

ACCURACY
<1M

CONSTANT
AVAILABILITY

UNCERTAINTY
ESTIMATION

NO
PRE-INSTALLATION

Conclusions

Augmented Reality is a viable technology for indoor positioning without any pre-installations.

Not all requirements for the case of ER are met.

Better noise filters and outlier detection.

Discussion

Method can be generalised to 3D and any AR device.

Limited implications for Emergency Response:

- Fragile device
- No vertical accuracy
- Initial transformation requires manual work
- Limited support for floor plan formats

Future work

- Indoor navigation using AR

- 3D models

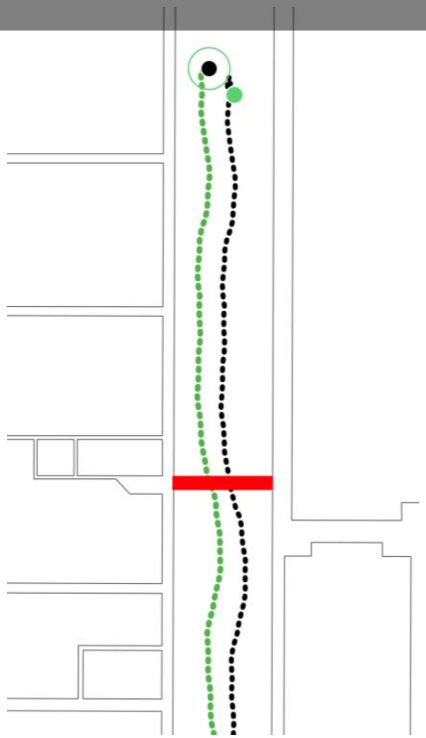
- Country-wide system

- Situational awareness

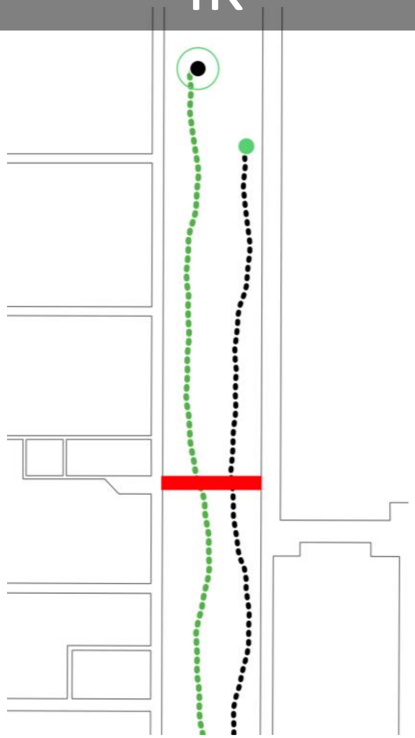
Questions?

Robustness – doors

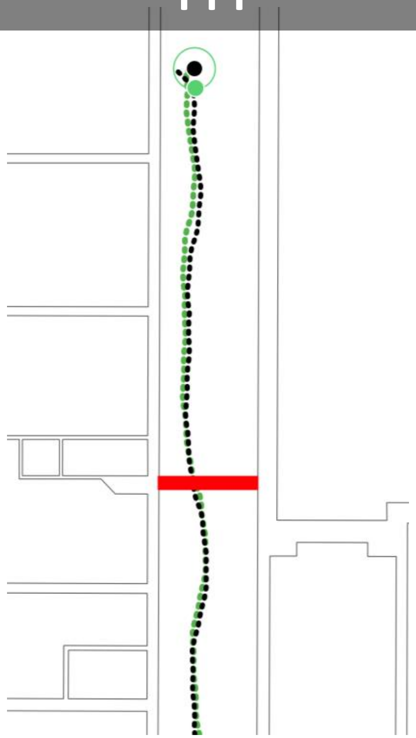
ICP

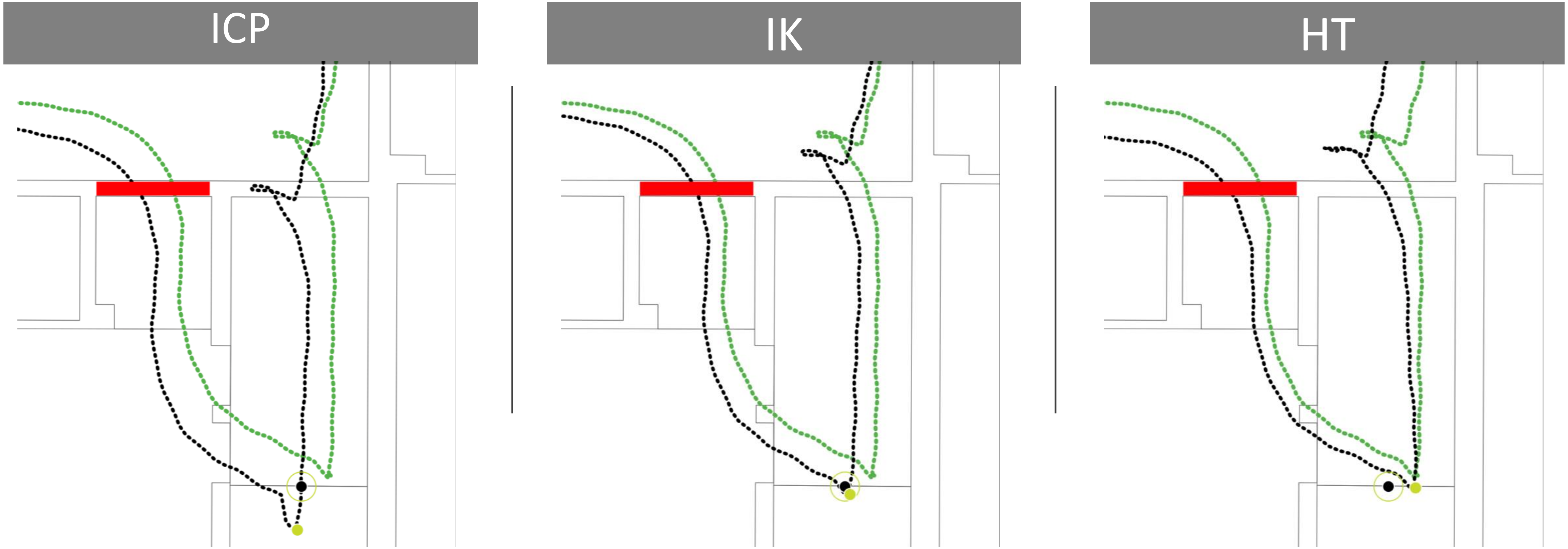


IK



HT

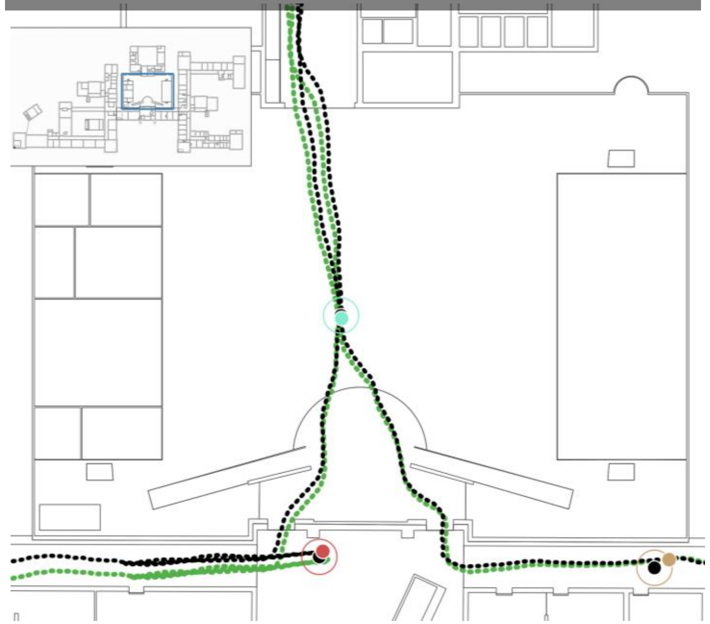




Robustness – walls in floor plan

ICP

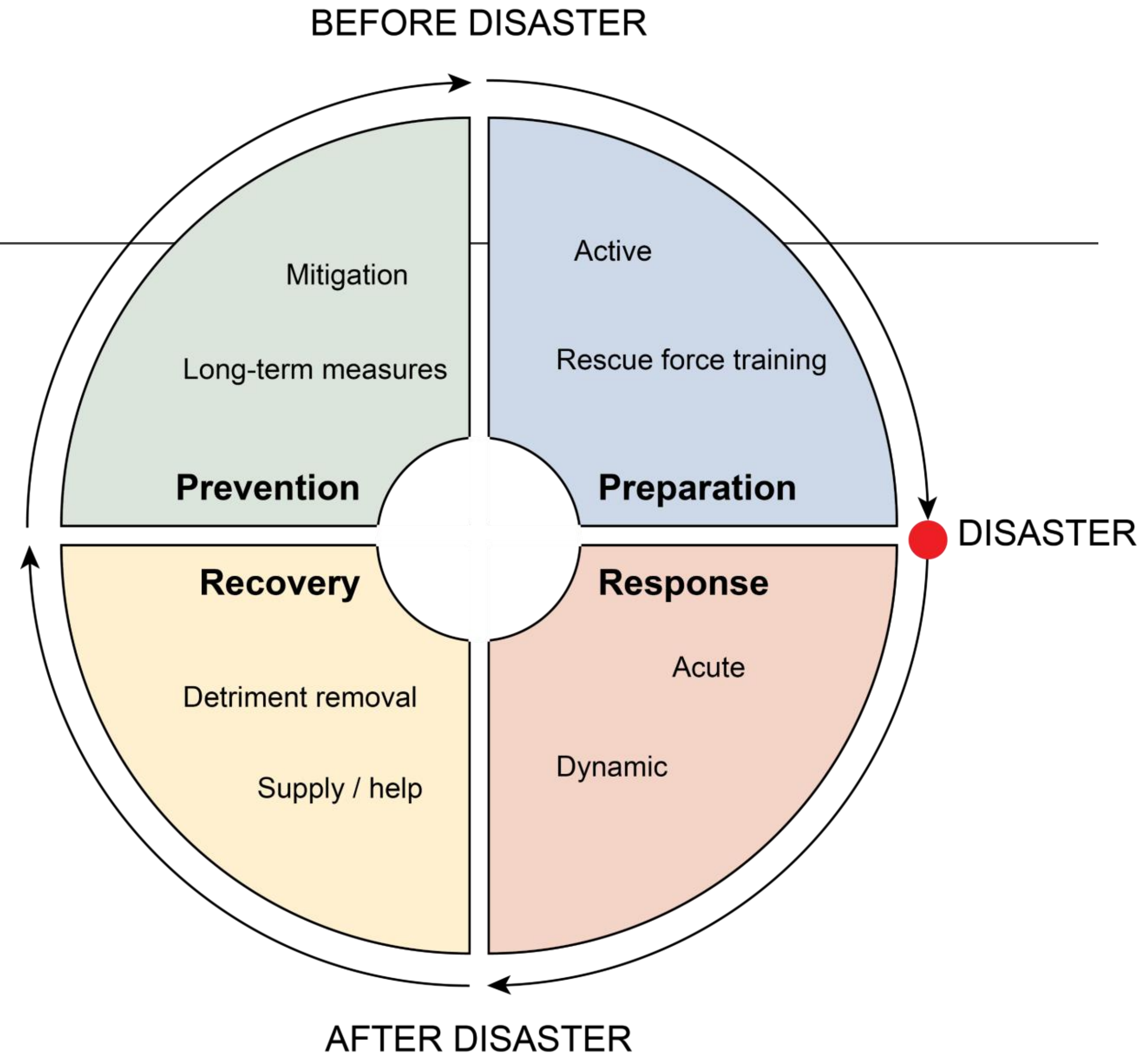
IK

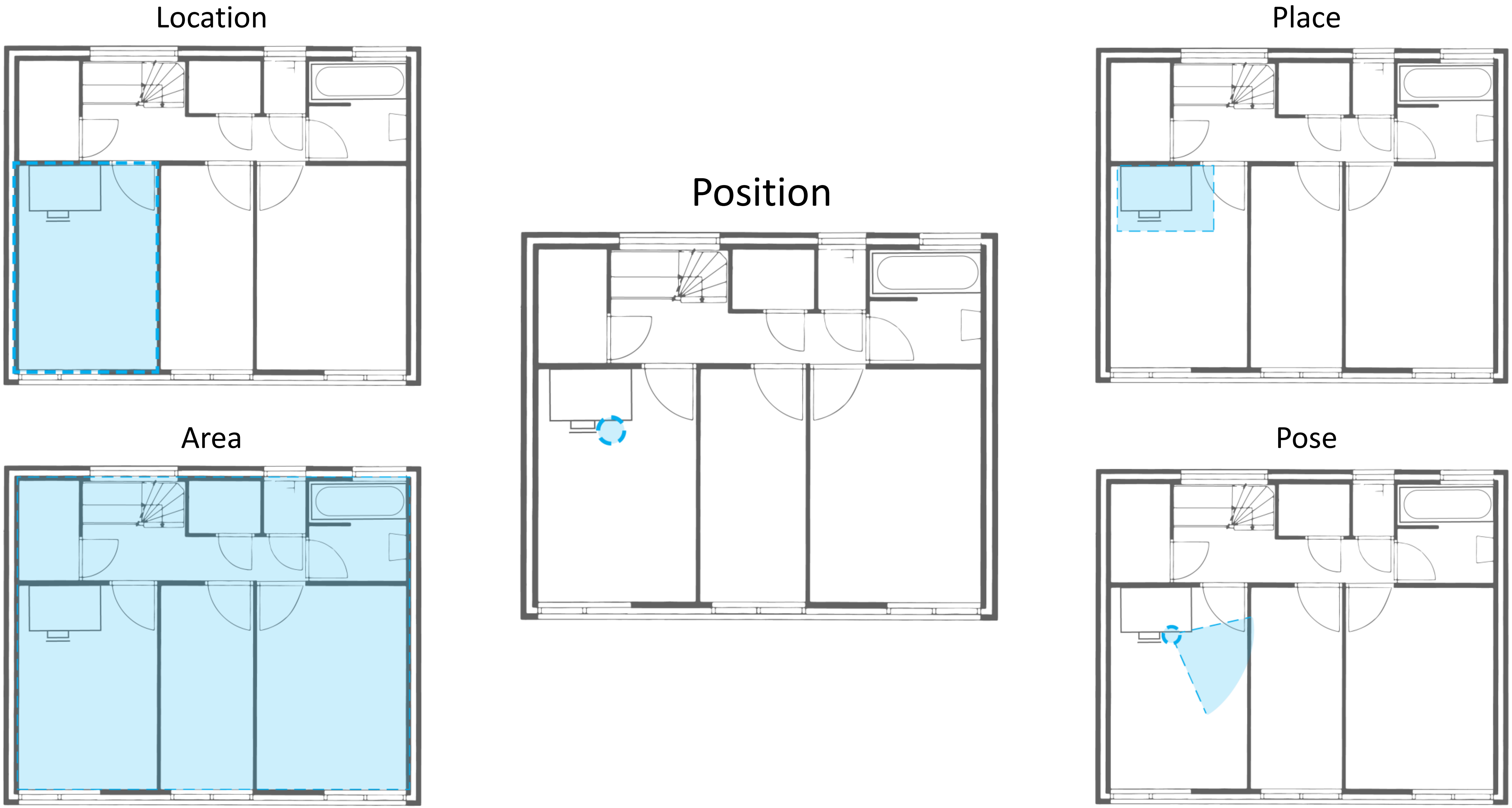


HT

Robustness – large space

Emergency Response




Location
Place
Position
Area
Pose

16 requirements for indoor positioning in ER

1. Location accuracy in the horizontal plane of no greater error than one meter.

2. Stringent location accuracy in the vertical plane of no greater error than two meters

3. Constant accessibility for those who need the positioning data.

4. Physical robustness so that the system will operate reliably even under harsh conditions

5. Encrypted voice communications and data transfer.

6. Integrity monitoring (**uncertainty estimation** + detection of electronic attacks)

7. Positioning data to be compatible to and integrated with other information

8. Real-time map-building capability in the form of simultaneous localization and mapping (SLAM)

9. The system should not be bulky

10. Weight less than 1 kg

11. Energy-efficient system

12. Presentation of positioning data to be intuitive and easy to understand

13. A modular system

14. No pre-installation

15. In any armed operation, the visualization system should present heading to own troops and in particular the heading of the weapon. Data for distance and direction to targets and threats should also be presented.

16. System costs below €1000

IK proof

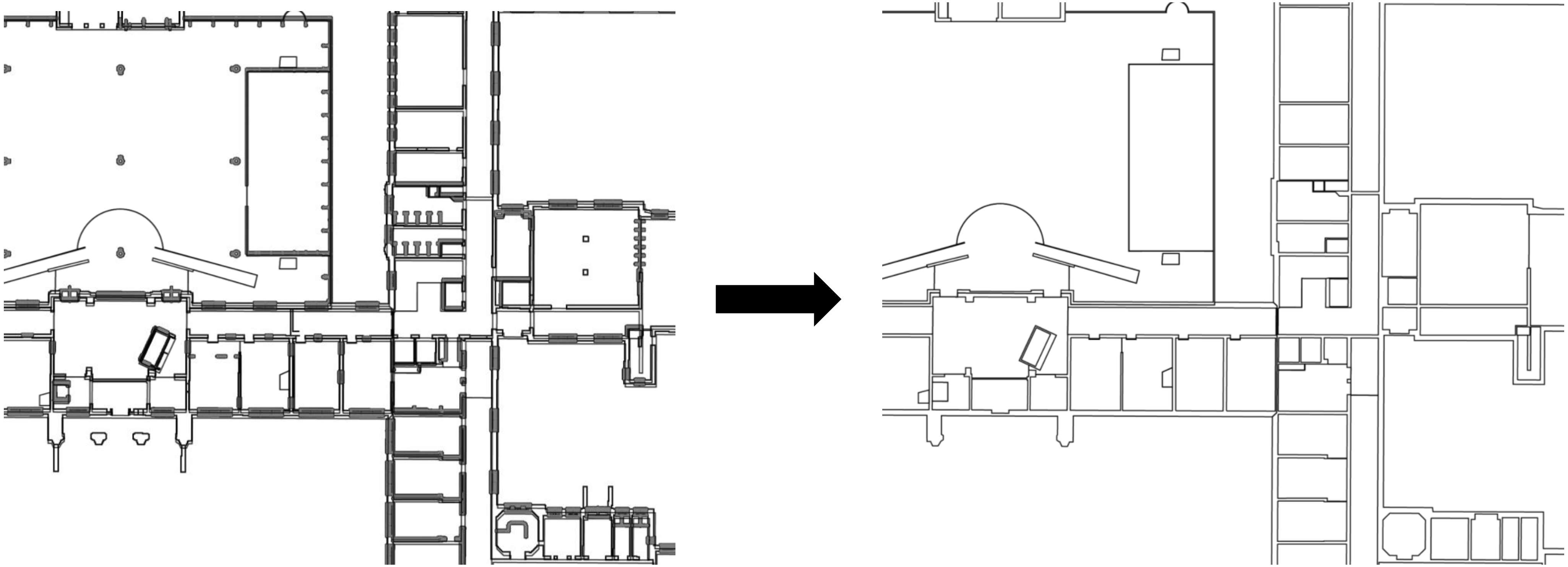
$$F(C) := F(\mathbf{c}, \bar{\mathbf{c}}) = \sum_i (d_i + \mathbf{n}_i \cdot (\bar{\mathbf{c}} + \mathbf{c} \times \mathbf{x}_i))^2,$$

$$d_i + \mathbf{n}_i \cdot \bar{\mathbf{c}} + (\mathbf{x}_i \times \mathbf{n}_i) \cdot \mathbf{c} \;=\; d_i + (\mathbf{x}_i \times \mathbf{n}_i, \mathbf{n}_i) \begin{pmatrix} \mathbf{c} \\ \bar{\mathbf{c}} \end{pmatrix} \;=\; d_i + A_i C,$$

$$\begin{aligned} F(C) &= \sum_i (d_i + A_i C)^2 \\ &= \sum_i d_i^2 + 2 \sum_i d_i A_i C + \sum_i C^T A_i^T A_i C \\ &= D + 2B^T C + C^T A C \end{aligned}$$

Pottmann, Leopoldseder & Hofer (2004)

Pre-processing floor plan

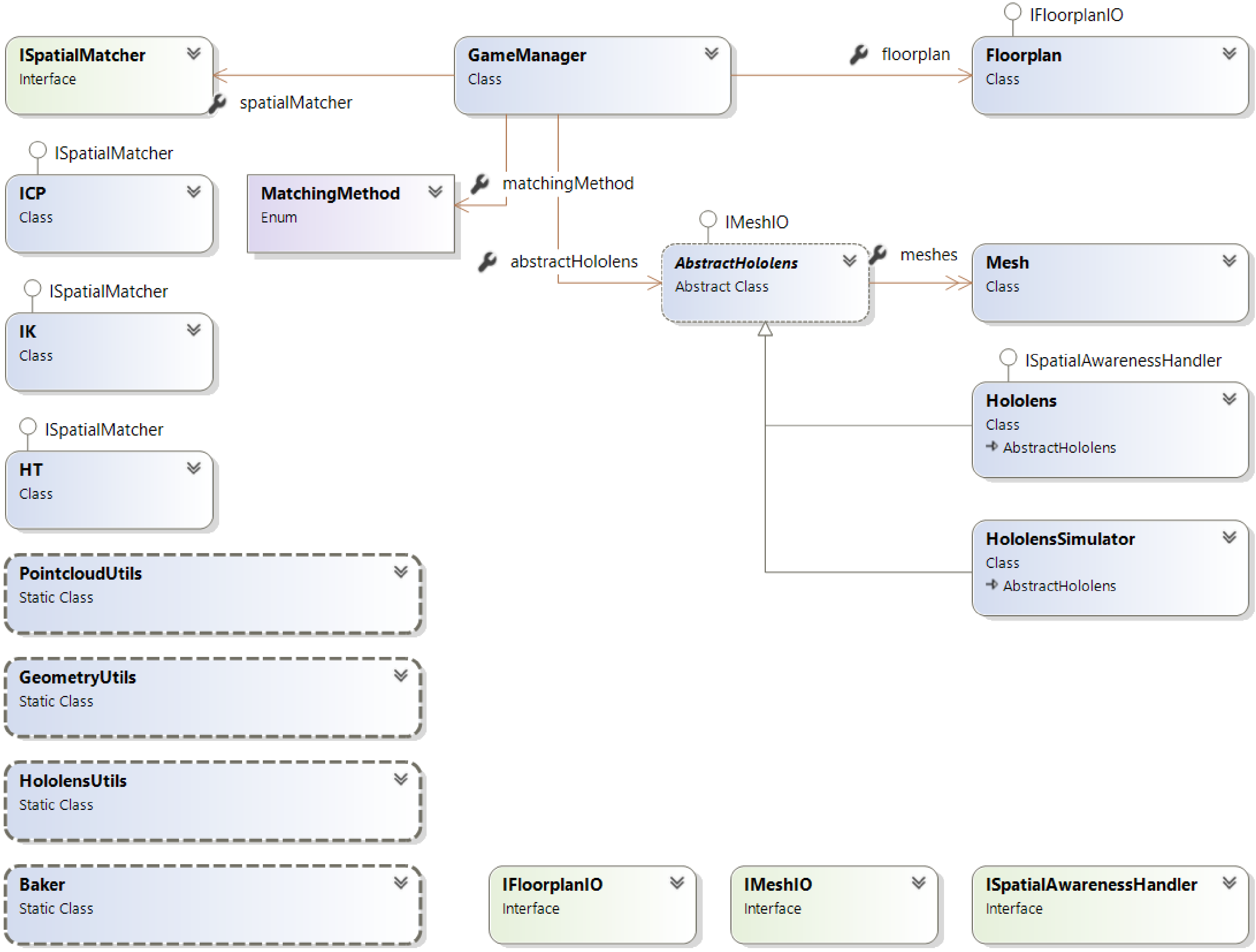


MANUALLY:
ROTATING/TRANSLATING

AUTOMATIC:
DOOR DETECTION

SEMI-AUTOMATIC:
SELECTING WALLS

Initial value


ISpatialMatcher
Interface
GameManager
Class
floorplan
IFloorplanIO
Floorplan
Class
spatialMatcher
ISpatialMatcher
ICP
Class
MatchingMethod
Enum
matchingMethod
IMeshIO
abstractHololens
AbstractHololens
Abstract Class
meshes
Mesh
Class
ISpatialMatcher
IK
Class
ISpatialAwarenessHandler
Hololens
Class
AbstractHololens
ISpatialMatcher
HT
Class
HololensSimulator
Class
AbstractHololens
PointcloudUtils
Static Class
GeometryUtils
Static Class
HololensUtils
Static Class
Baker
Static Class
IFloorplanIO
Interface
IMeshIO
Interface
ISpatialAwarenessHandler
Interface

Instantaneous Kinematics

Distance to be minimized:

$$F(C) \coloneqq F(\bar{\boldsymbol{c}}, \boldsymbol{c}) = \sum_{i=0}^{N} \left(d_i + \boldsymbol{n}_i \cdot (\bar{\boldsymbol{c}} + \boldsymbol{c} \times x_i)\right)^2$$

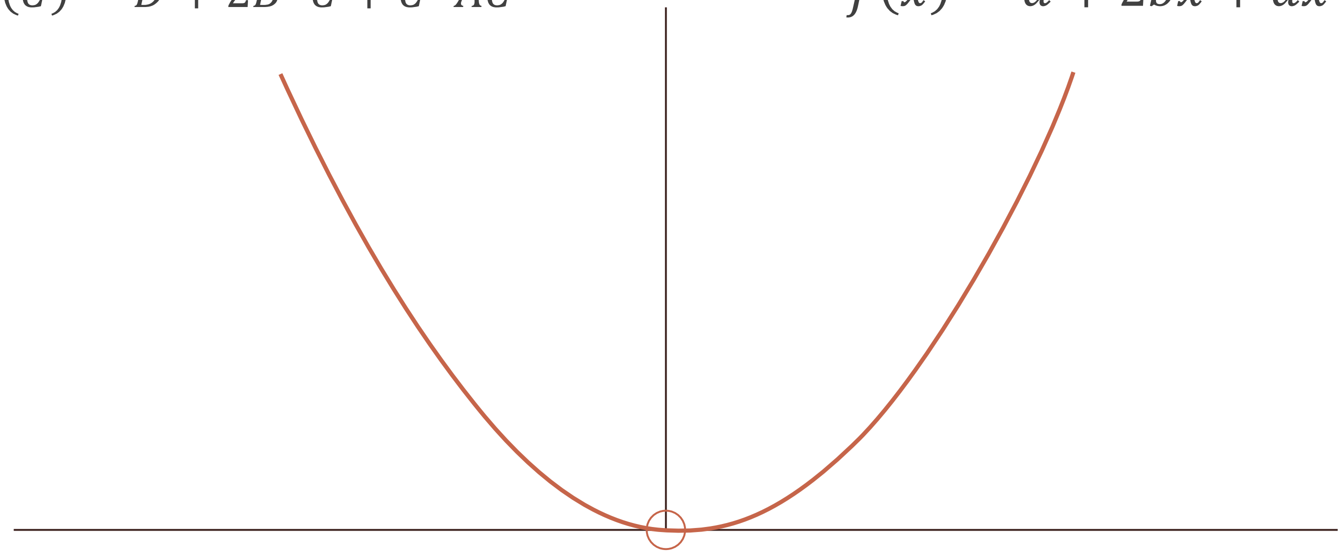
Same function but as matrix function:

$$F(C) = D + 2B^T C + C^T A C$$

Instantaneous Kinematics

$$F(C) = D + 2B^T C + C^T AC$$

$$f(x) = d + 2bx + ax^2$$



$$AC + B = 0$$

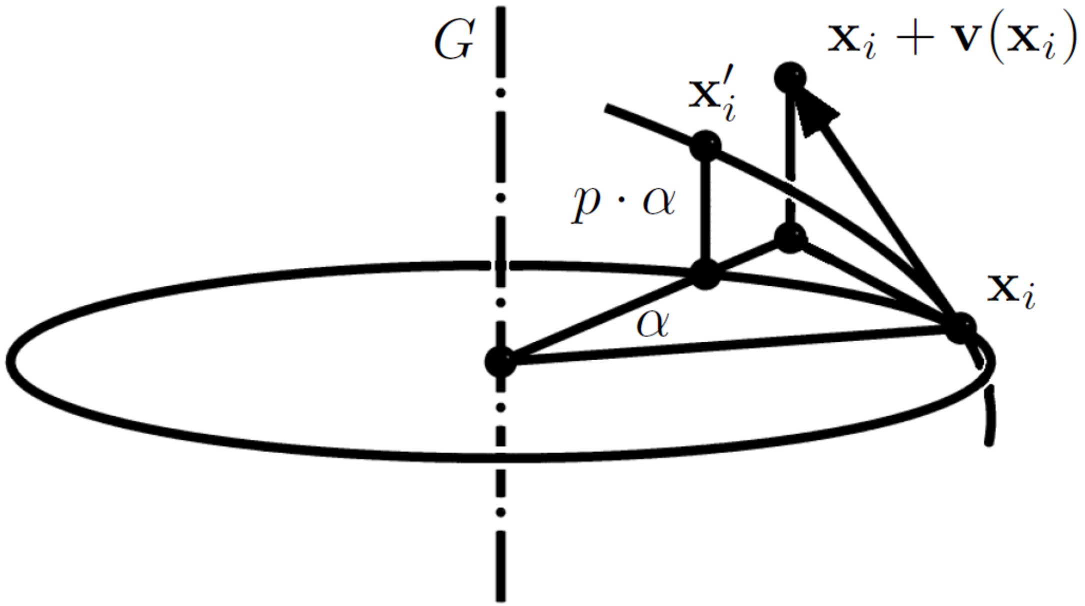
$$ax + b = 0$$

Solving rigidity constraint of IK

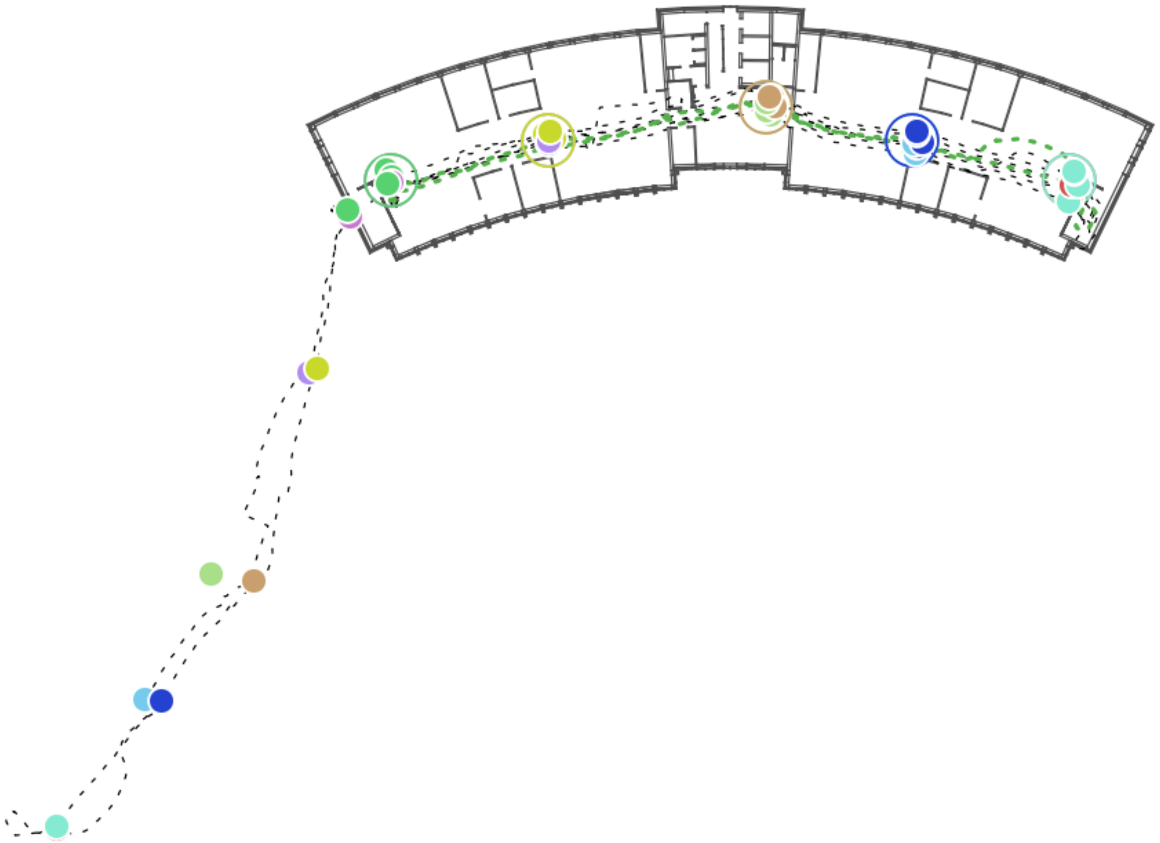
Rotation: $\tan^{-1}\|\boldsymbol{c}\|$

Axis origin:
Axis orientation:

Pitch:



Hough Transform CGI



Hough Transform Library