Andrew and Erna Viterbi
Faculty of Electrical Engineering
Electronics
Computers
Communications

SIPL
Signal and Image Processing Lab

Omek Consortium

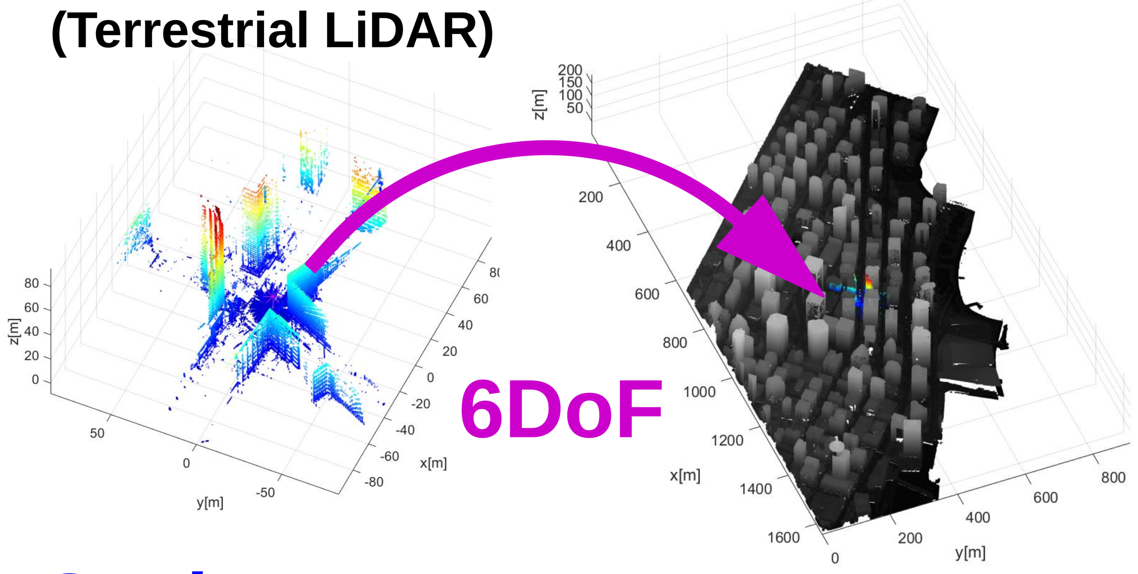
TECHNION
Israel Institute of Technology

# Point Cloud Registration Refinement in an Urban Environment using 2D Edge-Maps

David Avidar, David Malah, and Meir Barzohar

## Motivation

- High level-of-detail **3D modeling of large-scale urban environments** (i.e., cities).
- How? Registration (3D alignment) between:
  - Terrestrial LiDAR Scanning (TLS).
  - Airborne LiDAR Scanning (ALS).
- In a previous work we developed an efficient method to find a **coarse registration** between TLS and ALS using a **viewpoint dictionary** (Avidar et al., ICCV17).




## Goals

- Refinement of a coarse 3D registration between terrestrial and airborne LiDAR scans:
  - Robustness to occlusion and different point density distributions.
  - Computationally efficient algorithm.

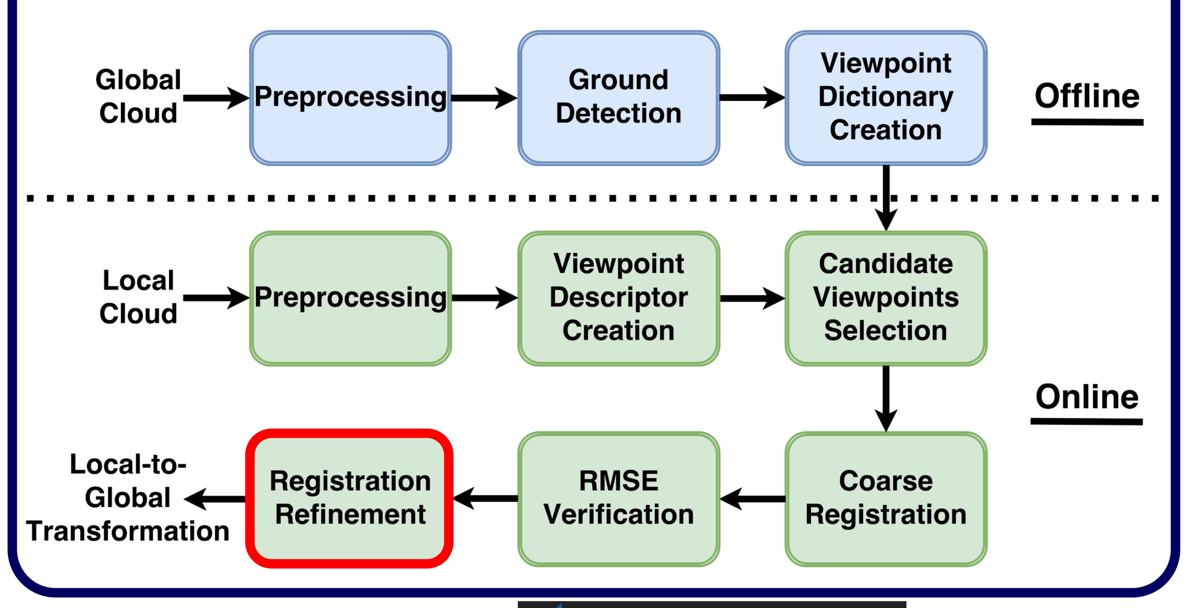
## Challenges

- Airborne vs. terrestrial data characteristics:
  - **Very different point density distributions** (airborne → more points on horizontal surfaces, terrestrial → more points on vertical surfaces).
  - Missing data, **different types of occlusion**.
- **Large-scale** point clouds with **millions of data points**.

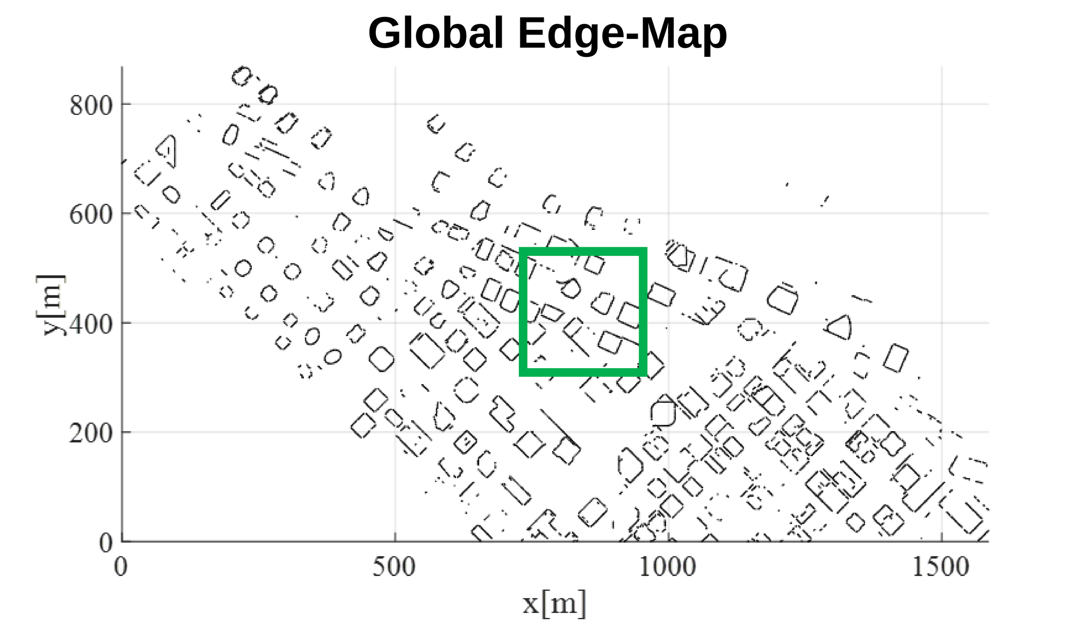
## Local-to-Global Point Cloud Registration using a Dictionary of Viewpoint Descriptors

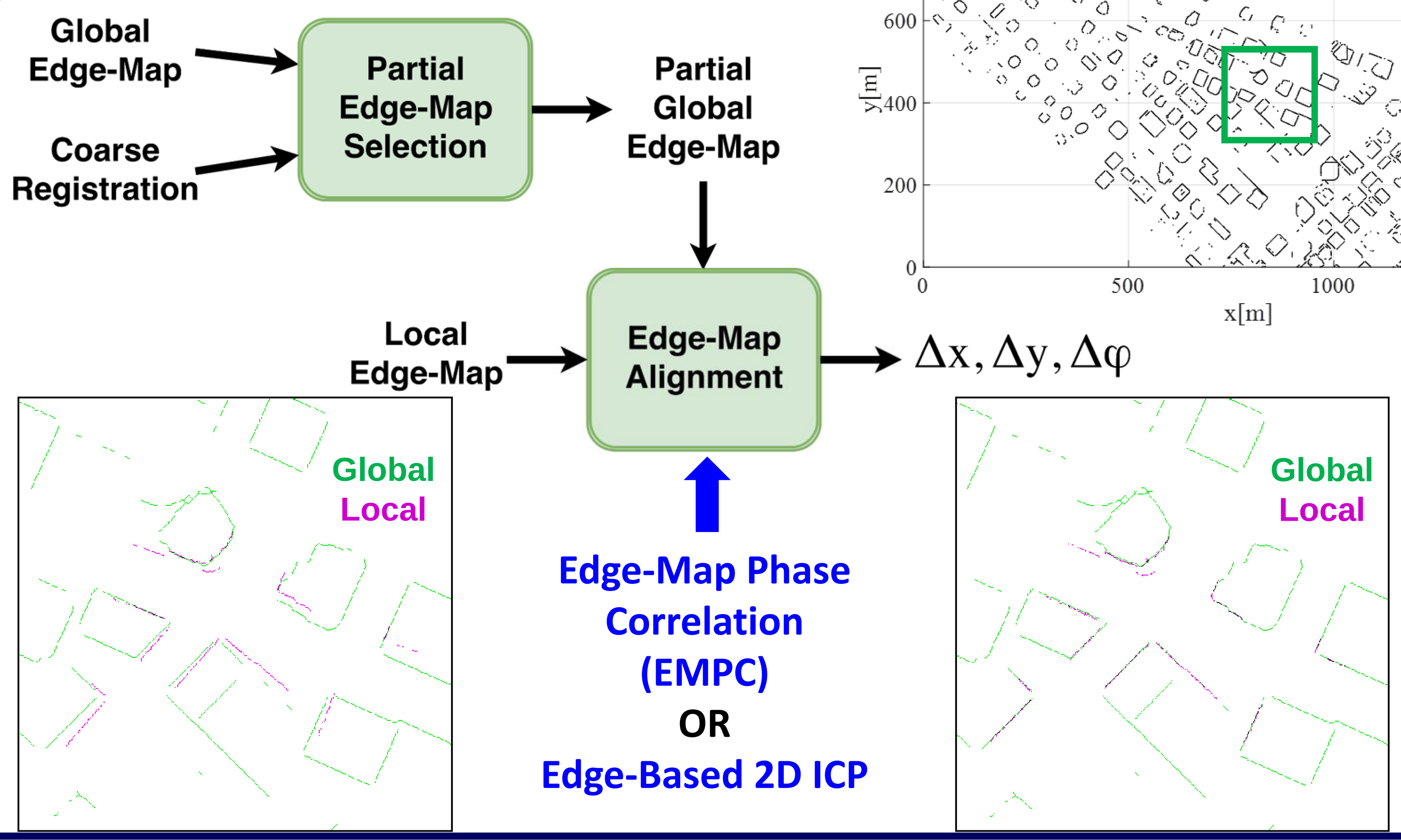
Avidar et al., ICCV17

- Main concepts:
  - Convert the global cloud into a **dictionary of viewpoint descriptors** (panoramic range-images)
  - Find **local-to-global registration via dictionary search** (using phase-correlation between range-images)

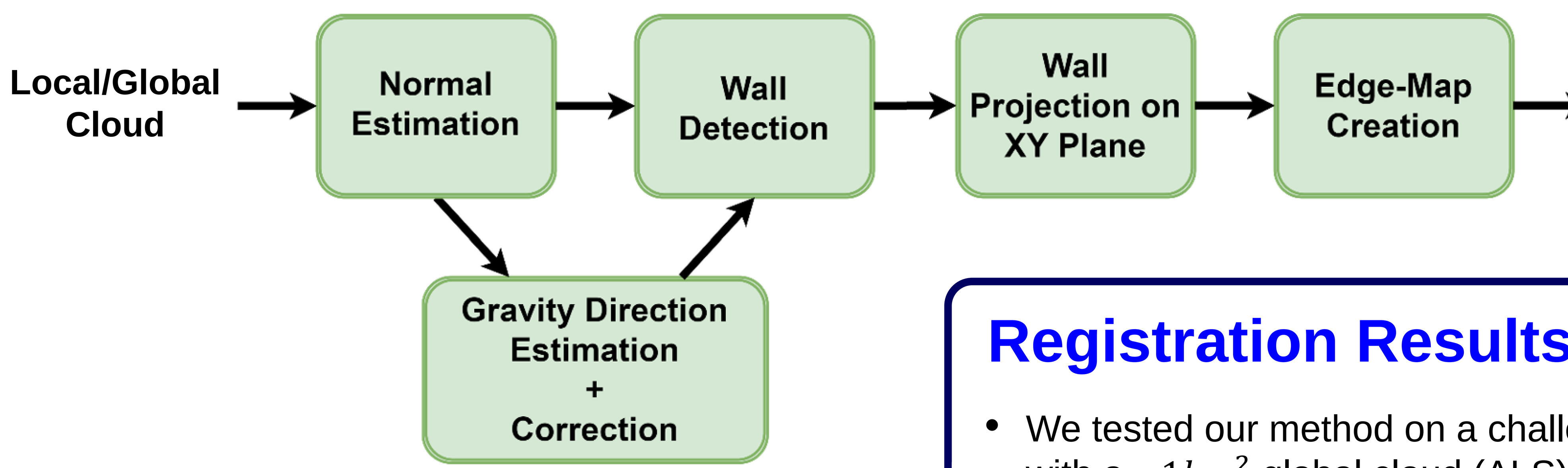



## Local-to-Global Edge-Map Alignment




## 3D Point Cloud to 2D Edge-Map Conversion

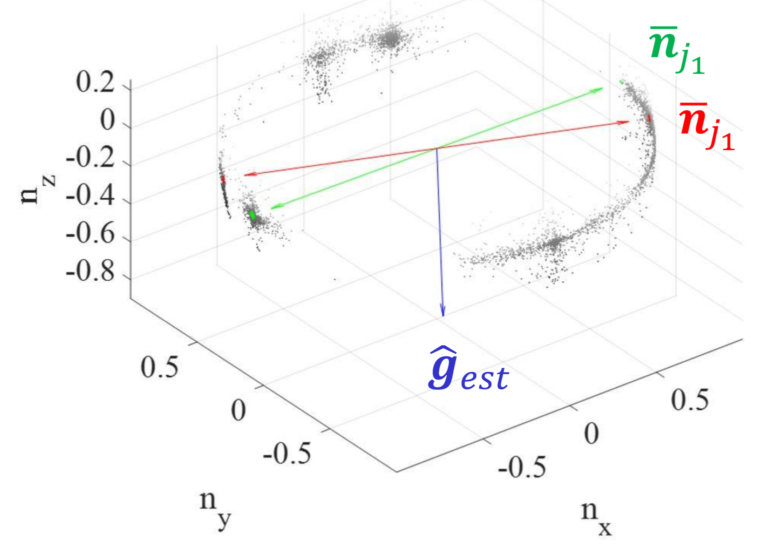



## Gravity Direction Estimation

- For each pair of dominant normal vector orientations ($\bar{n}_{j_1}$ and $\bar{n}_{j_2}$) in the point cloud:
  - Estimate gravity direction $\hat{g}_{est} = -\bar{n}_{j_1} \times \bar{n}_{j_2}$
  - Count inliers: # normals $\boldsymbol{n}_i$ where:

    $|\sphericalangle(\boldsymbol{n}_i, \hat{g}_{est}) - 90^o| < 2\beta \quad$ (e.g., $\beta = 0.5^o$)
- Finally, select $\hat{g}_{est}$ with the most inliers

Mapping of normal vectors in a TLS point cloud to the unit sphere:




## Edge-Based 2D ICP

- ICP (Iterative Closest Point) is a widely used method for registration of 3D point clouds (Besl and McKay, 1992)
- It is an **iterative method** that alternates between:
  - Finding nearest neighbor pairs of points between two point clouds
  - Minimizing the distances between these pairs of points
- We use **ICP in 2D between the global and local edge-maps** by converting them to 2D point clouds

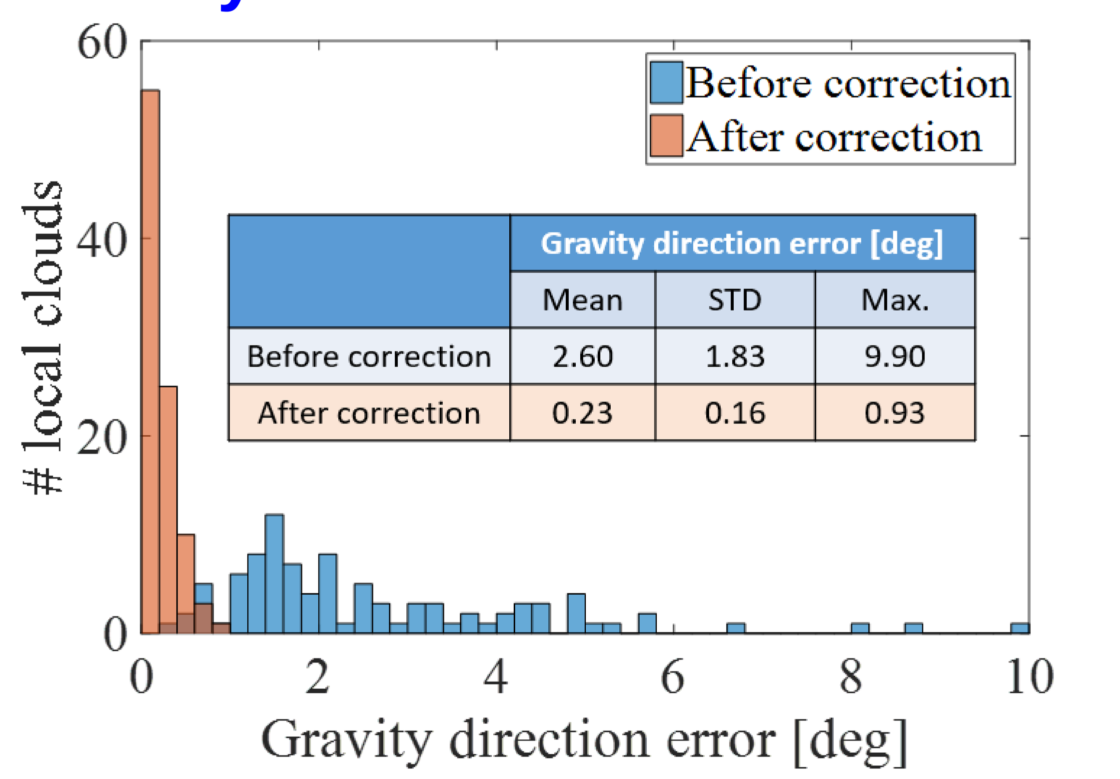
## Registration Results

- We tested our method on a challenging dataset with a $\sim 1km^2$ global cloud (ALS) and 94 local clouds (TLS)

| Registration Refinement Method | Localization Error [m] | | | Relative Rotation Error [deg] | | | Runtime* [sec] | | |
|---|---|---|---|---|---|---|---|---|---|
| | Mean | STD | Max | Mean | STD | Max | Mean | STD | Max |
| **3D ICP** | **0.40** | **0.20** | 1.13 | 0.67 | **0.30** | 1.73 | 1.62 | 0.35 | 2.71 |
| **Edge-Map Phase-Correlation (EMPC)** | 0.57 | 0.45 | 3.21 | 0.98 | 0.76 | 5.65 | 0.46 | **0.02** | **0.54** |
| **Edge-Based 2D ICP** | **0.40** | 0.21 | **1.09** | **0.60** | 0.38 | **1.61** | **0.44** | 0.08 | 0.83 |

*Run on PC with i7-5820 CPU @ 3.30GHz using MATLAB

## Gravity Estimation Results




| | Gravity direction error [deg] | | |
|---|---|---|---|
| | Mean | STD | Max. |
| Before correction | 2.60 | 1.83 | 9.90 |
| After correction | 0.23 | 0.16 | 0.93 |

## Conclusions

- The proposed method (using 2D ICP) achieves a **reduction in runtime by a factor of 3.7** in comparison to 3D ICP, while maintaining **similar registration accuracy**.
- The proposed **gravity direction estimation method** achieves a **mean error of $0.23^o$** - an **order of magnitude lower than before** gravity direction correction is applied).

In collaboration with GeoSim

ICSEE, December 2018