Hamburg University of Technology

Pavan Vishwanath

Institute of Medical Technology

Medical Technology | Science | Systems

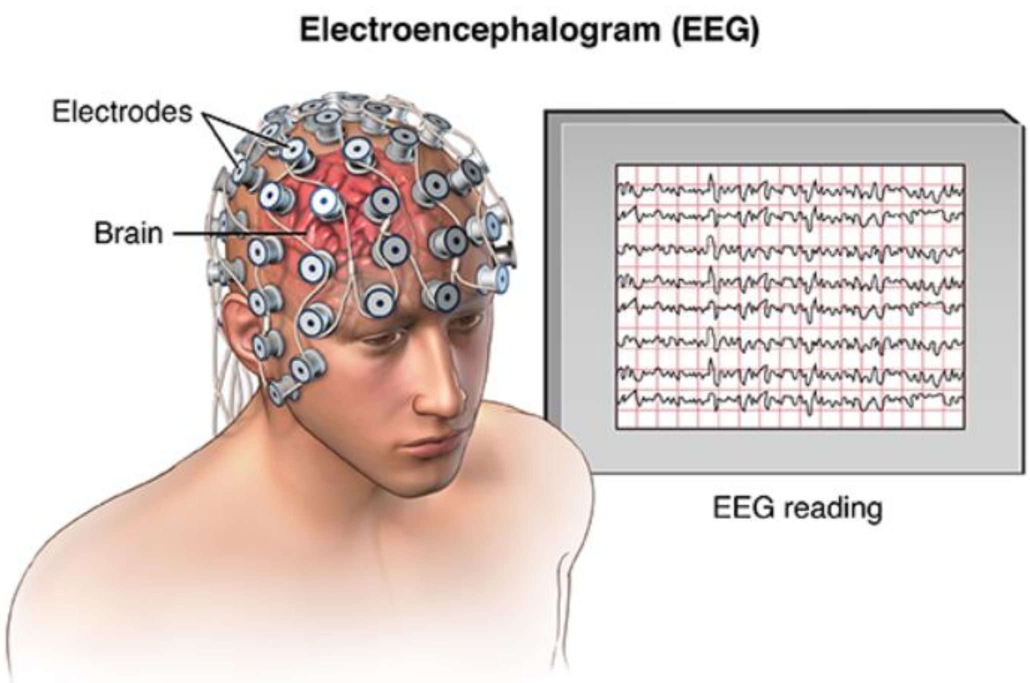
Simultaneous localization and mapping for camera-based EEG electrode digitalization

Agenda

- Motivation
- Master thesis overview
- Results & Discussions
- Conclusion and Next steps

Motivation

- Electrode position ?



EEG [1]

Accurate 3D electrode position using RGBD sensor

Literature

- TOF (depth) + color camera

FIGURE 1 | The camera system.

MESA-SR-4000 (~ several thousands $)

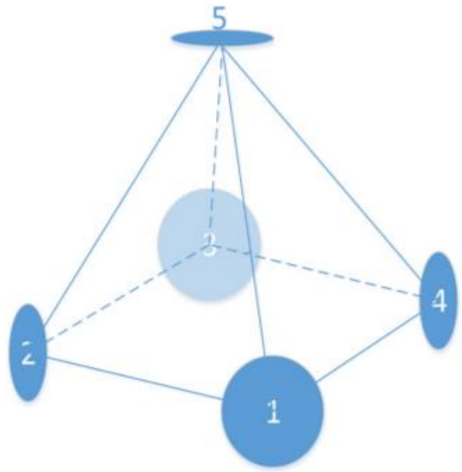


FIGURE 3 | The schematic diagram.

Fixed camera position

Spatial Localization of EEG Electrodes in a TOF+CCD Camera System [2]

Fixed Camera position and expensive depth camera

Motivation

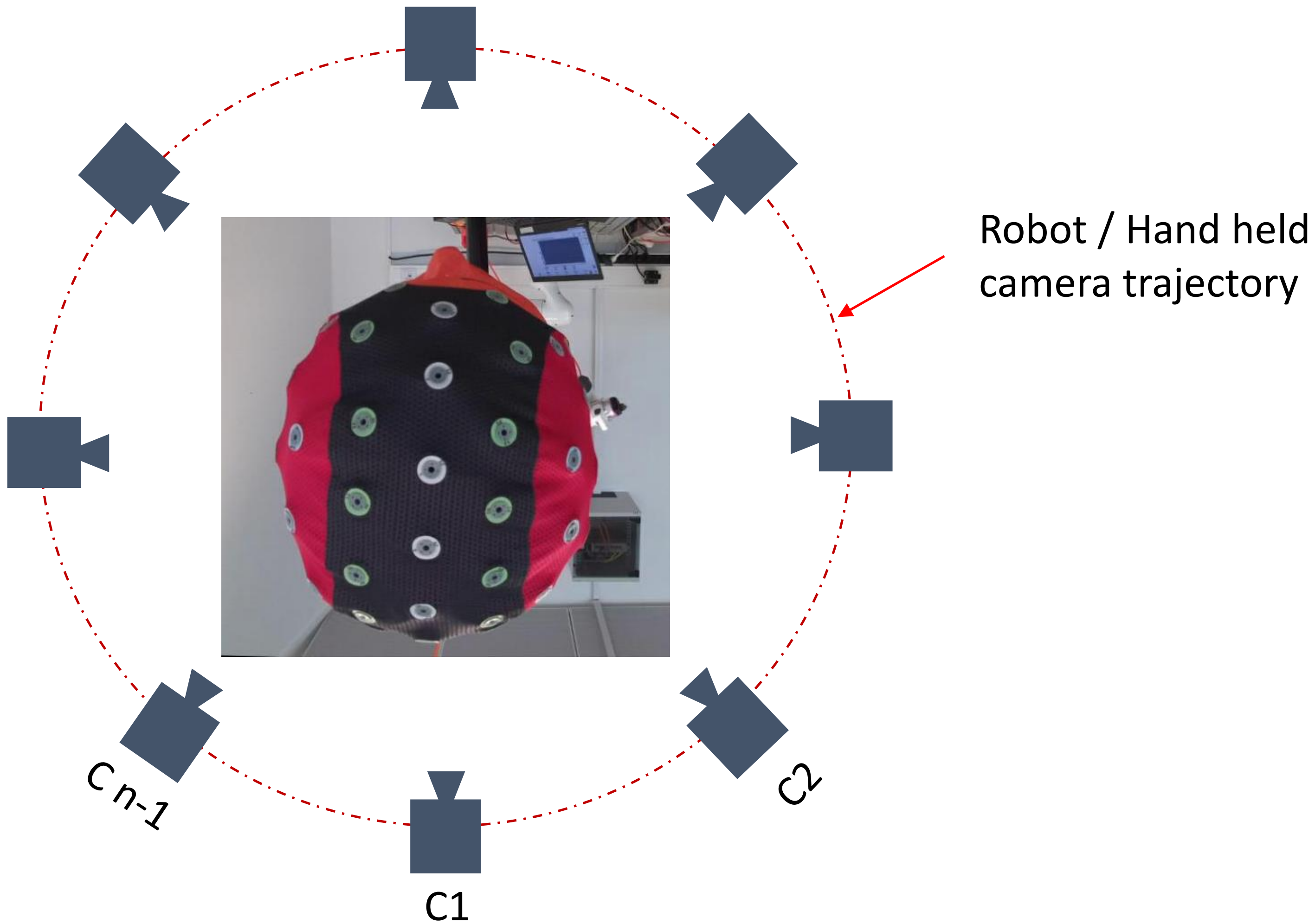
- Literature

TABLE 3 | Comparison of the typical methods.

Method	Principle	Equip size	Time	Accuracy	Reliability	Typical Ref.
Manual measurement	Coordinate measuring, calipers	Small	Very slow (>10 min)	0.4 mm	Mid	De Munck et al. (1991)
Camera matrix	Stereo vision	Large	Real-time (<0.1 s)	1.27 mm	Bad	Koessler et al. (2007)
Positioning tool	Electromagnetic digitizer	Small	5 min	2–8 mm	Mid	Dalal et al. (2014)
Photogrammetry	Structure-from-motion	Small	Slow (5–10 min)	0.8 mm	Mid	Clausner et al. (2017)
Laser scanner	Laser	Small	Slow	0.05–0.2 mm	Good	Jeon et al. (2018)
Color+depth	Color+TOF	Small	Real-time	0.3–3.3 mm	Good	This report

Spatial Localization of EEG Electrodes in a TOF+CCD Camera System [2]

Bird’s eye view



Continuous camera trajectory but few shown for clarity

Master Thesis : Overview

- Electrode position (YOLO + depth images)

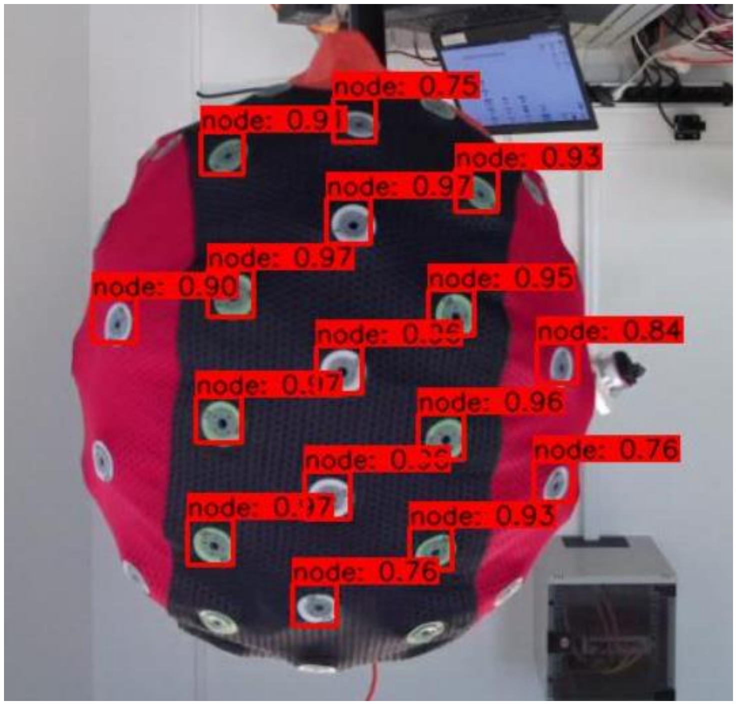
↓

- Sequential ICP registration
- Apply SLAM (Pose Graph, loop closure, trajectory optimization)
- electrode clustering and cluster centroid estimation (kMeans analysis)
- ~~IMU fallback (failure cases)~~
- Evaluation on robot and ~~hand held trajectory~~

↓

- Head pose estimation

YOLO + depth



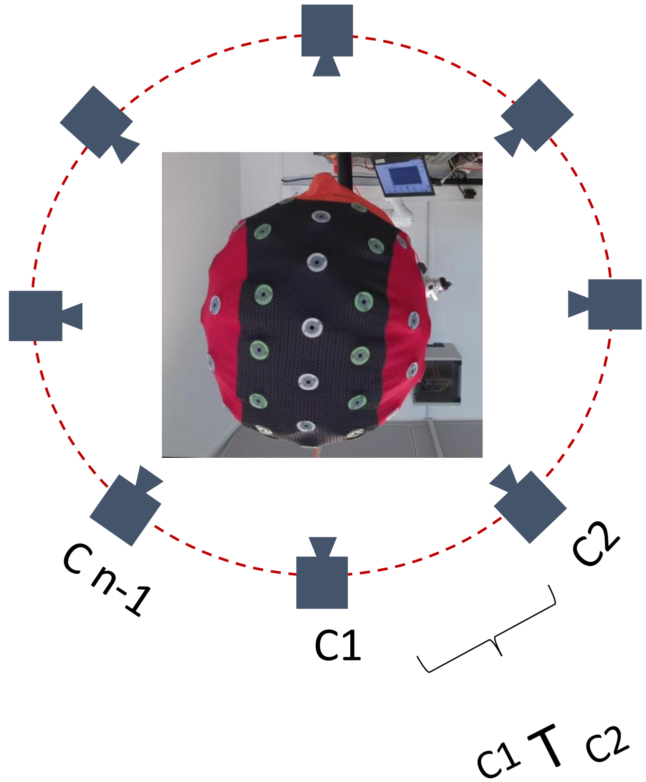
Courtesy: Debayan Bhattacharya

+

Cn

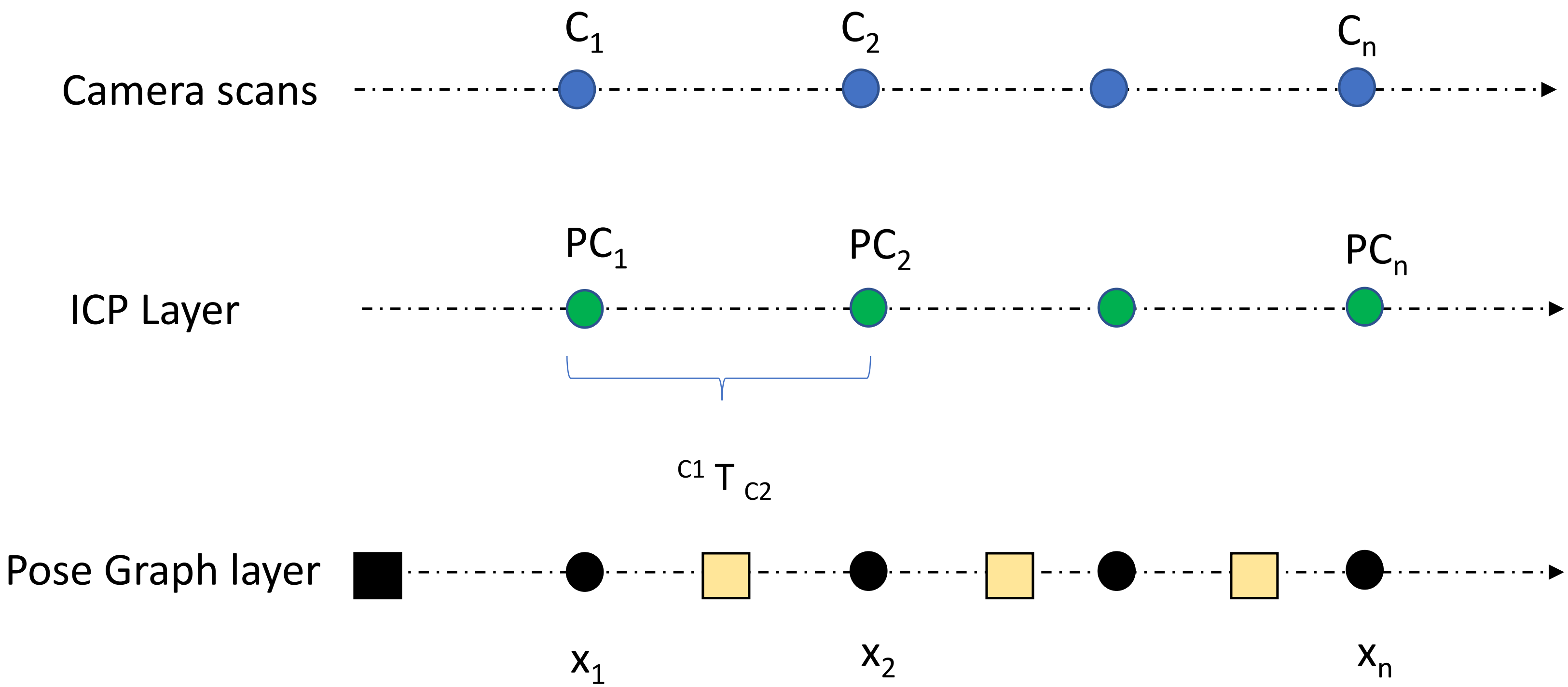
YOLO + Depth = 3D point cloud

Sequential ICP point cloud registration



Relative transformation for all the camera poses are calculated

Pose Graph



Camera position

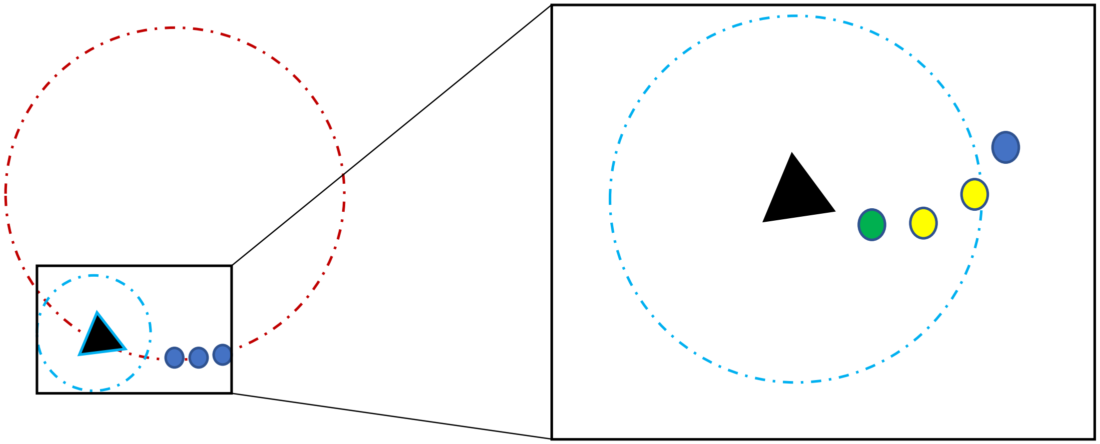
Point cloud

Pose graph node

Odometry constraints

Pose graph initialization

Loop closure



Current camera position

Search radius

Previous recorded Camera position

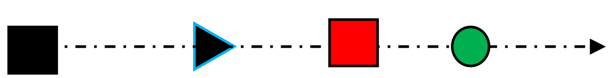
Potential loop Closure candidate

Loop closure candidate

- Loop closure
  - Euclidean distance – search radius
    - ICP - Fitness score
    - Fitness = # correspondences set / # points in the target point cloud

Loop closure constraint

Pose Graph layer

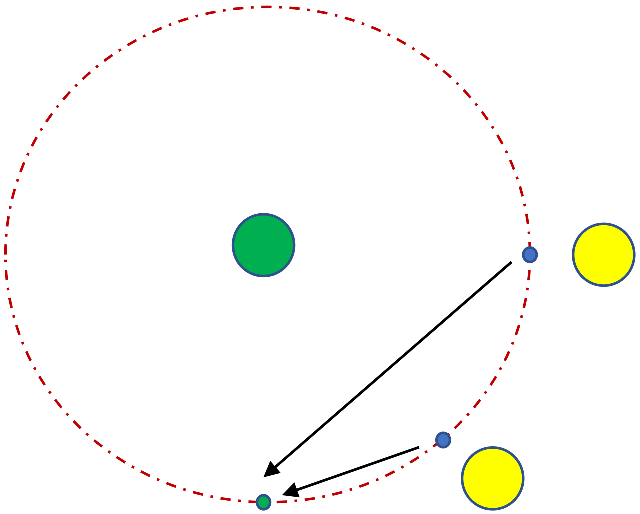


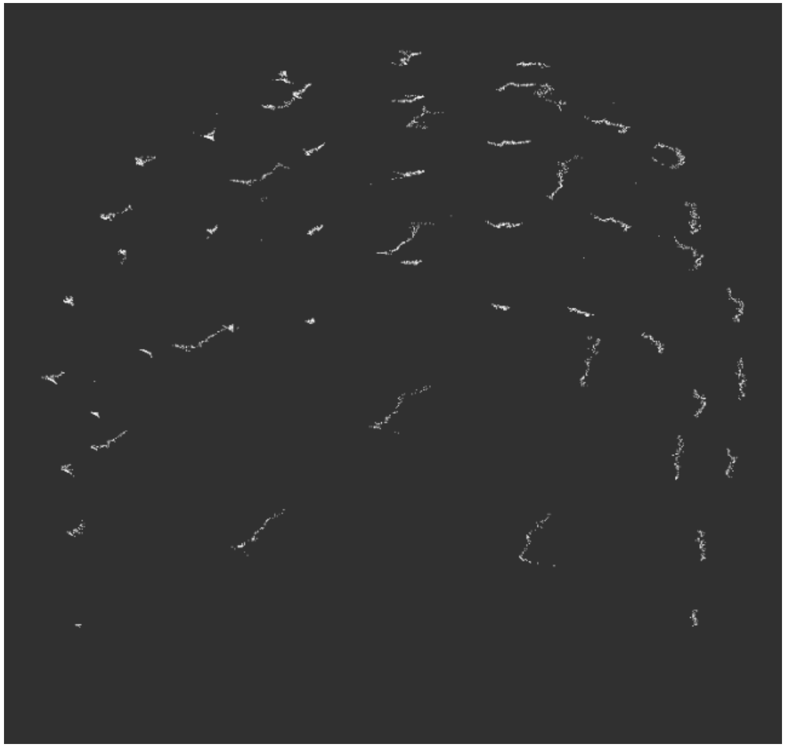
Pose graph optimization

- This is a probabilistic graph based slam approach.
- It’s a non linear least squares optimization problem.
- Library used: miniSAM https://arxiv.org/pdf/1909.00903.pdf

Pose graph with odometry and loop closure constraints → miniSAM → Optimized camera trajectory

Building Map (point cloud stitching)

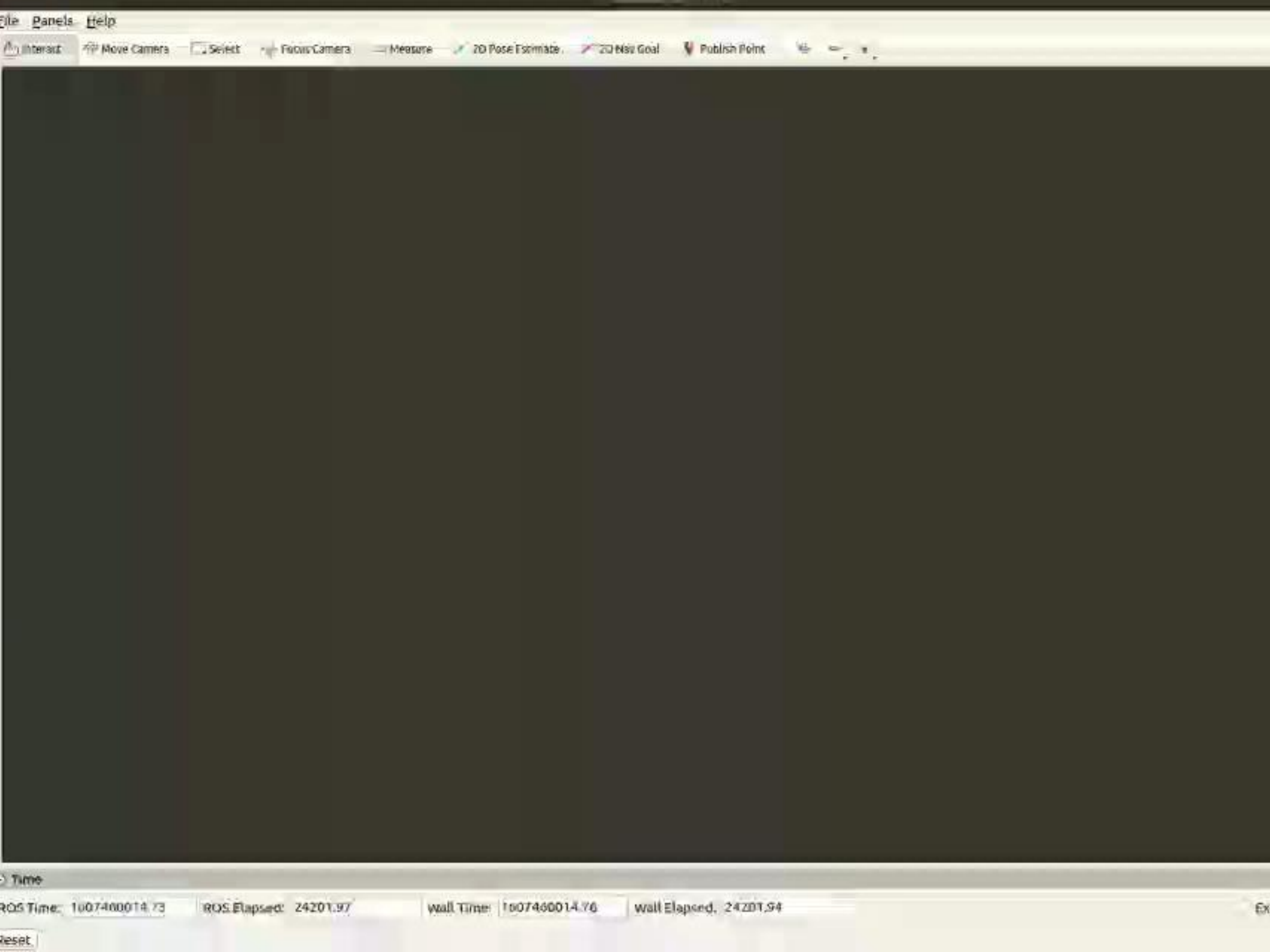




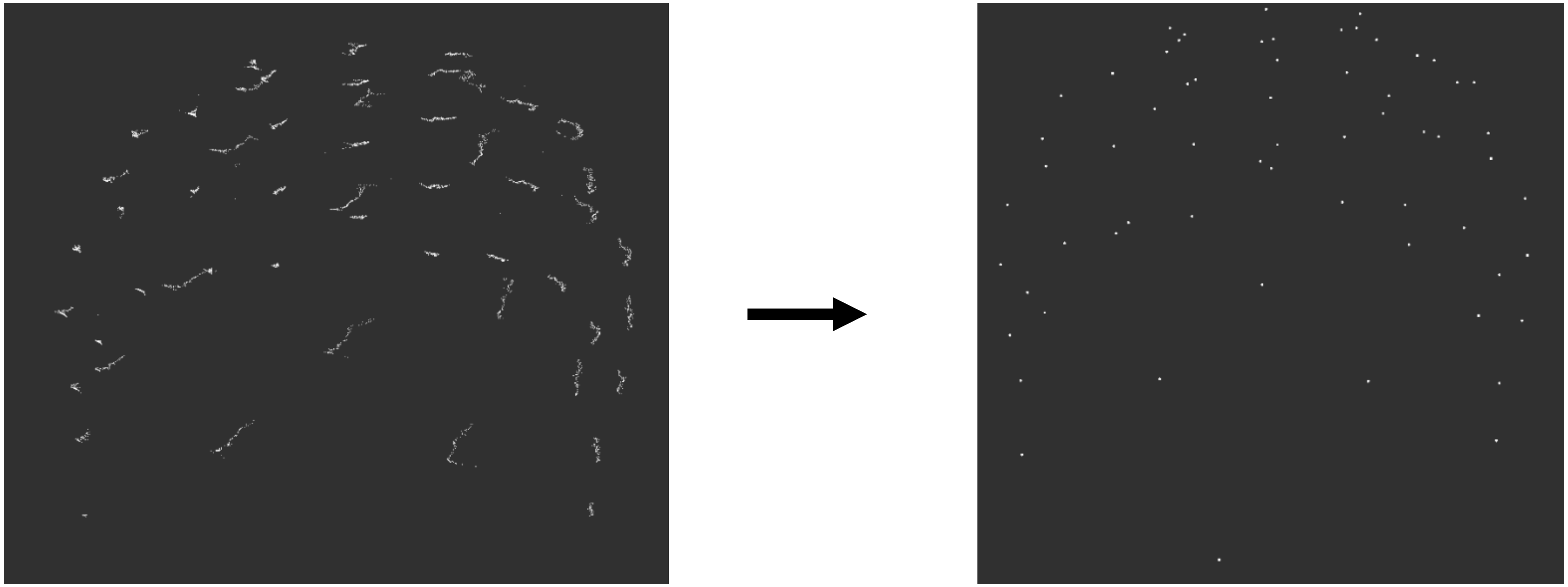
- Each camera position has associated point cloud.
- Calculate camera's absolute position w.r.t to first position.
- Stitch the point cloud.

- for loop :
  - ICP registration
  - Add not to pose graph
    - If (loop closure)
      - Optimize pose graph
  - Build map

ICP-based pose-graph SLAM [3]


File Panels Help
Interact
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
2D Nav Goal
Publish Point
Time
ROS Time: 1607460014.73
ROS Elapsed: 24201.97
Wall Time: 1607460014.76
Wall Elapsed: 24201.94
Reset

▸ Clusters centers



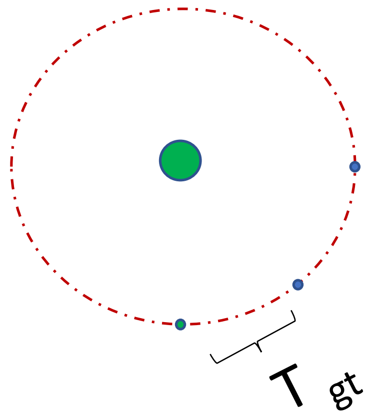
Number of clusters is already known

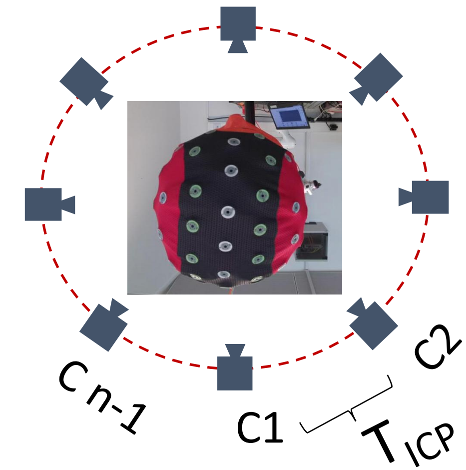
Brief look at data sets

- Ground truth data set
  - 3D point cloud
    - Axios 3D tracking camera
    - Stylus with reflective markers
  - Camera trajectory
    - Kinect
    - Robot (ground truth trajectory)

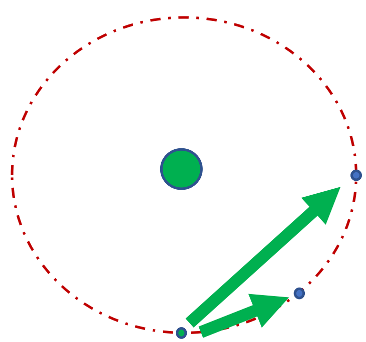
Metrics

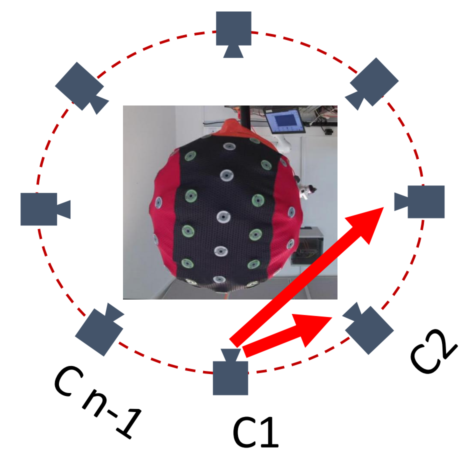
- ICP pose





- Absolute Pose

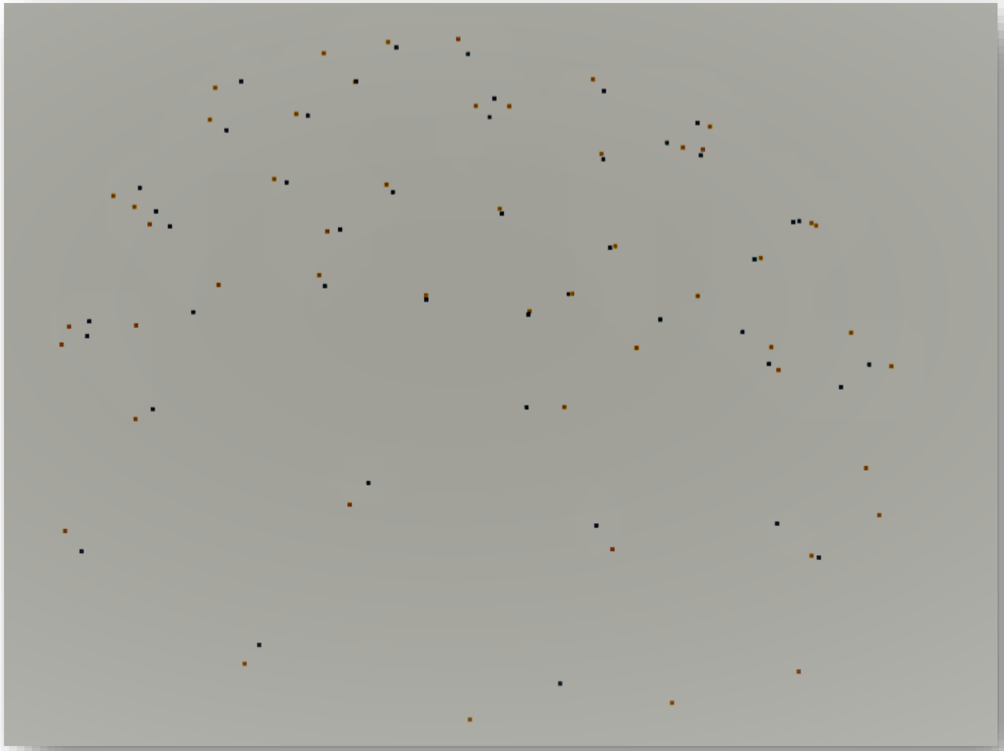




∑ Rotation (axis angle), ∑ L2 norm for translation, Mean +/- STD Dev

Metrics

- 3D electrode position
  - Directly compare the 3D position of cluster centers
- Relative transformation between 2 clusters centers point cloud
  - Calculate the RMSE of all correspondences



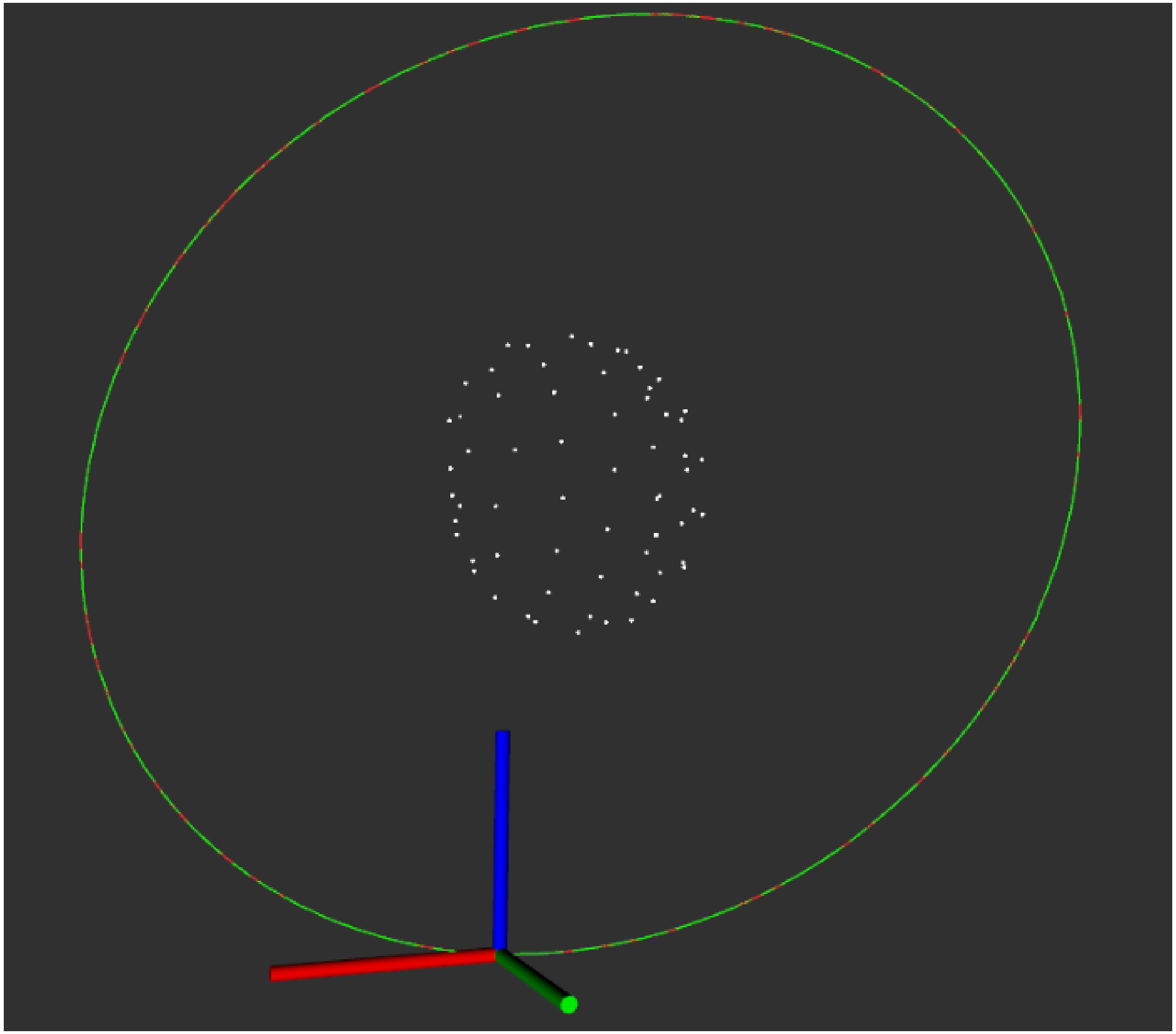
Different Caps

- EEG cap

23 electrode cap

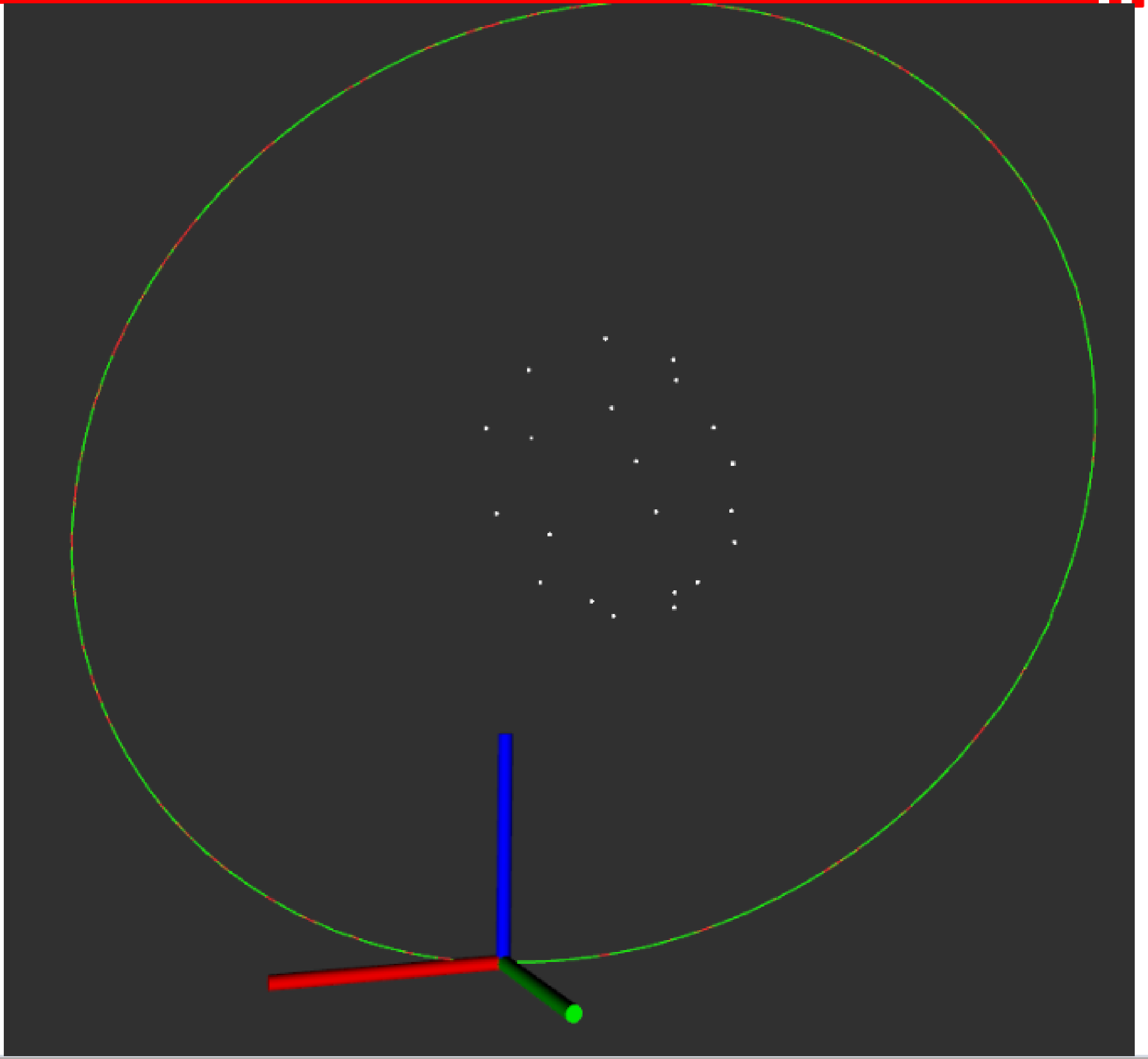
63 electrode cap

Results: Ground truth electrodes → SLAM CA_124_7, 63 electrodes

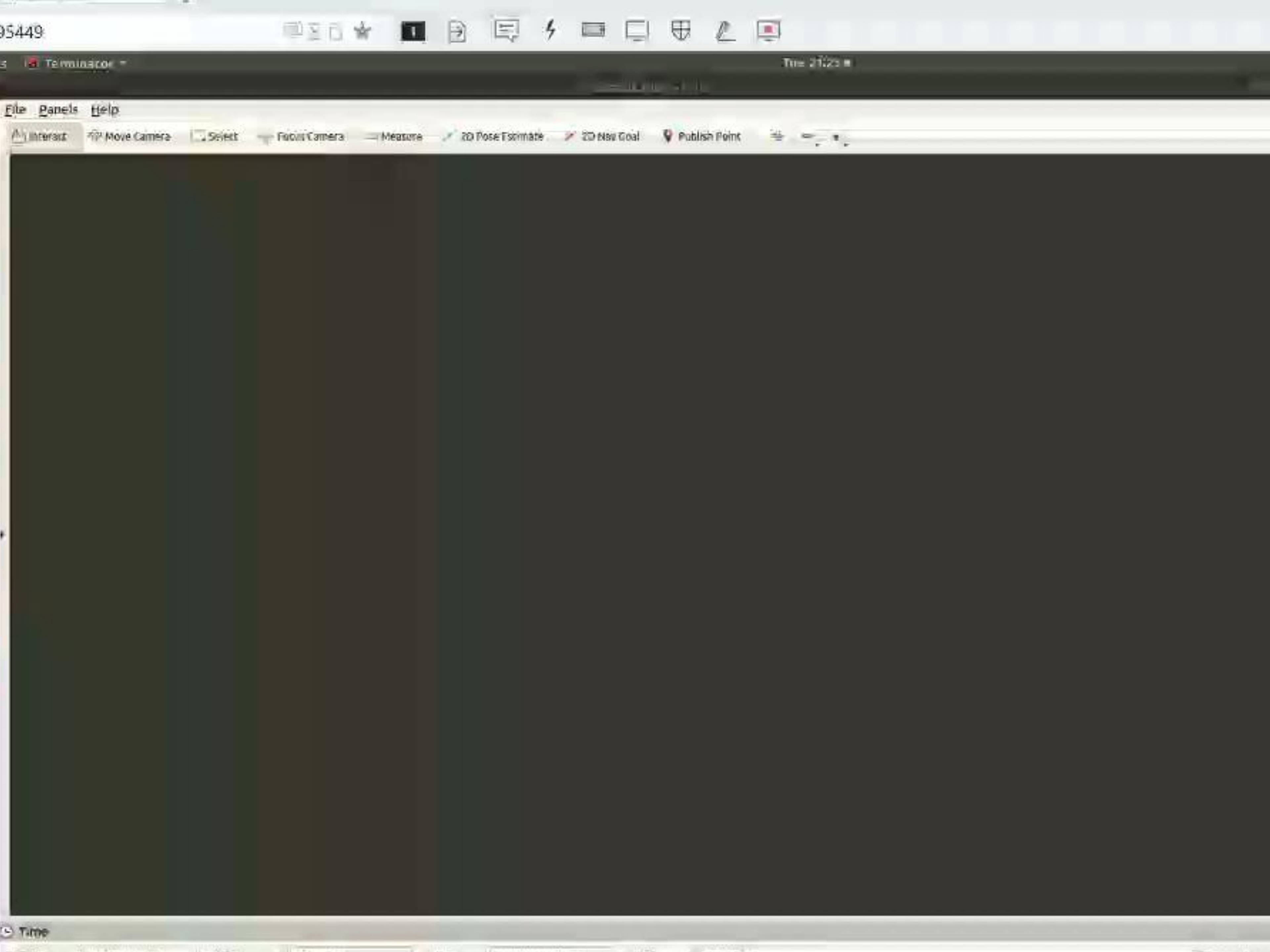


Accurate Localization and mapping is possible

Results: Ground truth electrodes → SLAM CS_301_7, 23 electrodes

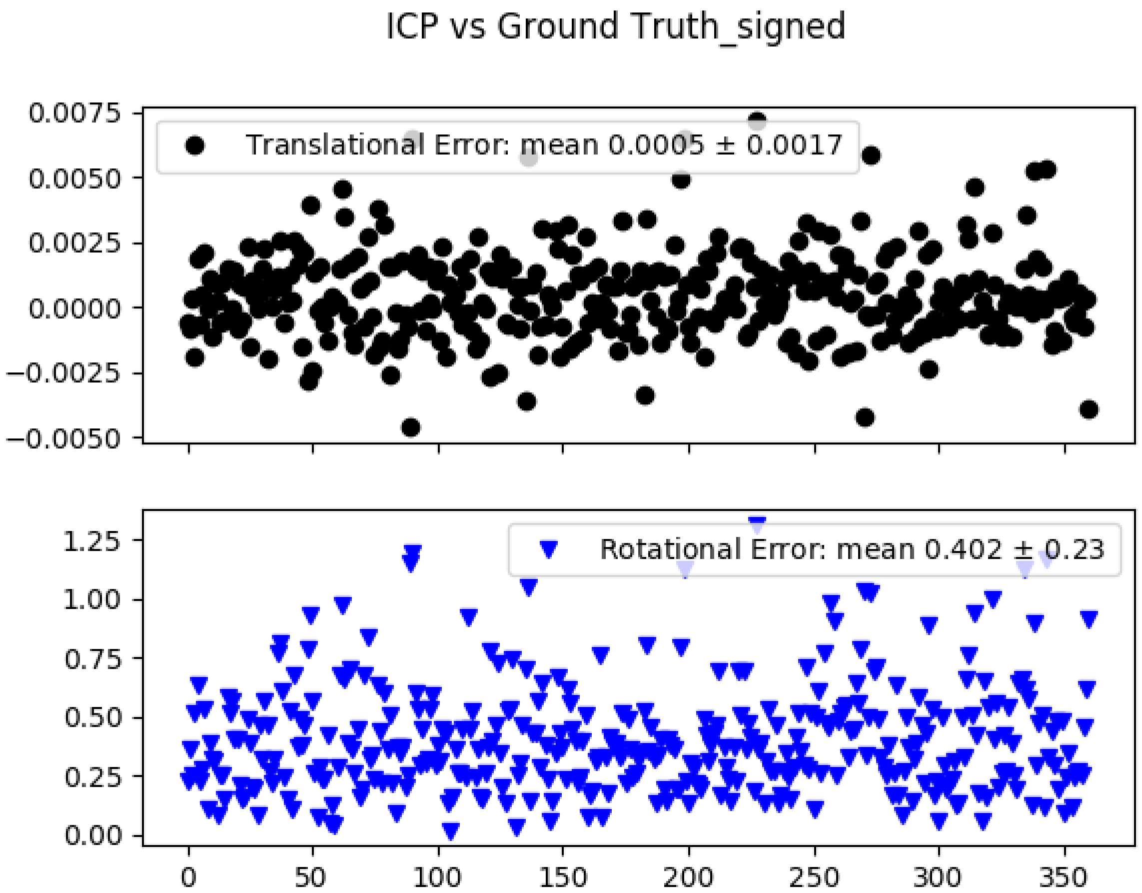


Accurate Localization and mapping is possible even with 23 electrode cap


File
Panels
Help
Interact
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
2D Nav Goal
Publish Point
Time

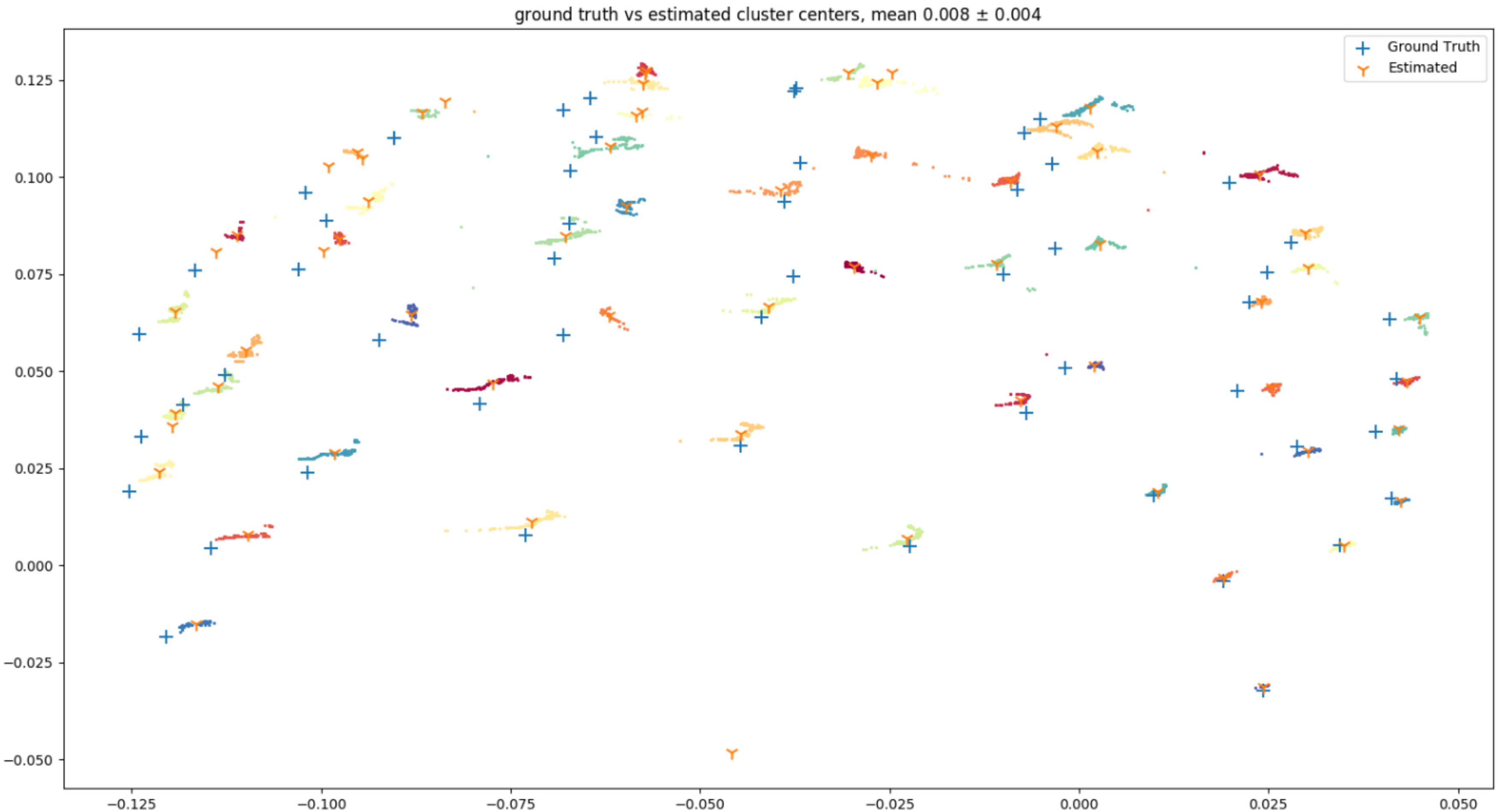
Results: YOLO + Depth → SLAM   Cap: CA_124_7 (63 electrodes)

- ICP pose vs ground truth pose



Results: YOLO + Depth → SLAM Cap: CA_124_7 (63 electrodes)

- Cluster centers (2D image)



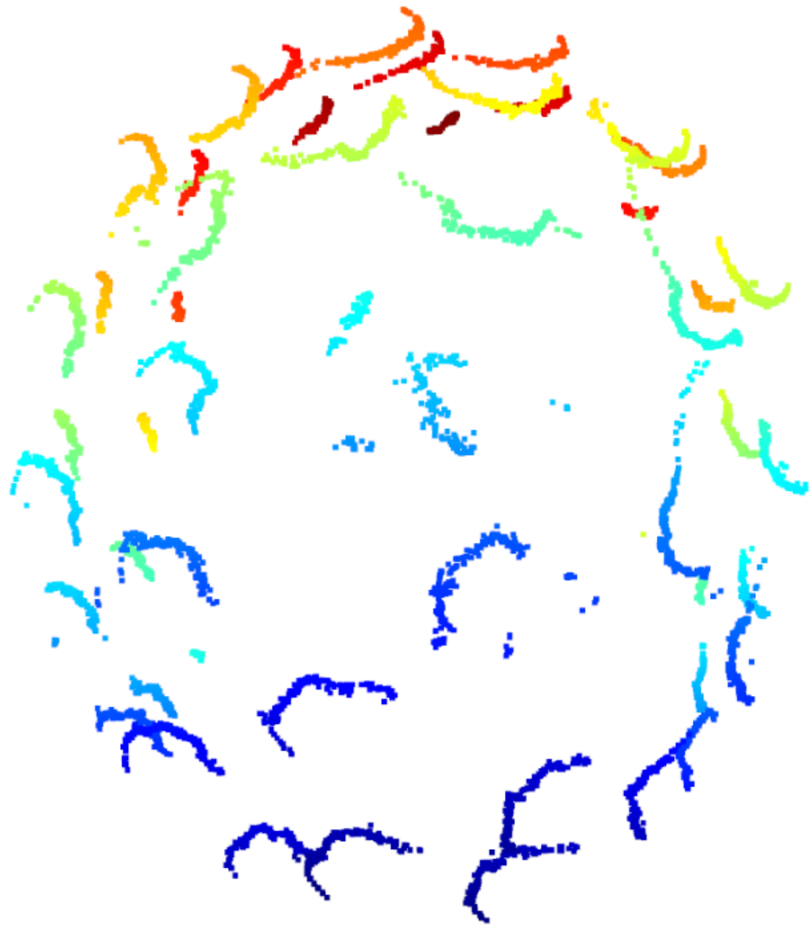
Mean 0.008 ± 0.004 (meters)

Results: Summary of all the caps

ICP vs Ground truth			
	CAP (63 electrodes)	CAP (23 electrodes) 1	CAP (23 electrodes) 2
Translation Error	0.4 ± 1.5 (mm)	1.15 ± 2.45 (mm)	2.6 ± 13.4 (mm)
Rotation Error	0.38 ± 0.2°	0.6 ± 0.3°	1.2 ± 6°

3D electrode position			
	CAP (63 electrodes)	CAP (23 electrodes) 1	CAP (23 electrodes) 2
L2 Norm	11 ± 4.5 (mm)	30 ± 8.5 (mm)	24.5 ± 11 (mm)
RMSE post Registration	2.85 ± 0.3 (mm)	21 ± 7.3 (mm)	19 ± 6.5 (mm)

YOLO + depth + ground truth trajectory



YOLO + Depth = 3D point cloud needs to be more accurate

- Conclusion
  - Accurate localization is possible provided accurate 3D point clouds
- Next steps
  - 3D point cloud from YOLO + depth needs more quantitate evaluation
  - Test algorithms on hand held trajectory.
  - IMU integration.

Depth ambiguity

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & S & C_x \\ 0 & \alpha_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}$$

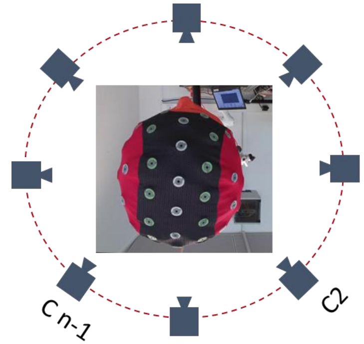
Depth map + Yolo (pixel space)

3D point cloud

Initial depth calibration using checker board revealed 14 mm offset

3D point cloud = f (depth)

Depth offset determination



At each camera position, both ground truth and Yolo + depth point cloud are known

ICP: ground truth and yolo depth

RMSE

Entire point cloud stitch (mapping)

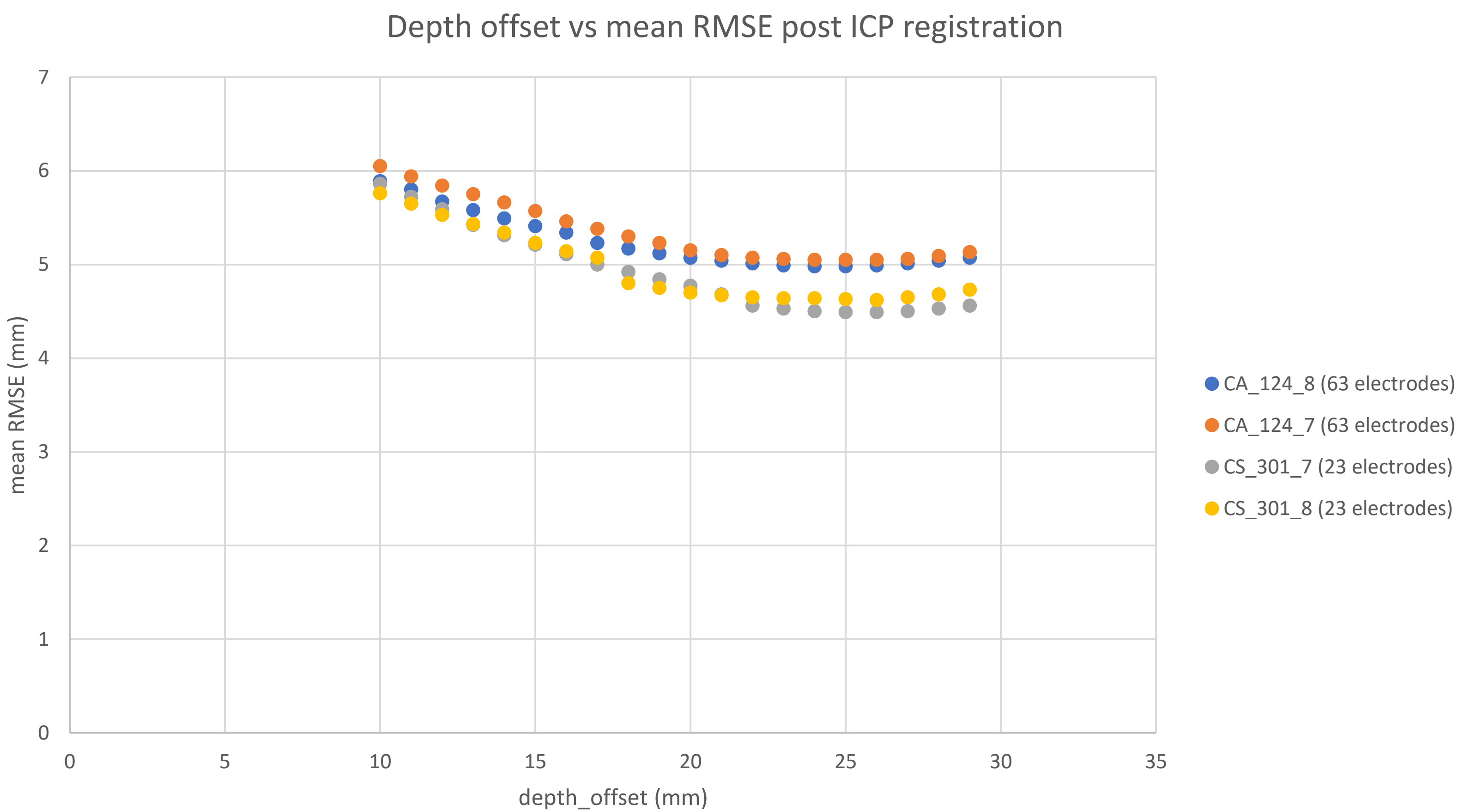
Cluster center (ground truth)

Cluster center (yolo+depth)

L2 norm

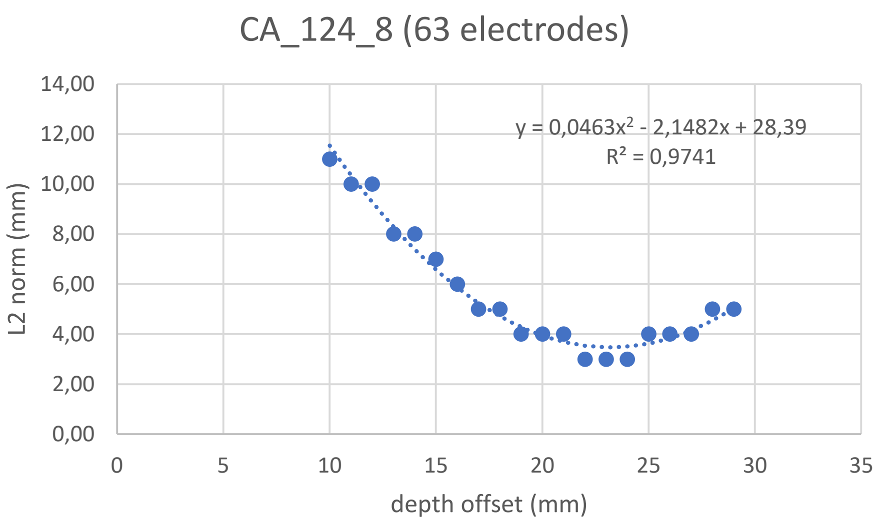
RMSE & L2 norm for different depth offset values.

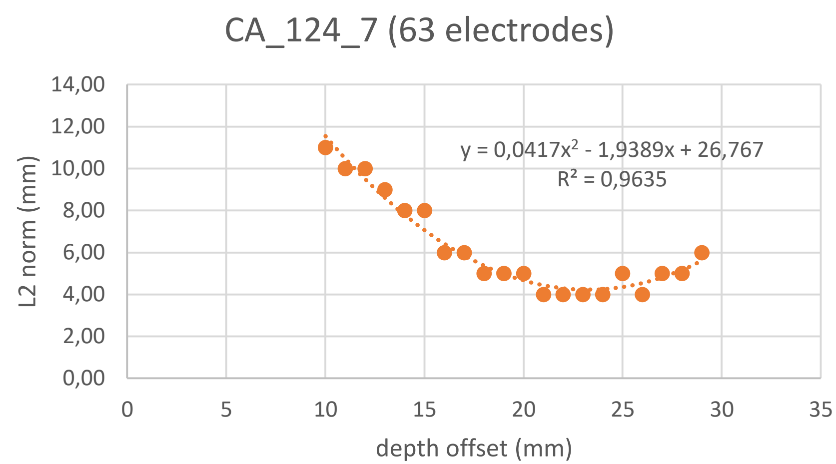
ICP: ground truth and yolo depth (only mean values are plotted)

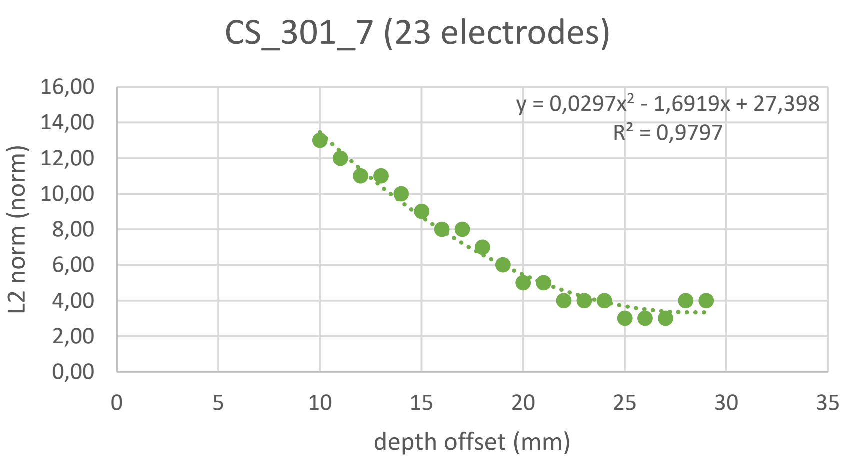


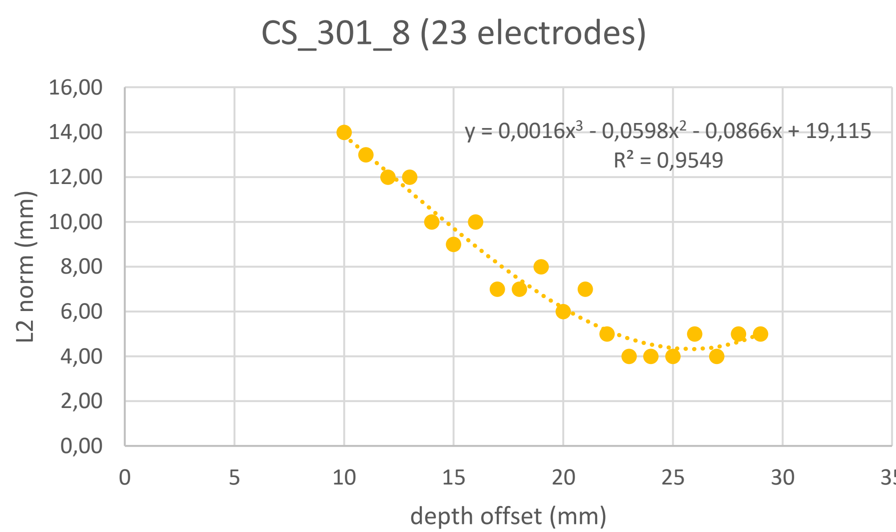
Low mean RMSE for all caps @ 25 depth offset.

Cluster center L2 norm (mean values are plotted)



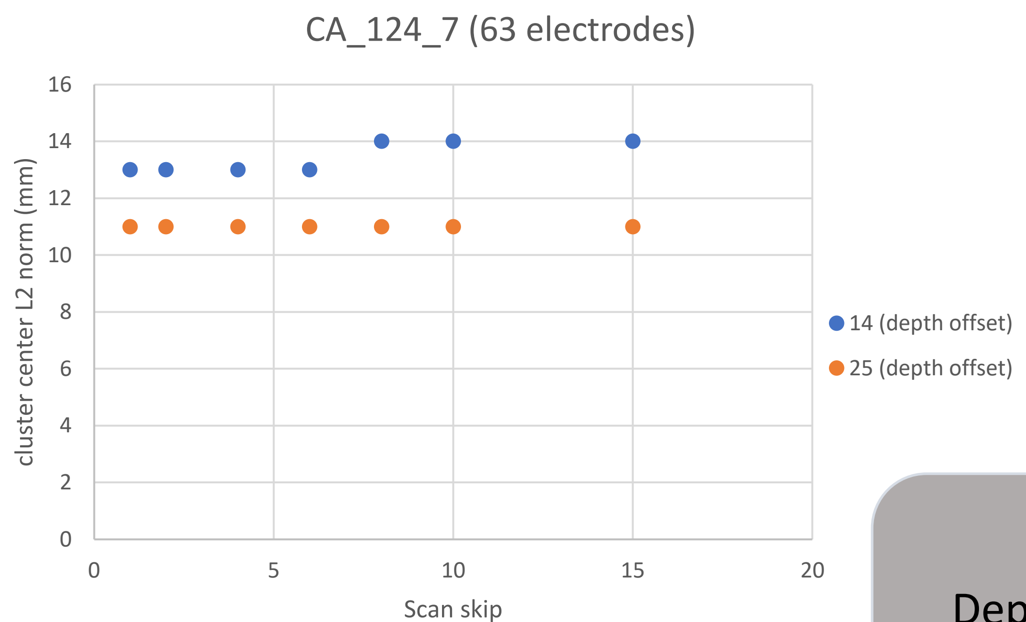




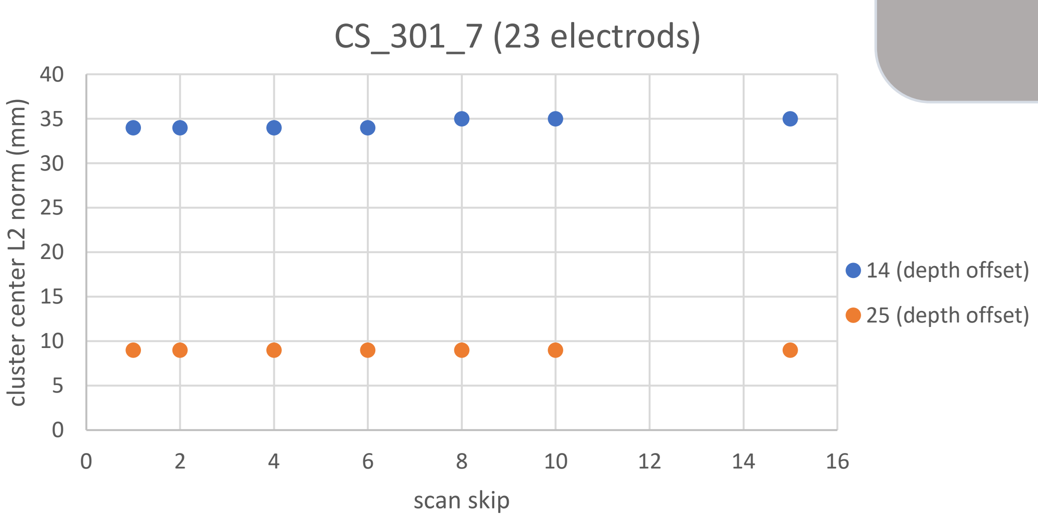


Low mean RMSE for all caps between 23 - 25 depth offset.

Results depth offset 14mm VS 25mm



Depth offset 25mm has low L2 norm
Skipping scans doesn't help much



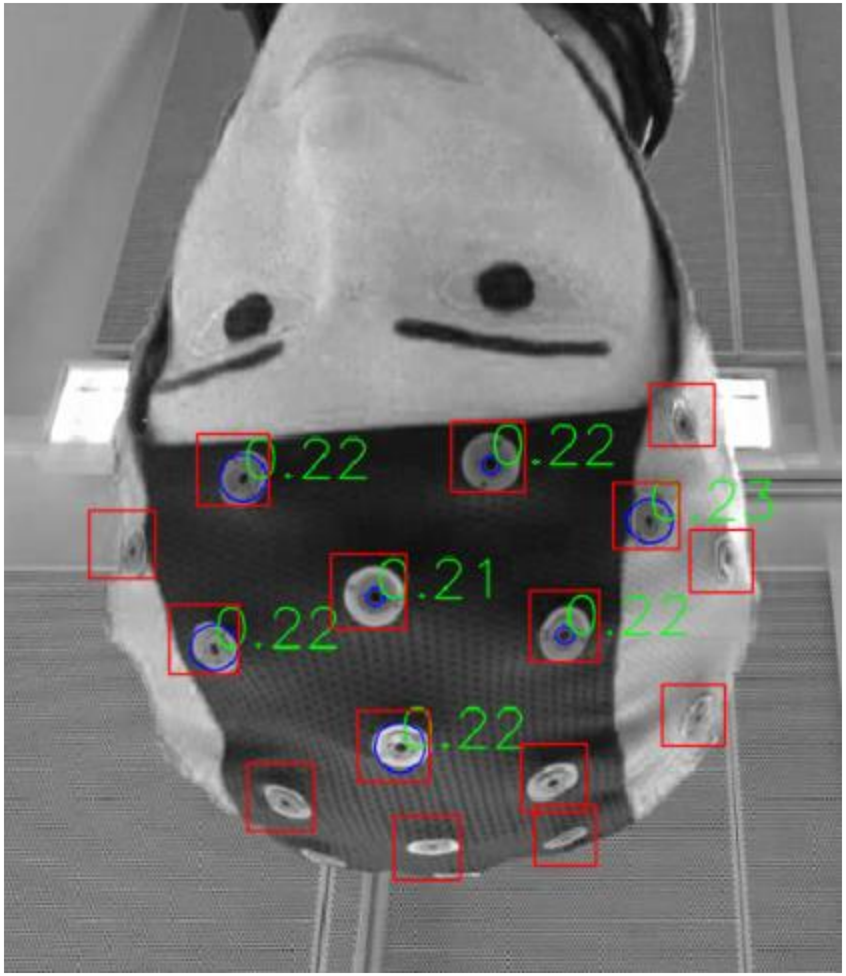
Results: Summary of all the caps depth offset = 25mm

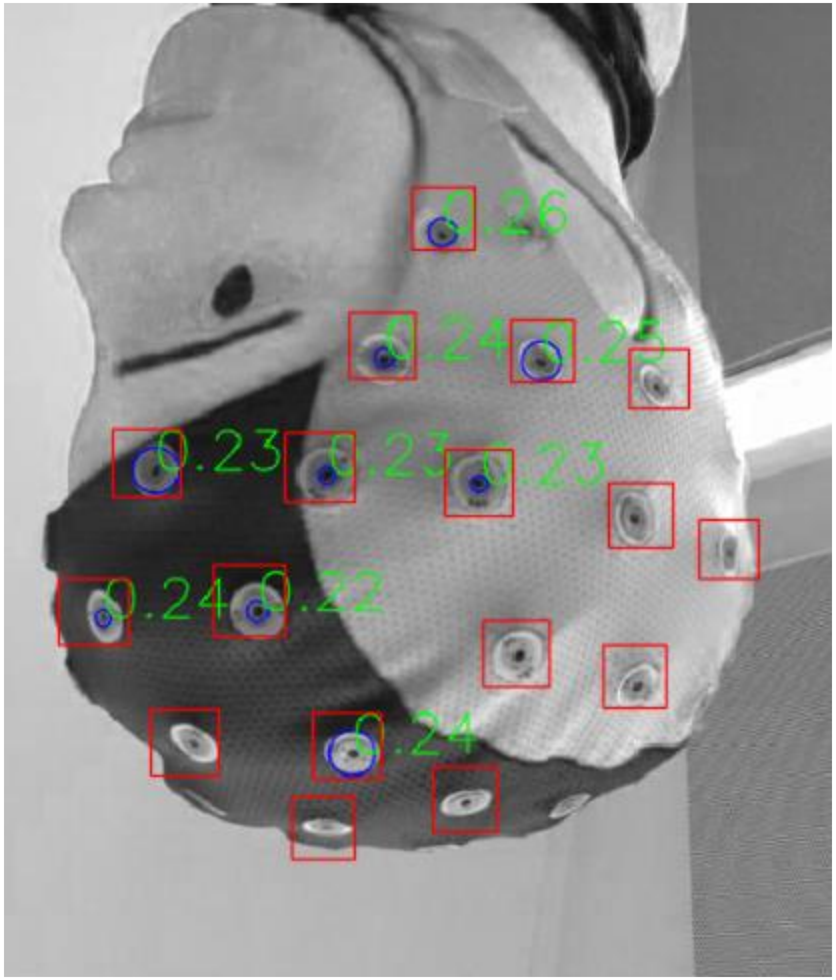
ICP vs Ground truth		
	CAP (63 electrodes)	CAP (23 electrodes) 1
Translation Error	0.25 ± 1.6 (mm)	0.8 ± 2.4 (mm)
Rotation Error	0.4 ° ± 0.2 °	0.6 ° ± 0.3 °

3D electrode position		
	CAP (63 electrodes)	CAP (23 electrodes) 1
L2 Norm	11 ± 5 (mm)	11.5 ± 4.5 (mm)
RMSE post Registration	3.5 ± 0.5 (mm)	5.5 ± 0.5 (mm)

There is some error in the point cloud of CAP (23 electrodes) 2, hence its is not considered

Blob detection after yolo





Blob detection after YOLO helps for corner electrodes. Blob detection needs to be tuned as it doesn't recognize electrodes in all the YOLO boxes

The End

Thank You

Bibliography

[1] en.wikipedia.org. (1999). “MS Windows NT kernel description,” [Online]. Available: https://en.wikipedia.org/wiki/Electroencephalography (visited on 04/15/2010).

[2] Chen S, He Y, Qiu H, Yan X and Zhao M (2019) Spatial Localization of EEG Electrodes in a TOF+CCD Camera System. Front. Neuroinform. 13:21. doi: 10.3389/fninf.2019.00021

[3] Ellon Mendes, Pierrick Koch, Simon Lacroix. ICP-based pose-graph SLAM. International Symposium on Safety, Security and Rescue Robotics (SSRR), Oct 2016, Lausanne, Switzerland. pp.195 - 200, ff10.1109/SSRR.2016.7784298ff. ffhal-01522248f

Backup

	SR4000	D-IMager	Kinect v2
Released	2010	2012	2014
Image sensor size	176 x 144	160 x 120	512 x 424
FOV	43.6 x 34.6	60 x 44	70 x 60
Range (specified)	0.5 - 5.0 m	1.2 - 5.0 m	0.5 - 4.5 m
Max range (actual)	up to 15 m	15 m	$\tilde{9}$m
Depth Resolution	$\pm$ 10mm at 5m, $\pm$ 15mm at 10m	3cm at 0 lx, 14cm at 100 klx	N/A
Depth data	16bit (0-65535)	11bit (0-1500)	13bit (0-8096)
Amplitude data	16bit (0-65535)	8bit (0-255)	16bit (0-65535)
Dimensions (mm)	65 x 65 x 68	170 x 54 x 50	250 x 66 x 67
Weight	470 g	550 g	970 g
Connection type	USB2.0	USB2.0	USB3.0
Cooling system	passive	passive	Active
Power rating	12V, 0.8A (1.0A peak)	24V, 0.35A (2.5A peak)	12V, 2.67A

Courtesy: https://aaltodoc.aalto.fi/bitstream/handle/123456789/19184/master_Laukkanen_Matti_2015.pdf?sequence=1

Backup

2.3.10.3. Homogeneity, completeness and V-measure ¶

Given the knowledge of the ground truth class assignments of the samples, it is possible to define some intuitive metric using conditional entropy analysis.

In particular Rosenberg and Hirschberg (2007) define the following two desirable objectives for any cluster assignment:

- **homogeneity**: each cluster contains only members of a single class.
- **completeness**: all members of a given class are assigned to the same cluster.

We can turn those concept as scores `homogeneity_score` and `completeness_score`. Both are bounded below by 0.0 and above by 1.0 (higher is better):

```
>>> from sklearn import metrics
>>> labels_true = [0, 0, 0, 1, 1, 1]
>>> labels_pred = [0, 0, 1, 1, 2, 2]

>>> metrics.homogeneity_score(labels_true, labels_pred)
0.66...

>>> metrics.completeness_score(labels_true, labels_pred)
0.42...
```

Their harmonic mean called **V-measure** is computed by `v_measure_score`:

```
>>> metrics.v_measure_score(labels_true, labels_pred)
0.51...
```

This function's formula is as follows:

$$v = \frac{(1 + \beta) \times \text{homogeneity} \times \text{completeness}}{(\beta \times \text{homogeneity} + \text{completeness})}$$

Backup

2.3.10.5. Silhouette Coefficient

If the ground truth labels are not known, evaluation must be performed using the model itself. The Silhouette Coefficient (`sklearn.metrics.silhouette_score`) is an example of such an evaluation, where a higher Silhouette Coefficient score relates to a model with better defined clusters. The Silhouette Coefficient is defined for each sample and is composed of two scores:

- **a**: The mean distance between a sample and all other points in the same class.
- **b**: The mean distance between a sample and all other points in the *next nearest cluster*.

The Silhouette Coefficient s for a single sample is then given as:

$$s = \frac{b - a}{max(a, b)}$$

The Silhouette Coefficient for a set of samples is given as the mean of the Silhouette Coefficient for each sample.

2.3.10.7. Davies-Bouldin Index ¶

If the ground truth labels are not known, the Davies-Bouldin index (`sklearn.metrics.davies_bouldin_score`) can be used to evaluate the model, where a lower Davies-Bouldin index relates to a model with better separation between the clusters.

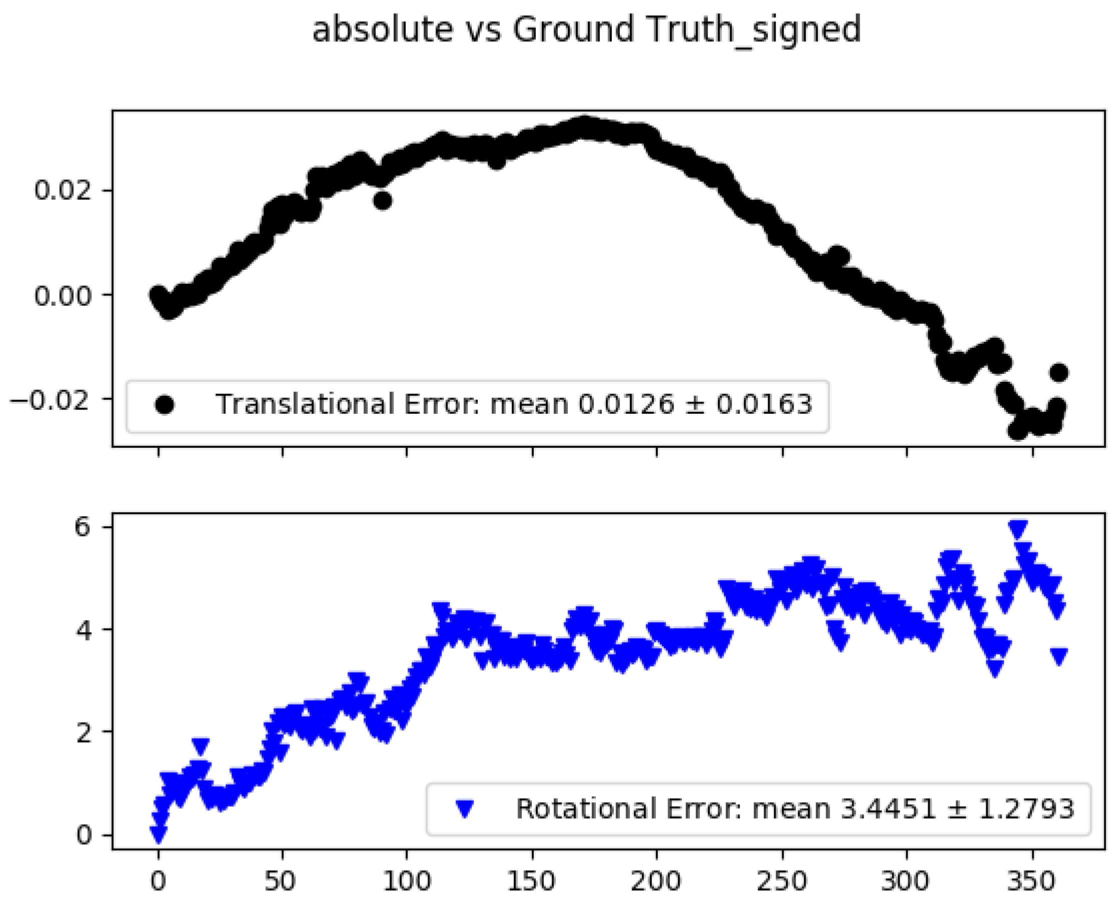
This index signifies the average 'similarity' between clusters, where the similarity is a measure that compares the distance between clusters with the size of the clusters themselves.

Zero is the lowest possible score. Values closer to zero indicate a better partition.

In normal usage, the Davies-Bouldin index is applied to the results of a cluster analysis as follows:

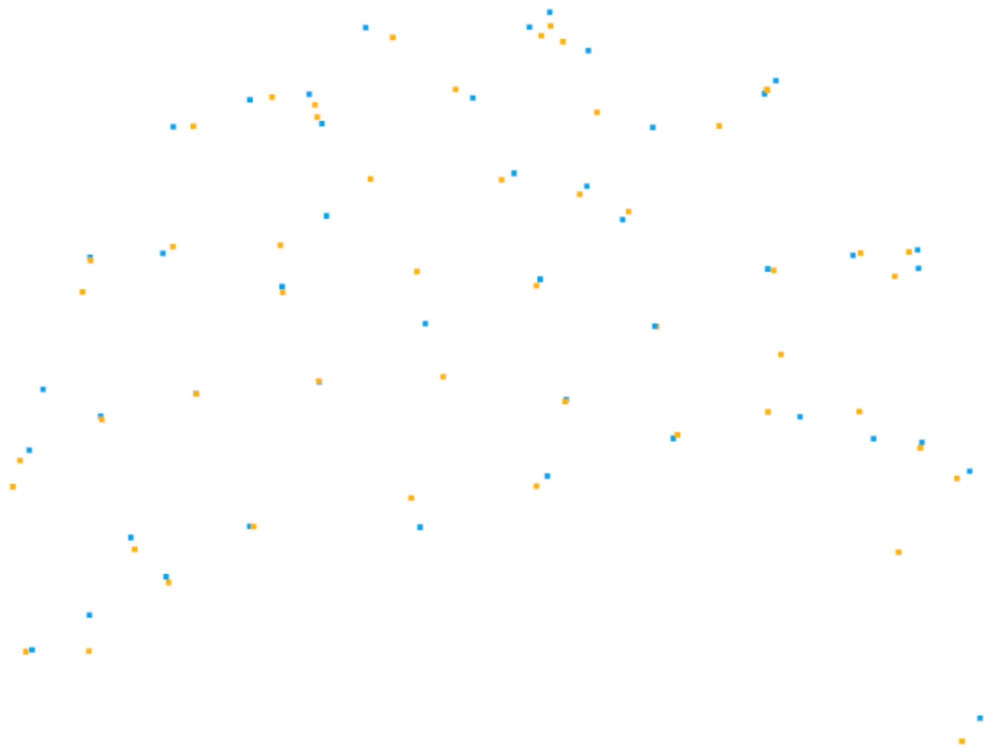
Results: YOLO + Depth → SLAM Cap: CA_124_7 (63 electrodes)

- Absolute trajectory vs ground truth trajectory



Results: YOLO + Depth → SLAM Cap: CA_124_7 (63 electrodes)

- Cluster center point cloud registration



RMSE after registration: 0.004 meters