FSM: Correspondenceless scan-matching of panoramic 2D range scans

Alexandros Filotheou, Antonis Dimitriou & Georgios Sergiadis

Aristotelian University of Thessaloniki (AUTh), Greece

ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

Summary

- It is possible to match 2D range scans **without establishing correspondences** *if their FOV is* 2π rad

Summary

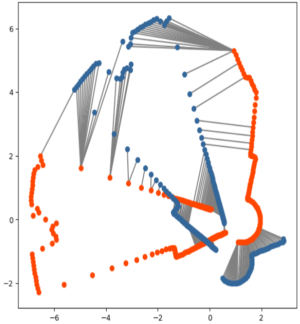
- It is possible to match 2D range scans **without establishing correspondences** *if their FOV is* 2π rad
- Leads to increased accuracy & robustness in the face of sensor noise

Summary

- It is possible to match 2D range scans **without establishing correspondences** *if their FOV is* 2π rad
- Leads to increased accuracy & robustness in the face of sensor noise and fewer parameters to be set

Why dispense with correspondences?

Euclidean Space | $\mathcal{S}_{\text{ref}}, \mathcal{S}_{\text{sens}}$

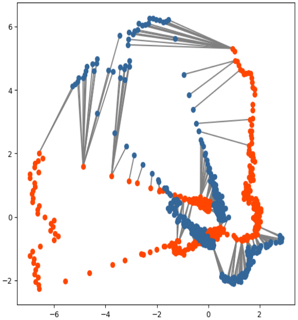


Correspondences' space

Figure: Matching with correspondences in ideal conditions. Sensor noise: $\mathcal{N}(0,0)$ [m,m^2]. Final estimation errors: 0.0023 m, 0.006 rad. Adapted from `https://nbviewer.org/github/niosus/notebooks/blob/master/icp.ipynb`, courtesy of Igor Bogoslavskyi

Why dispense with correspondences?

Euclidean Space | $\mathcal{S}_{\text{ref}}, \mathcal{S}_{\text{sens}}$



Correspondences' space

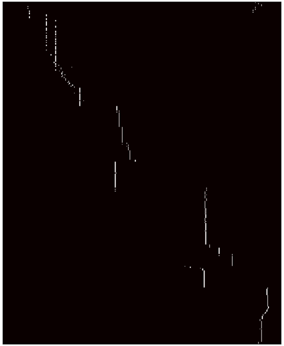


Figure: Matching with correspondences in real conditions. Sensor noise: $\mathcal{N}(0, 0.10^2)$ [m,m^2]. Final estimation errors: 0.035 m, 0.011 rad. Adapted from `https://nbviewer.org/github/niosus/notebooks/blob/master/icp.ipynb`, courtesy of Igor Bogoslavskyi

Why dispense with correspondences?

Euclidean Space | $\mathcal{S}_{\text{ref}}, \mathcal{S}_{\text{sens}}$

Correspondences' space

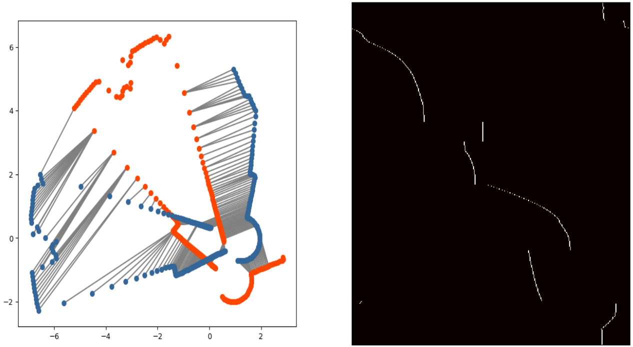


Figure: Matching with correspondences in real conditions, with outliers/void correspondences. Sensor noise: $\mathcal{N}(0, 0.10^2)$ [m,m^2]. Adapted from `https://nbviewer.org/github/niosus/notebooks/blob/master/icp.ipynb`, courtesy of Igor Bogoslavskyi

Why dispense with correspondences?

Ultimately:

- Due to sensor noise (higher as sensor cost decreases)
- Void (and false) correspondences' rejection is based on offline- and user-set parameters

More details:
Filotheou, A. et al. "Passive Global Localisation of Mobile Robot via 2D Fourier-Mellin Invariant Matching".
In *Journal of Intelligent & Robotic Systems* 104 (2022). https://doi.org/10.1007/s10846-021-01535-7

Why dispense with correspondences?

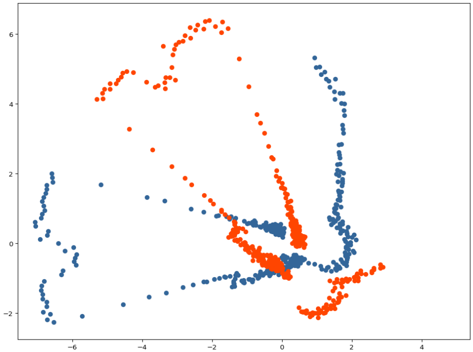


Figure: Matching without correspondences in real conditions, with outliers/void correspondences. Sensor noise: $\mathcal{N}(0, 0.10^2)$ [m,m^2]. Final estimation errors: 0.018 m, 0.0008 rad

The Fourier Scan Matcher (FSM)

- Operates on panoramic scans (FOV $= 2\pi$ rad)
- Does not deal in correspondences between inputs
- Requires minimal (if any) tuning

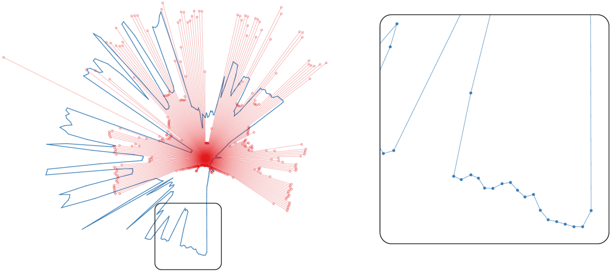
The Fourier Scan Matcher (FSM)

Operating principle:
Scan–to–map-scan matching

- $\mathcal{S}_{\text{ref}}$ projected to 2D plane $\Rightarrow$ map $\boldsymbol{M}$
- $\mathcal{S}_{\text{sens}}$ matched against scans derived within $\boldsymbol{M}$ (so-called *map-scans*)

Correspondenceless solution to orientation estimation

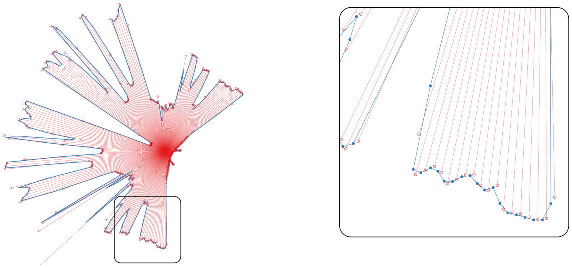
- Via 1D Phase-only Matched Filtering (error $\leq \dfrac{\text{angle increment}}{2}$)



$\Delta\theta = 1.20000$ rad

Correspondenceless solution to orientation estimation

- Via 1D Phase-only Matched Filtering + angular oversampling (error $\leq \frac{\text{angle increment}}{2^{1+\nu}}$)

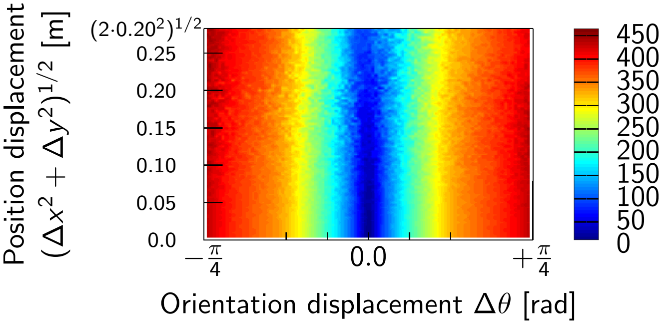


$\Delta\theta = -0.00428$ rad

Correspondenceless solution to orientation estimation

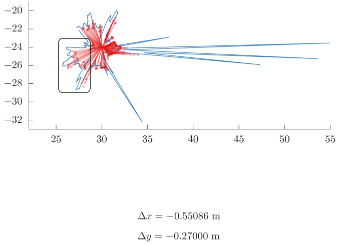
Final orientation estimate $\Leftarrow$ Rehearse translation estimation for 2^ν pose estimates $\hat{\boldsymbol{p}}_i$, $i = 0, 1, \ldots, 2^\nu - 1$, and rank

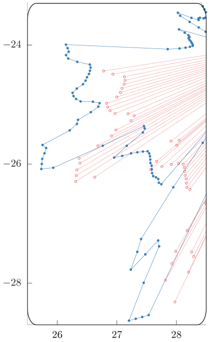
$$\text{CAER}(\mathcal{S}_{sens}(\boldsymbol{p}), \mathcal{S}^{\boldsymbol{M}}(\hat{\boldsymbol{p}}_i)) = \frac{1}{N_s} \sum_{n=0}^{N_s-1} \Big| \mathcal{S}_{sens}(n)|_{\boldsymbol{p}} - \mathcal{S}^{\boldsymbol{M}}(n)|_{\hat{\boldsymbol{p}}_i} \Big|$$



Correspondenceless solution to translation estimation

- Continuous space solution; feedback of DFT difference between scans

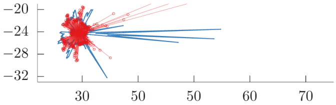




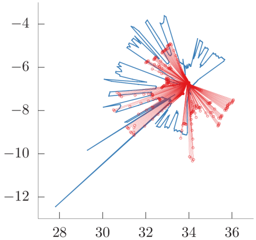
More details:

Filotheou, A. "Correspondenceless scan-to-map-scan matching of homoriented 2D scans for mobile robot localisation". In *Robotics and Autonomous Systems* 149 (2022). `https://doi.org/10.1016/j.robot.2021.103957`

FSM alignment progress and properties

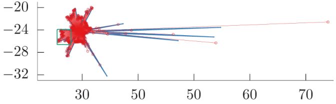


$\Delta x = -0.03252$ m
$\Delta y = 0.03603$ m
$\Delta\theta = -0.35689$ rad

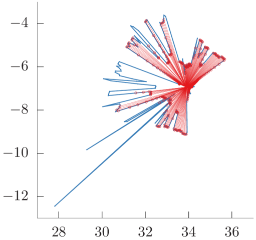


$\Delta x = -0.02856$ m
$\Delta y = -0.19805$ m
$\Delta\theta = -1.31500$ rad

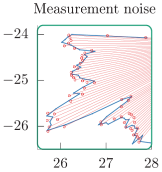
FSM alignment progress and properties



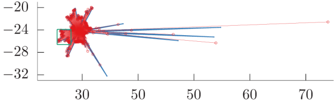
$\Delta x = -0.00085$ m
$\Delta y = 0.00337$ m
$\Delta \theta = -0.00346$ rad



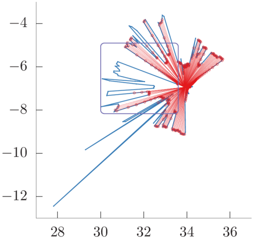
$\Delta x = 0.00643$ m
$\Delta y = 0.00371$ m
$\Delta \theta = 0.00194$ rad



FSM alignment progress and properties

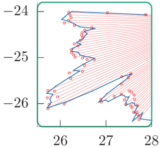


$\Delta x = -0.00085$ m
$\Delta y = 0.00337$ m
$\Delta \theta = -0.00346$ rad

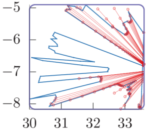


$\Delta x = 0.00643$ m
$\Delta y = 0.00371$ m
$\Delta \theta = 0.00194$ rad

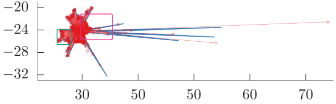
Measurement noise



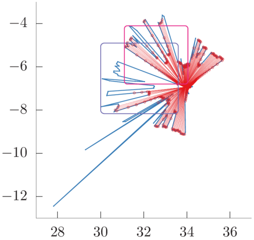
Void correspondences



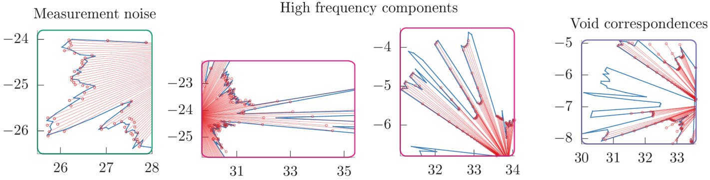
FSM alignment progress and properties



$\Delta x = -0.00085$ m
$\Delta y = 0.00337$ m
$\Delta \theta = -0.00346$ rad



$\Delta x = 0.00643$ m
$\Delta y = 0.00371$ m
$\Delta \theta = 0.00194$ rad



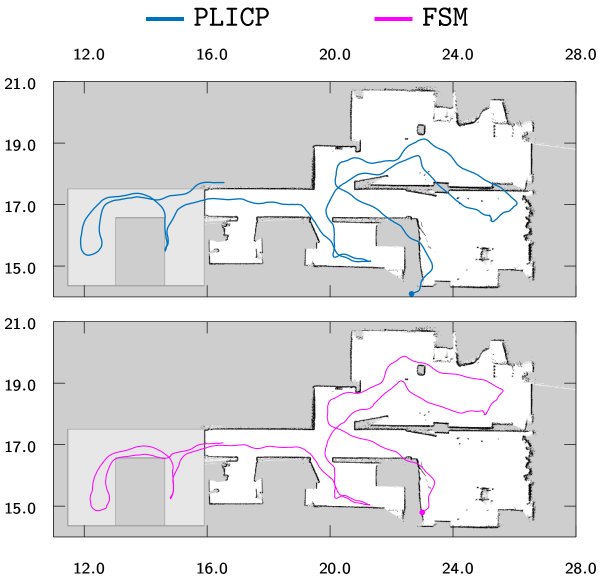
Laser odometry comparison

Sensor: YDLIDAR TG30

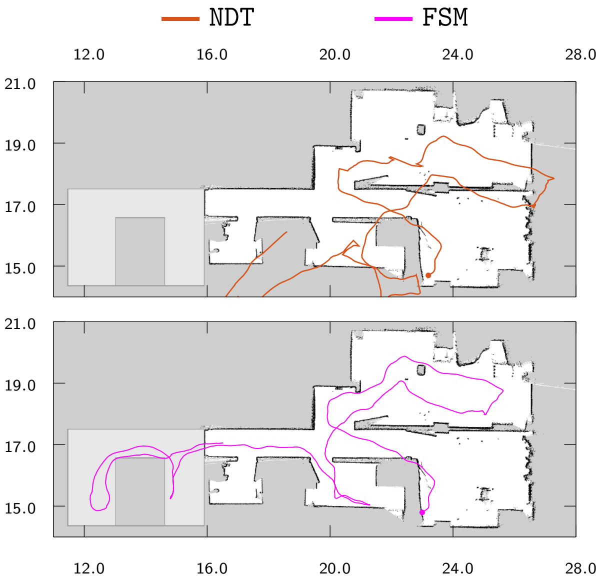
Range d [mm]	Mean error [mm]
50-5000	$\leq \pm 60$
5000-20000	$\leq \pm 40$
20000-30000	$\leq \pm 100$

Table: Sensor noise properties (knowledge of distribution N/A). Source: `www.ydlidar.com/Public/upload/files/2022-06-21/YDLIDAR%20TG30%20Data%20Sheet%20V1.4(211230).pdf`

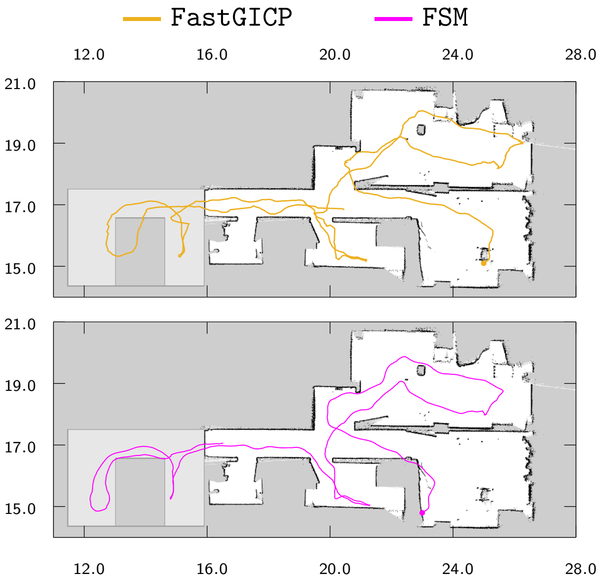
Laser odometry comparison



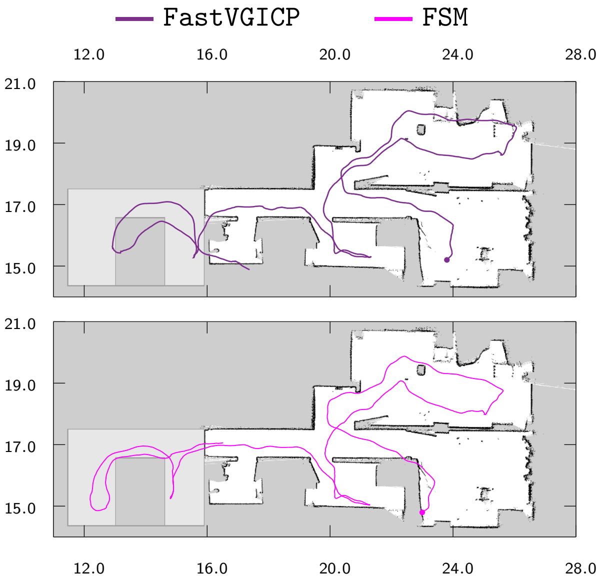
Laser odometry comparison



Laser odometry comparison



Laser odometry comparison



FSM: Correspondenceless scan-matching of panoramic 2D range scans

Thank you for your attention

- Presentation available at `https://github.com/li9i/fsm_presentation_iros22`
- Code available at `www.github.com/li9i/fsm`