Direct Embodied Data for Localisation and Mapping

Jakob Schwendner

DFKI Bremen & Universität Bremen
Robotics Innovation Center
Director: Prof. Dr. Frank Kirchner
www.dfki.de/robotics
robotics@dfki.de

Outline

Problem/Approach

Method

- Overview
- System/Contact Point Model
- Odometry Model
- Environment Model
- Measurement Model
- Particle Filter

Localisation Results

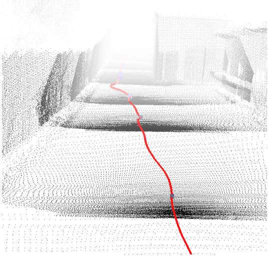
SLAM current state

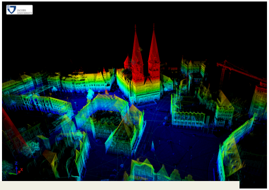
Conclusion/Outlook

Problem/Approach

Localisation and Mapping in 3D Environments

- Localisation
- Mapping
- Combined Localisation and Mapping
- Visual SLAM works

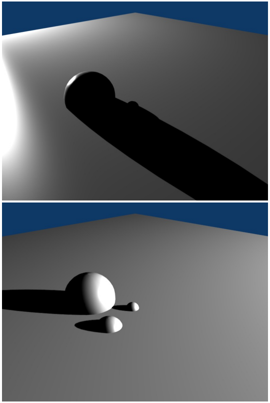




Problem/Approach

Motivation for using embodied data

- Could be used in blind scenarios
- Augment visual means of Localisation and Mapping
- Reduce requirements for vision
- Acknowledge the fact that robots have bodies, too
- Things are not always what they look like

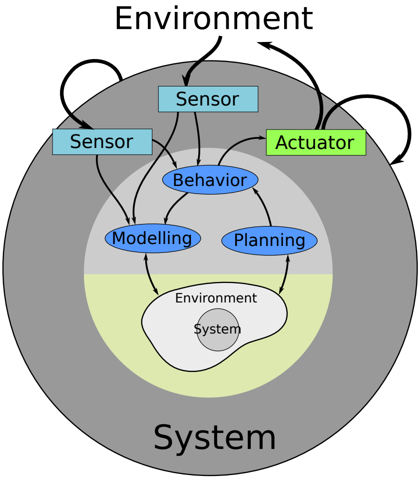


Problem/Approach

Embodied Data in Context

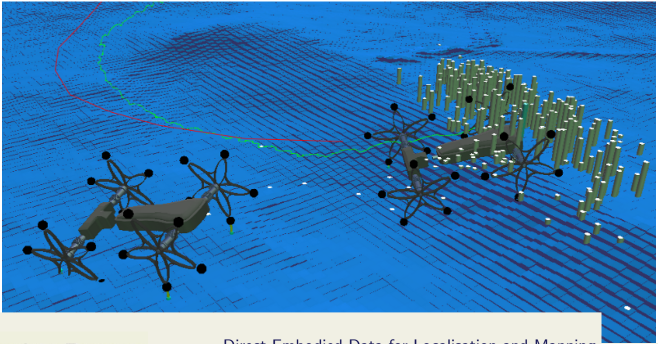
Embodied Data is defined as sensory information originated within or on the border of the system in question.

Two categories of Embodied Data: *Direct* and *Indirect*.



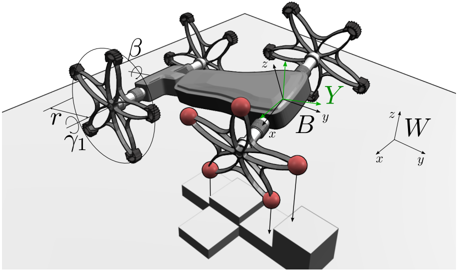
Method Overview

- System/Contact Point Model
- Odometry Model
- Environment Model
- Measurement Model
- Particle Filter



Asguard System/Contact Point Model

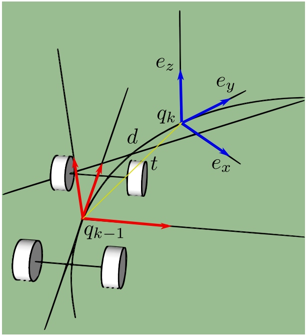
- Asguard has five degrees of freedom
- Four Wheels, free body joint
- $c \in C = (\gamma_1, \ldots, \gamma_4, \beta)$
- Contact with environment mainly through feet
- Modeling of Contact Points based on c and orientation q
- Frames W, B and Y



Method

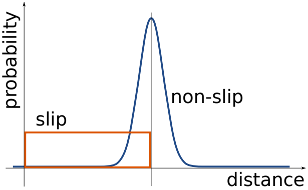
Asguard Odometry Model used for Approach

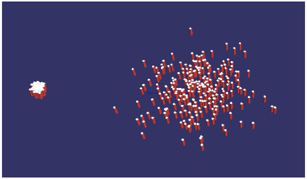
- Extended 2D skid steering to 3D
- Difference in orientation from IMU
- Travelled distance from wheel turns
- compensation for center of rotation



Odometry Error Model

- Mixture model
- Gaussian is with covariance $A(d, tilt, \Delta\theta, 1)^T$
- constant part for modeling slip
- Projection to Y frame

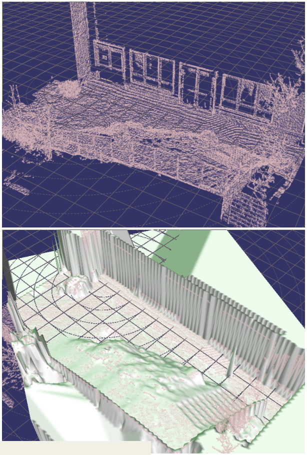




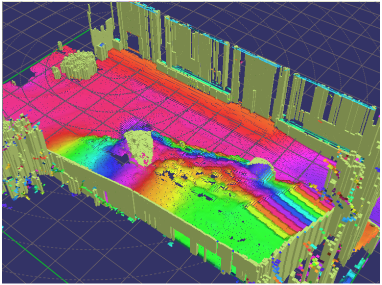
3D Environment Model

- Requirements: Cartesian, fast, handles Test-Track
- Modes: A-priori & Live
- Options

Pointcloud	simple, accurate, slow
DEM	simple, high information-loss, very fast
MLSM	more complex, medium information-loss, fast



Multi Level Surface Maps



- Regular grid cells partitioning xy-plane
- Multiple patches per cell
- Two cell types

  horizonal patch with μ, σ

  vertical patch μ, σ, h

- $m(p, l) =$

$$\begin{cases} (z, \sigma) & \text{surface with } z \in [p_z - l/2, p_z + l/2] \\ \emptyset & \text{no surface in interval} \end{cases}$$

Method

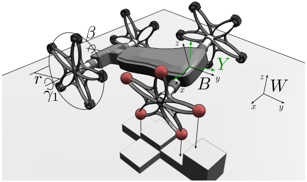
Single wheel contact estimation

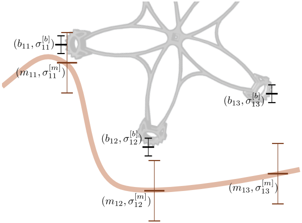
IMU (q), encoder readings (c)
Particle pose (T)

For each wheel

- Contact points in W from T and c
- Remove unlikely contacts
- Pick contact with lowest z diff to map
- Wheel is valid if all feet have map value

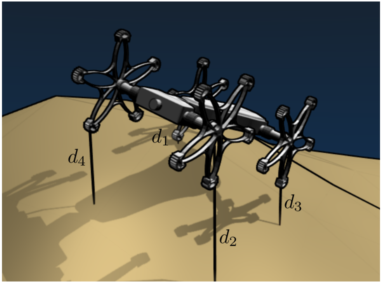
Not needed if contact information available





Robot body measurement

- $\hat{p}(z_k|m, c, T) = \prod_{(d,\sigma)\in z_k} \phi(\frac{d+\xi}{\sigma})$
- maximise for ξ to get z offset
- probability is not normalised yet



Particle Encoding and Measurement Normalisation

- Particle distribution over pose space (x, y, θ)
- Carries extra information (z, σ_z)
- measurement z_k and state x_k
- $\bar{p}$ discounted probability of found contacts
- $p(z_k|x_k^{[m]}) = \hat{p}(z_k|x_k^{[m]})\bar{p}^{4-|z_k|}$
- normalisation factors contacts per pose sample

Method

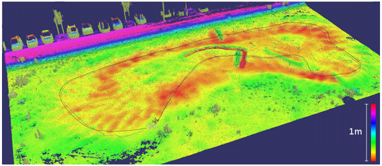
Description of Particle Filter used

- Sampling Importance Resampling (SIR) filter
- Initial particles created with given distribution
- Project particles using odometry
- Update particle weight based on $p(z_k|x_k)$
- Update particle z and σ
- Mark floating particles
- Resample if Effective Particle measure fall below threshold

Localisation Results

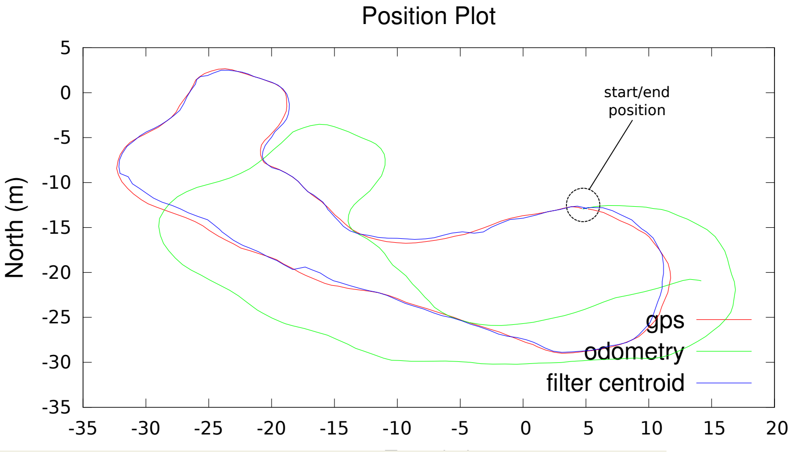
Sand Field Experiments

- 50 m x 30 m sand field
- height variation up to 1 m
- a-priori map
- grid spacing 0.05 m



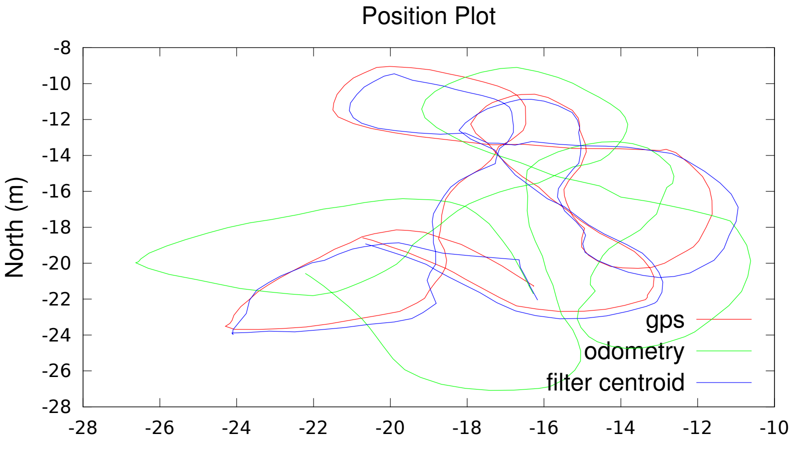
Localisation Results

Track lap (125 m)



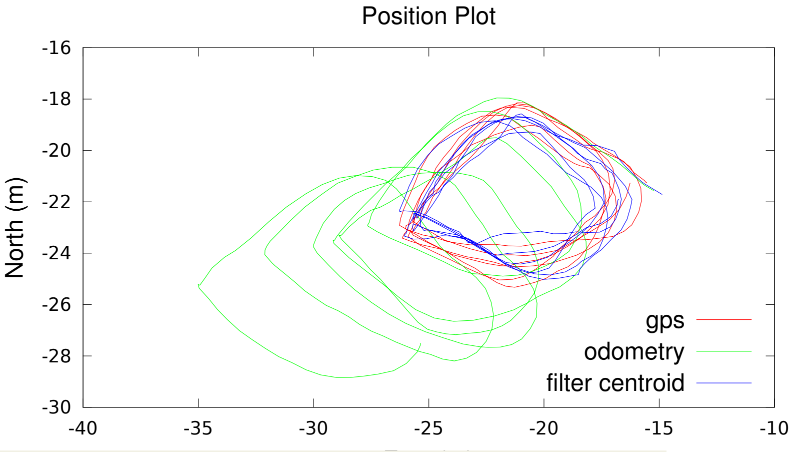
Localisation Results

Track cross (88 m)



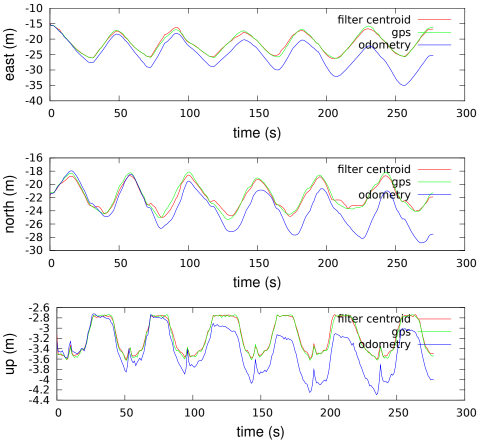
Localisation Results

Side Loop (143 m)



Localisation Results

Side Loop (143 m) vs time



Localisation Results

Total Distance Travelled

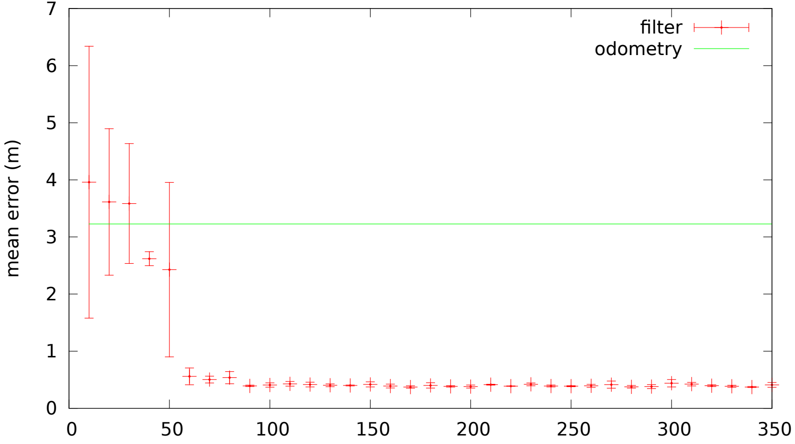
	Distance Travelled [m]		
Run	Centroid	Odometry	GPS
Lap1	125.83	141.97	125.19
Lap2	128.28	140.96	127.51
Lap3	124.81	135.85	123.85
Side Loop	136.84	161.63	143.89
Cross	89.67	100.31	88.46

Localisation Results

Position Error

	Mean Position Error [m]		Max Error [m]	
Run	Centroid	Odometry	Centroid	Odometry
Lap1	0.35	8.74	0.83	12.60
Lap2	0.37	9.34	1.06	12.92
Lap3	0.36	10.33	1.02	16.79
Side Loop	0.49	4.29	1.46	11.09
Cross	0.40	3.23	0.97	5.78

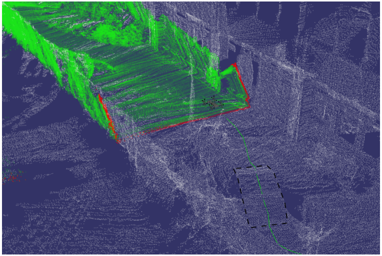
Error vs Particle Count



SLAM current state

Mapping

- Use Laserscanner for Mapping
- Uncertainty transformation into map
- One map per particle
- Work in progress ...



Conclusion/Outlook

Concluding the work and further steps

- Approach improves localisation over odometry alone
- Localisation filter has bounded error
- Should benefit from improved odometry
- Look into indirect embodied data
- Combine vision and embodied data to improve SLAM

Conclusion/Outlook

Thank you for your attention!