July 29, 2023

Survey

Abstract

Contents

1 ROS

2 Mapping

3 Localization

3.1 Dead Reckoning

3.2 KF - EKF

3.3 MCL

Likelihood

The best way to check the likelihood of a laser scan depends on the specific context and requirements of your application. However, here are a few commonly used methods for assessing the likelihood of a laser scan in the context of localization or mapping:

Comparison with Ground Truth:
If you have access to ground truth or reference data, you can compare the observed laser scan with the expected or ground truth scan. This can be done

by calculating a similarity metric such as point-to-point distance, point cloud matching, or scan matching algorithms like Iterative Closest Point (ICP). The closer the observed scan matches the ground truth, the higher the likelihood.

Statistical Analysis:
You can perform statistical analysis on the laser scan data to assess its likelihood. This can involve analyzing the distribution of scan points, such as calculating mean, variance, or higher-order statistical moments. Additionally, you can use techniques like hypothesis testing or density estimation to compare the observed scan distribution with a reference distribution.

Feature Extraction:
Extracting relevant features from the laser scan data and comparing them with expected or known features can help evaluate the likelihood. Features could include lines, corners, or other distinctive patterns in the scan. Techniques like feature matching or descriptor-based methods can be employed to compare the observed features with the expected ones.

Machine Learning Approaches:
Machine learning methods, such as classification or regression models, can be trained to estimate the likelihood of a laser scan. This requires labeled training data that includes scans labeled with their corresponding likelihood or quality. The trained model can then be used to predict the likelihood of new scans based on their features.
It's important to note that the best approach will depend on the specific requirements and constraints of your application. It's recommended to experiment with different methods and evaluate their performance in your specific context to determine the most suitable approach for assessing the likelihood of laser scans.

Codes Examples

Here are three methods to assess the likelihood of a laser scan based on range measurements:

Point-to-Point Distance Comparison: as shown in Source Code 1

```
1
2 import numpy as np
3
4 # Assuming you have two laser scans: observed_scan and
    ground_truth_scan
5
6 # Calculate the Euclidean distance between corresponding points
    in the scans
7 distances = np.abs(observed_scan - ground_truth_scan)
8
9 # Calculate the average distance or any other desired metric
10 average_distance = np.mean(distances)
11
12 # The lower the average distance, the higher the likelihood of
```

```
    the observed scan
```

Source Code 1: p2p code

Iterative Closest Point (ICP):
as shown in Source Code 2

```

import numpy as np
from sklearn.neighbors import NearestNeighbors
from scipy.spatial.transform import Rotation

# Assuming you have two laser scans: observed_scan and
    ground_truth_scan

# Find the nearest neighbors between the scans
nbrs = NearestNeighbors(n_neighbors=1, algorithm='kd_tree').fit(
    ground_truth_scan)
distances, indices = nbrs.kneighbors(observed_scan)

# Estimate the transformation between the scans using ICP or
    similar methods
transformation, _ = Rotation.align_vectors(observed_scan,
    ground_truth_scan[indices[:, 0]])
# The closer the transformation is to identity, the higher the
    likelihood of the observed scan
```

Source Code 2: ICP code

Point Cloud Matching:
as shown in Source Code 3

```


import numpy as np
from sklearn.metrics.pairwise import euclidean_distances

# Assuming you have two laser scans: observed_scan and
    ground_truth_scan

# Compute pairwise distances between points in the scans
distances = euclidean_distances(observed_scan, ground_truth_scan)

# Calculate the average distance or any other desired metric
average_distance = np.mean(distances)

# The lower the average distance, the higher the likelihood of
    the observed scan
```

Source Code 3: Point Cloud matching

Please note that these code snippets are general examples and may need modification based on the specific format and structure of your laser scan data.