

Protokoll – 1. AG Treffen

Bachelorpraktikum 2023 – Gruppe 43

07.11.2023

Verlauf:

1. Kurzes Kennenlernen
2. Projektvorstellung:
 - Projektkontext: Innotram, eine Tram die Daten per Lidar, GPS, Kameras etc. und perspektivisch autonom und mit intelligenter Kollisionsassistentz fahren soll.
 - Hierfür ist es notwendig, (halb)statische von dynamischen Objekten zu unterscheiden: statische Objekte wie Häuser, Straßenschilder, Wartehäuschen sind immer da und stellen keine Kollisionsgefahr da, während statische Objekte wie Fußgänger oder Autos eine Gefahr sind.
 - Teilweise werden allerdings statische Objekte wie die Wartehäuschen als Kollisionsgefahr erkannt.
 - Projektziel: Mithilfe der Lidar-Daten (Pointcloud) aus mehreren Fahrten dynamische von statischen Objekten unterscheiden, dynamische Objekte herausrechnen und eine Kartierung der statischen Objekte erstellen
3. Minimalziel:
 - Linux-Terminalprogramm, welches die Lidar-Daten als Input erhält und die Kartierung ohne dynamische Objekte ausgibt
4. Darüberhinausgehende Ziele:
 - Entwicklung eines effizienten Datenformats zum Speichern der Kartierung
 - GUI/Möglichkeit zur Visualisierung der Output-Karte
5. Weitere Details zum Projekt
 - Eigengeschwindigkeit der Bahn muss beachtet werden, verzerrt die Daten etwas
 - Erstmal nur Daten von einem Sensor, perspektivisch Daten aus bis zu 4 Sensoren
 - Rosbag/PCL Point Cloud Data als Input
 - Muss auf Ubuntu 20.04 laufen, perspektivisch auf 22.04
 - Keine Echtzeitanforderungen, auch wenn es nice to have wäre, wenn das Programm mit Cuda laufen würde
 - Daten sollten über LTE verschickbar sein
 - Visualisierung eines Rosbag z.B. mit Avis
 - Es gibt Daten zum Liniennetz der Schienen und GPS-Daten, damit lassen sich Daten mehrerer Bahnen/Fahrten überlagern
 - Wenn das Projekt nativ auf Ros2 laufen würde, wäre super
 - Programmiersprache: Primär C++, für zusätzliches (z.B. irgendwelche side packages) kann gegebenenfalls auch Python verwendet werden
 - Lizenz: eine Open-Source-Lizenz z.B. MIT, BSD, GPL sodass das Projekt auf einem öffentlichen GitLab veröffentlicht werden kann, genaue Lizenz muss noch festgelegt werden
6. Organisatorisches:
 - Repo im RWTH Gitlab ist bereits angelegt
 - Kommunikation über Element
 - Tool für schöne Sprintplanung gewünscht, z.B. Jira oder Taiga, Wahl obliegt der Gruppe
 - Sprintmeetings werden voraussichtlich zweiwöchentlich stattfinden

- Nächster Termin: Mittwoch, 14. November, 9 Uhr im selben Raum, Inhalt: kleine Technologieeinführung/Demo und User Stories schreiben, NDA für Daten
- Die nächsten Tage schickt Alexander noch beispielhafte Inputdaten/Rosbags und die genutzte Ros-Version, sodass sich die Gruppe ein bisschen einarbeiten kann

Aufgaben bis zum nächsten Treffen:

1. Namen für Gitlab, Element etc. an AG schicken
2. Einlesen/Einarbeiten in die Technologien
3. Gedanken über mögliche QS-Ziele machen