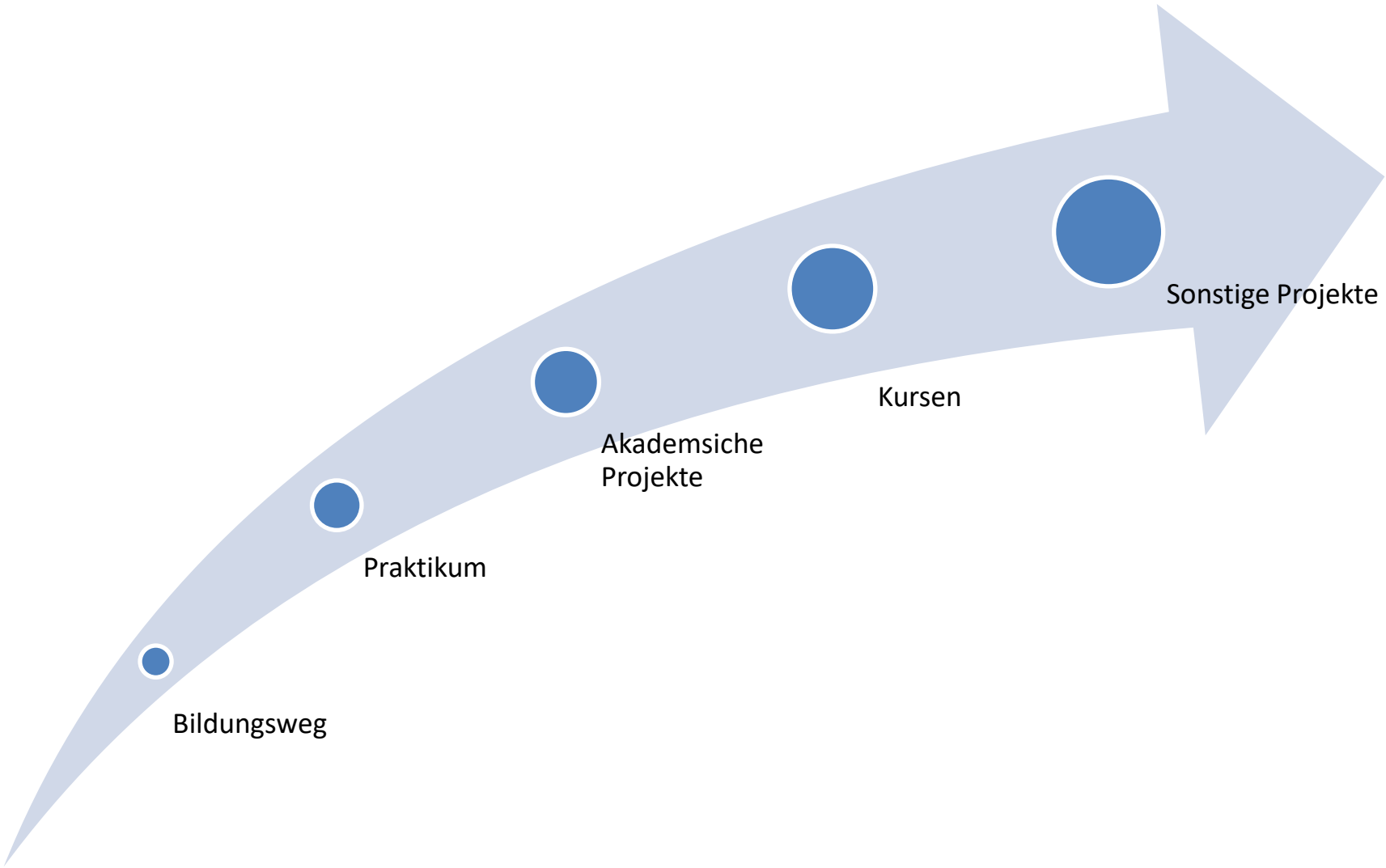
Portfolio

Roshan Sathyanarayana Shenoy

Agenda


Bildungsweg
Praktikum
Akademsiche Projekte
Kursen
Sonstige Projekte

Roshan Sathyanarayana Shenoy

Masters Student bei Technische Universität Kaiserslautern

Bildungsweg

Seit 10/19

Masters in Nutzfahrzeugtechnik, Vorläufige Noten: 2,9

Vertiefung:

Intelligente Systeme

Relevante Kursen:

2D Bildverarbeitung, Grundlagen des Software Engineering, Machine Learning I: Foundations, 3D Computer Vision

09/11 – 07/15

Bachelors in Maschinenbau, Deutsche Äquivalente Noten: 1,5

Relevante Kursen:

Engineering Mathematics, Computer Concepts and C Programming, Fahrzeugtechnik, Regelungstechnik

Bachelorarbeit:

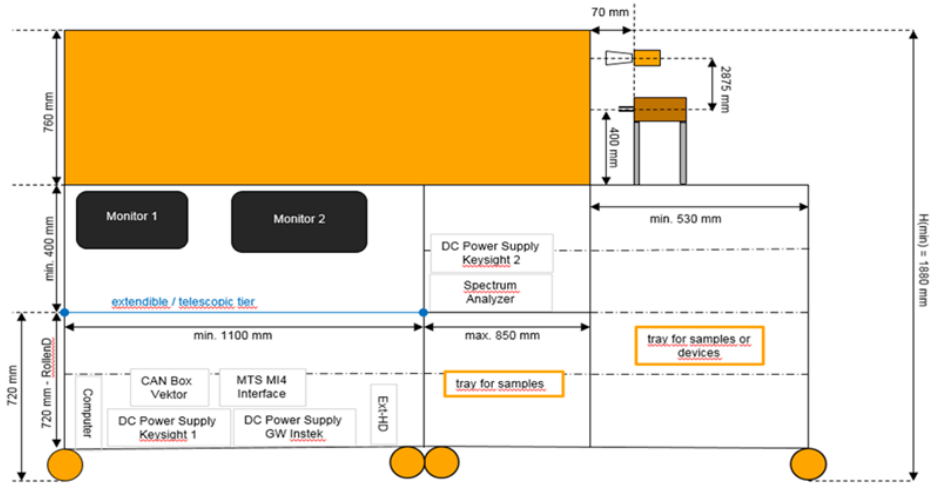
Effect of Boron Nitride on Mechanical, Friction and Wear Behaviour of Polyaryletherketone Composites.

Engagement und Erfolge

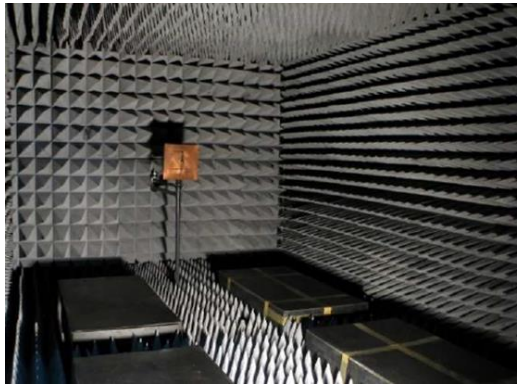
- Deutschland Stipendium 2019-20.
- ABCD (Above and Beyond the Call of Duty) Auszeichnung für vorbildlichen Beitrag zum Aufbau einer Belt-Routing-Kooperation.
- Patent zum Thema, Replacement of Femoral Metal Stem in Total Hip Arthroplasty with a Novel Composite Material beantragt (Patentanmeldungs-nummer: 5807/CHE/2015).
- Ein Forschungsbericht über "Optimization of wear behaviour of boron nitride filled polyaryletherketone composites by Taguchi approach" wurde am 31. Mai 2019 in Material Research Express veröffentlicht.

Praktikum (09/21 – 05/21) bei Continental ADC GmbH

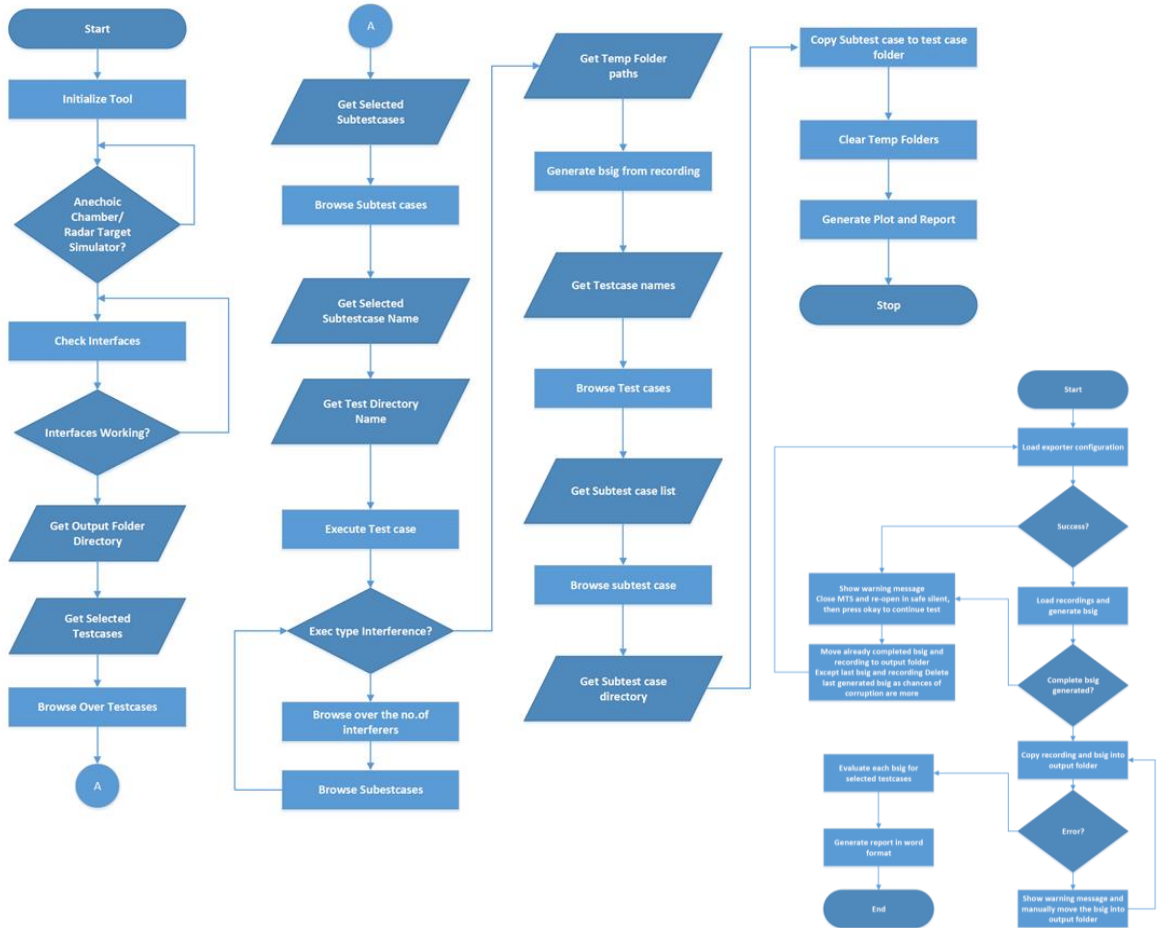
1. Test Automation Framework Entwicklung



RTS Blockschaltbild



Anechoic Chamber

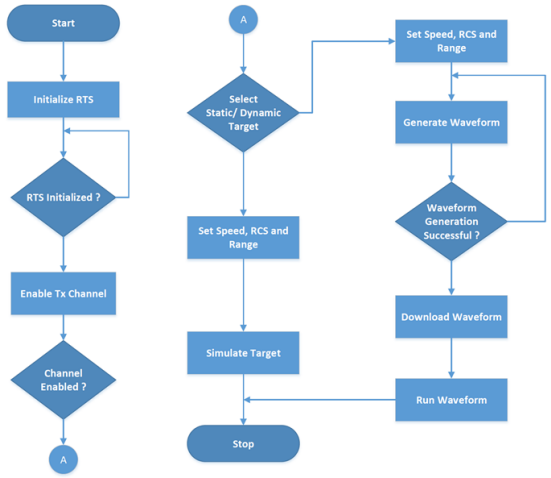


Test Exekution und Evaluation Flow Diagramm

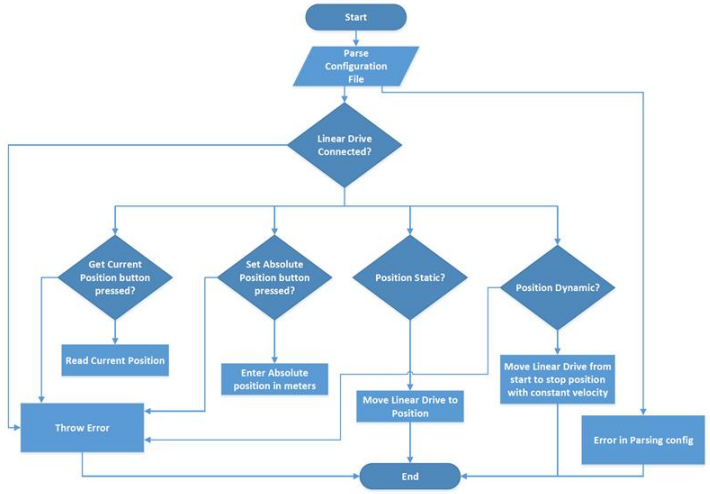
Praktikum (09/21 – 05/21) bei Continental ADC GmbH

2. Kontrollierung von Verschiedene Steuergeräte

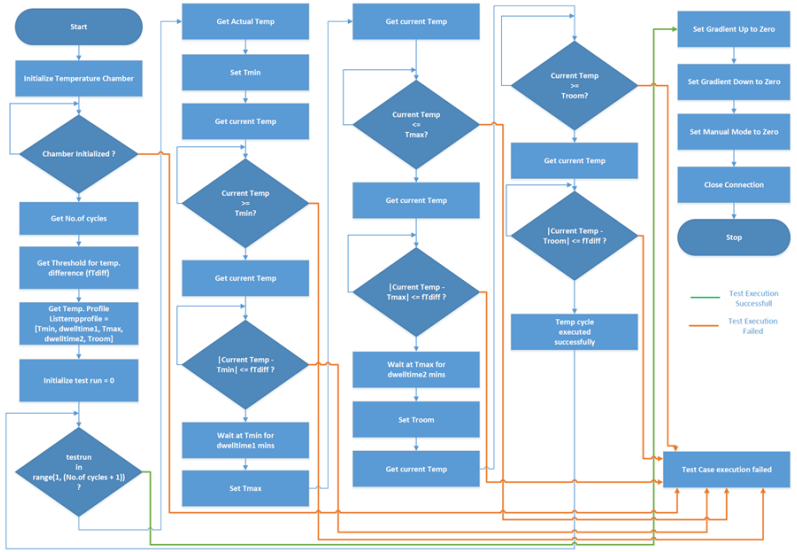
Radar Target Simulator



Lineare Antriebe



Temparatur Chamber

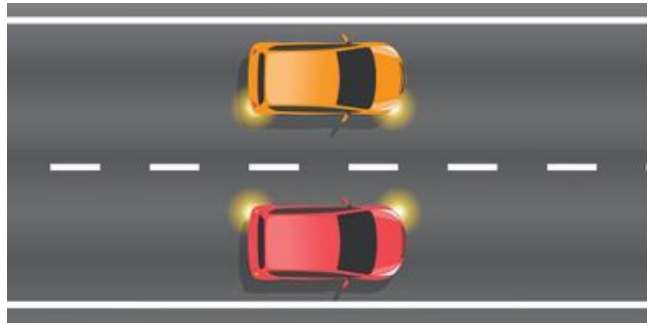


Drehmotoren

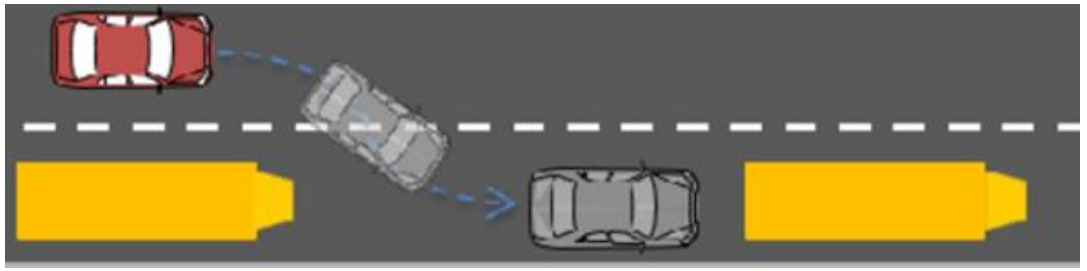
Datenerfassungssysteme

Praktikum (09/21 – 05/21) bei Continental ADC GmbH

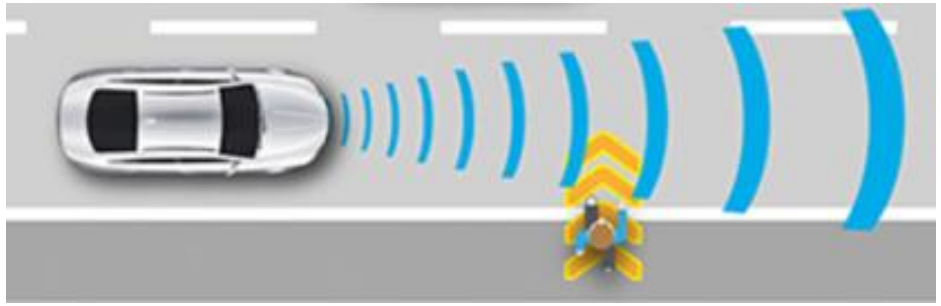
3. Integrierung von EURO NCAP Test Szenarian



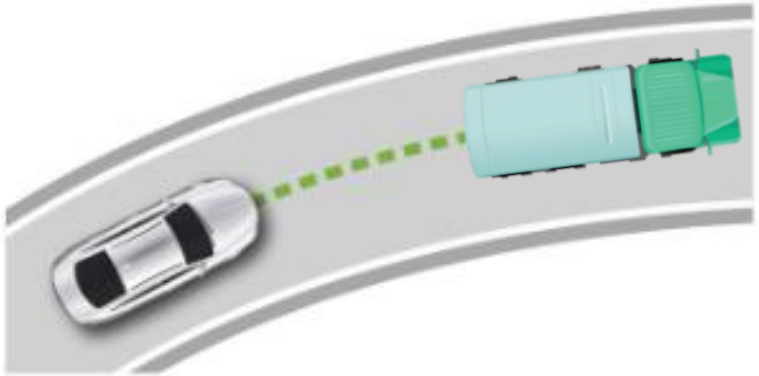
Parallel Fahren



Cut-in Fahren



Cross Moving Fahren



Kurve Fahren

- Python Bibliotheken: PyQt, Pyserial, ctypes, pandas, numpy und matplotlib
- MATLAB
- CANoe, Esys

Akademische Projekte

Projekt

10/20 – 04/21

Environment perception, mapping and motion planning for Autonomous mobile robots

- Recherchierte der Verwendung von Dense-Objektdeskriptoren zur Erkennung von 1-D-Objekten für die Roboternavigation.
- Bewegungsplannung durch A star und sample based (Rapidly-exploring Random Tree) algorithmen.
- Simultaneous Lokalizierung und Mapping (SLAM) für die Schätzung von Roboterposition und Umgebung.

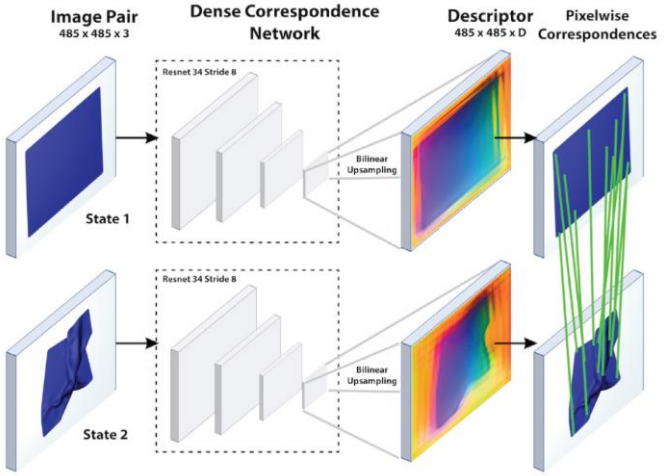
Aufgaben

- Detektion von 1D objekte wie kabeln zur roboter navigation über ein unbekanntes Terrain.
- Detektion von Kannten, Ecke und Topgraphie von objekt

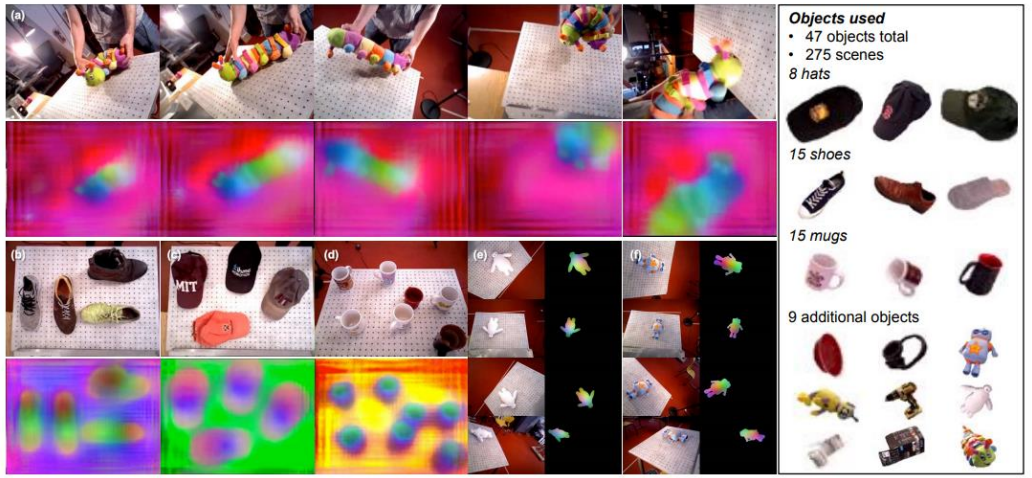
Tools Used

- Python Bibliotheken: pytorch, pandas, numpy and matplotlib

Images



Dense Descriptor generation pipeline



Gelernte Deskriptor Korrespondenz

Github link

https://github.com/rshn1994/pytorch-dense-correspondence.git

Projekt

Hearausforderungen

Toolkette

Images

Github link

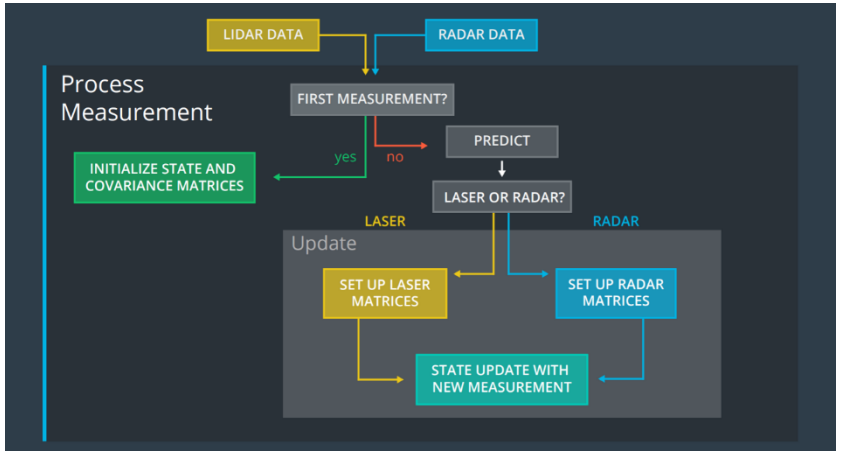
10/19 – 09/20

A Comparative Study: Sensor Fusion of LIDAR and RADAR using Kalman Filter and Extended Kalman Filter

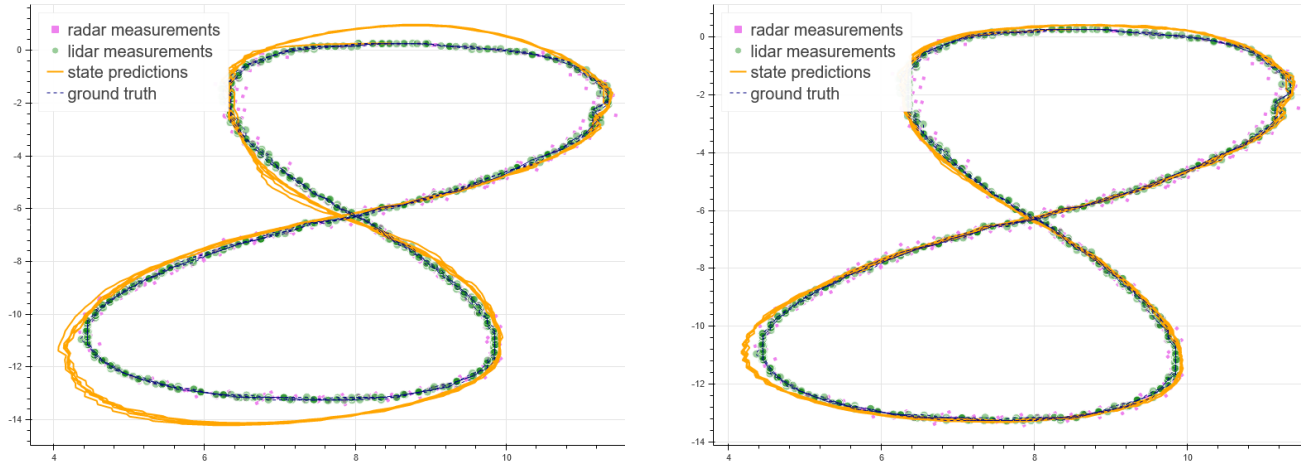
- untersuchte die Parameter, die die Auswahl von LIDARs und RADARs beeinflussen.
- Modelierung von Kalman Filter und Erweiterte Filter für abstand and geschwindigkeit schätzung.
- Vorhersage vergleich von Kalman Filter and Erweiterte Kalman Filter mit dem aktuelle ground truth werten.

- Problem bei der Umstellung von polaren in kartesische Koordinaten für RADAR.

- Python Bibliotheken: pandas, numpy und matplotlib



Kalman Filter Working Flowchart



Application of Kalman Filter on (left) and Extended Kalman Filter (Right) on RADAR and LiDAR

https://github.com/rshn1994/A-Comparative-Study-Sensor-Fusion-of-Lidar-and-Radar-using-Kalman-Filter-and-Extended-Kalman-Filter.git

Kursen

Machine Learning 1: Foundations

Projekt

04/21 – 09/21

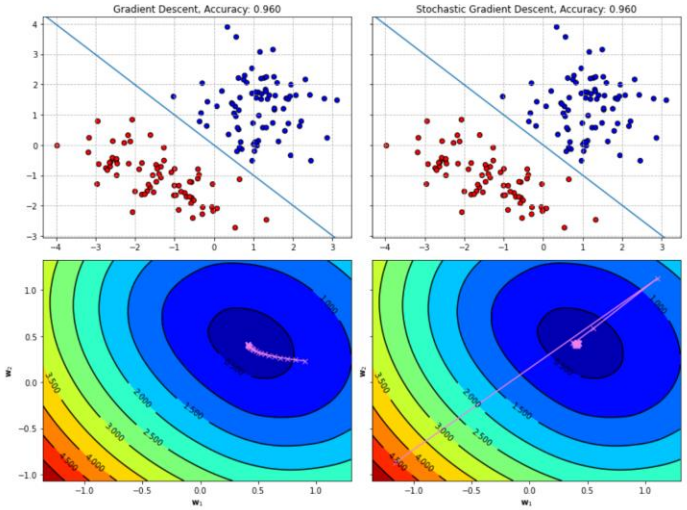
Programmierung Übungen

- Übung 1: Linear Support Vector Machine Reference implementation und Nearest Centroid Classifier.
- Übung 2: Gradient Descent und Linear Support Vector Machine Implementation.
- Übung 3: Multi Class und Kernel SVM Implementation.
- Übung 4: Artificial Neural Networks (LeNet-5 Architecture und ANN).
- Übung 5: K-Means und Dbscan clustering implementation.

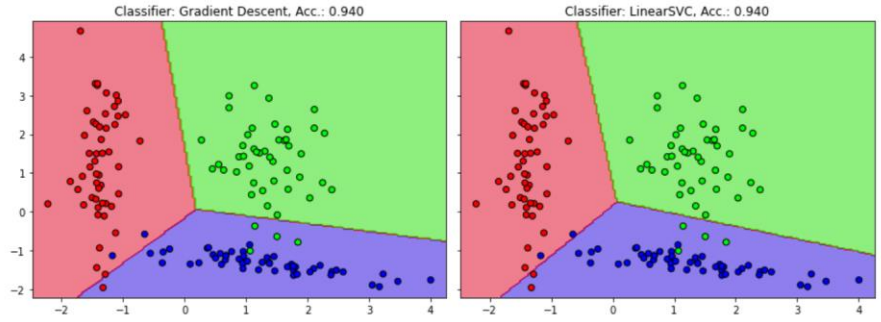
Toolkette

- Python Bibliotheken: scikit learn, pytorch, pandas, numpy, os, sys und matplotlib

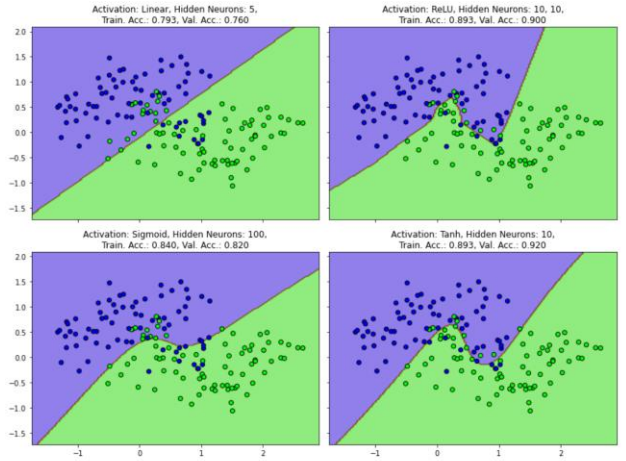
Ergebnisse



Gradient Descent



Multi Class SVM



Neural Network Classifier

Projekt

Toolkette

Ergebnissen

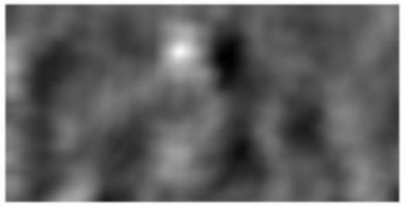
2D Bildverarbeitung

04/21 – 09/21

Programmierung Übungen

- Übung 1: Template Matching, Image smoothening, Image Noises, Fourier Transform, Integral Histogram.
- Übung 2: Canny Edge, Laplacian und Harris Point detector.
- Übung 3: Lucas Kanade Optical Flow.

- Python Bibliotheken: OpenCV, numpy, os und matplotlib



Template Matching

Image Noise

Kante und Ecke Detektoren

Projekt

Toolkette

Ergebnissen

3D Computer Vision

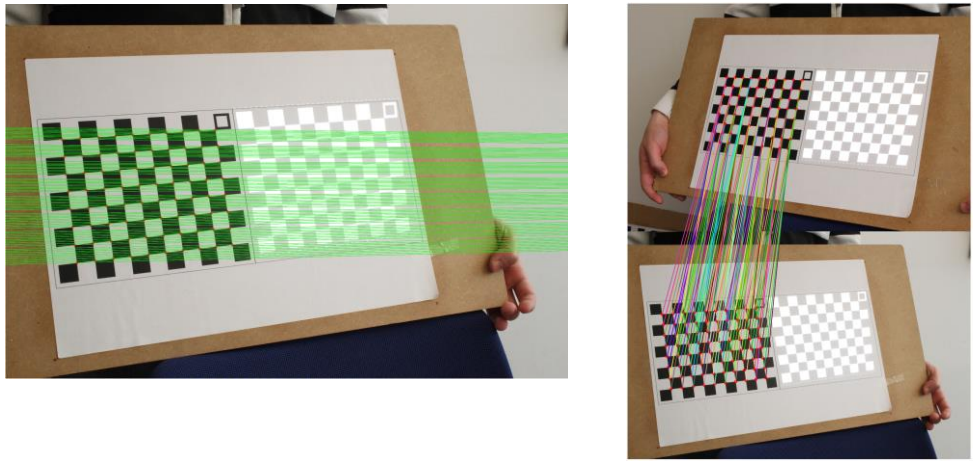
10/20 – 03/21

Programmierung Übungen

- Übung 1: Kamera Kalibrierung.
- Übung 2: Relativ Drehung und Pose Einschätzung vom Homography.
- Übung 3: Feature Matching und Struktur Rekonstruktion.

- Python Bibliotheken: OpenCV, scipy, numpy, os und matplotlib

Kamera Kalibrierung



Epilines und Feature Matches

Sonstige Projekte

Project Name

Radar Target Generation and Detection

Duration

May 2020 - May 2020

Objective

The aim of the project was to track multiple cars on a highway using Lidar and Radar

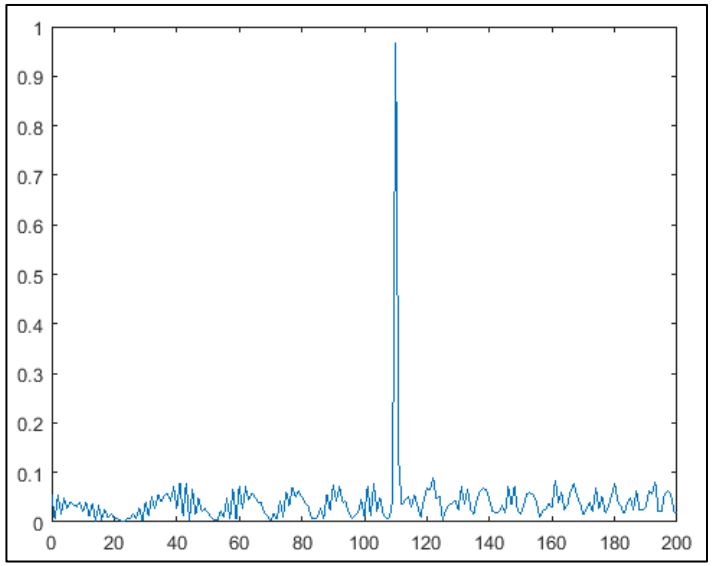
Highlights

- ❑ Developed a radar target generation and detection system by configuring the Frequency Modulated Continuous Wave (FMCW) radar waveform.
- ❑ Implemented Range/Doppler Fast Fourier Transform (FFT) and 2D Constant False Alarm Rate (CFAR) techniques for detecting position and velocity using MATLAB.

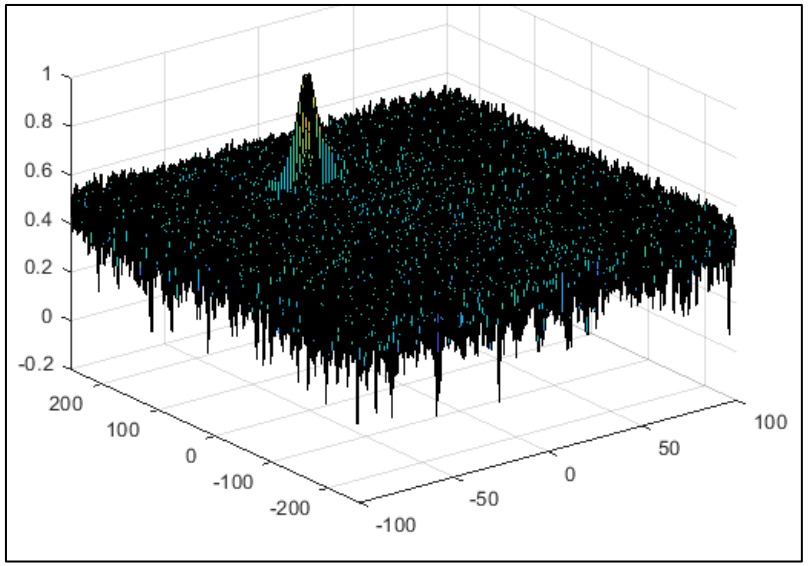
Software used

- ❑ MATLAB

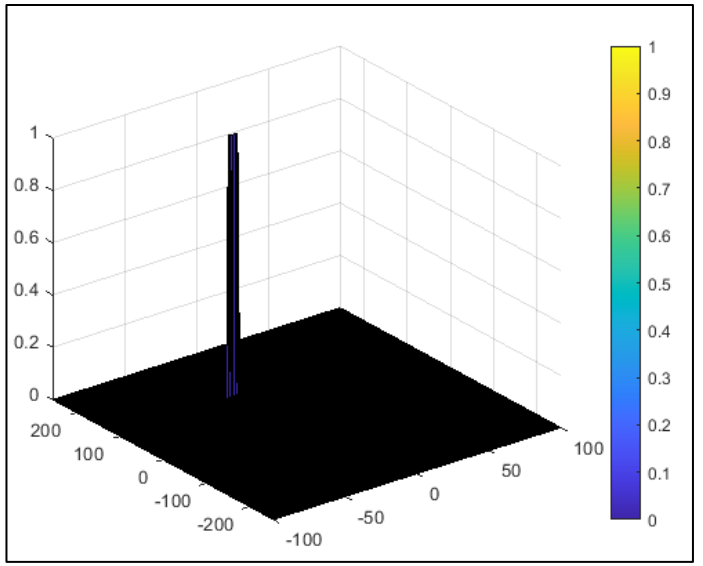
Images



The 1st FFT output for the target located at 110 m



2D FFT output - Range Doppler Map



The output of the 2D CFAR process

Project Name

Lidar Obstacle Detection

Duration

May 2020 - May 2020

Objective

The aim of the project was to detect cars and trucks on a narrow street using Lidar

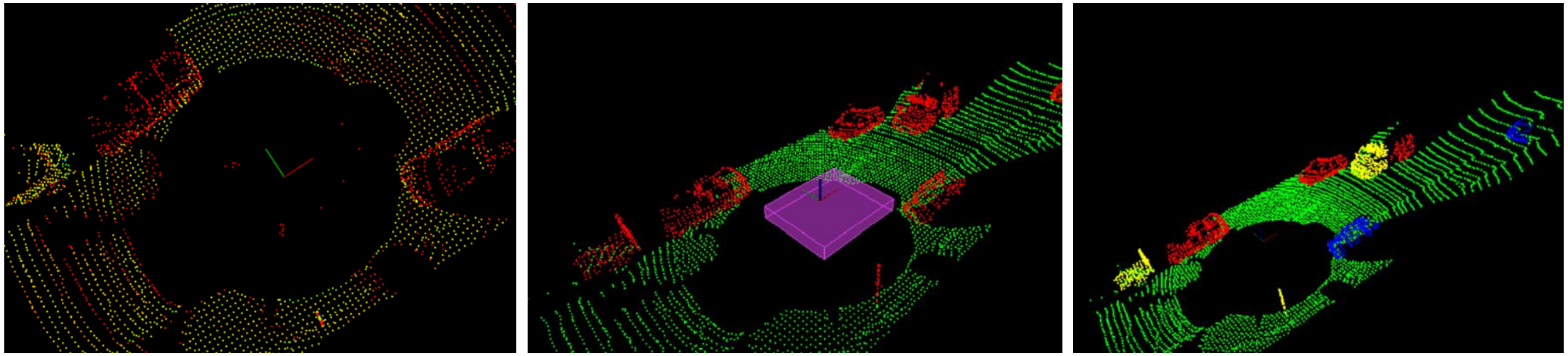
Highlights

- ❑ Applied voxel grid filtering technique to filter out the point clouds
- ❑ Used RANSAC algorithm to segment the point clouds
- ❑ Applied Euclidian clustering to group the point clouds into roads and obstacles
- ❑ Generated bounding boxes for the road and the obstacles
- ❑ Streaming of the point cloud data

Software used

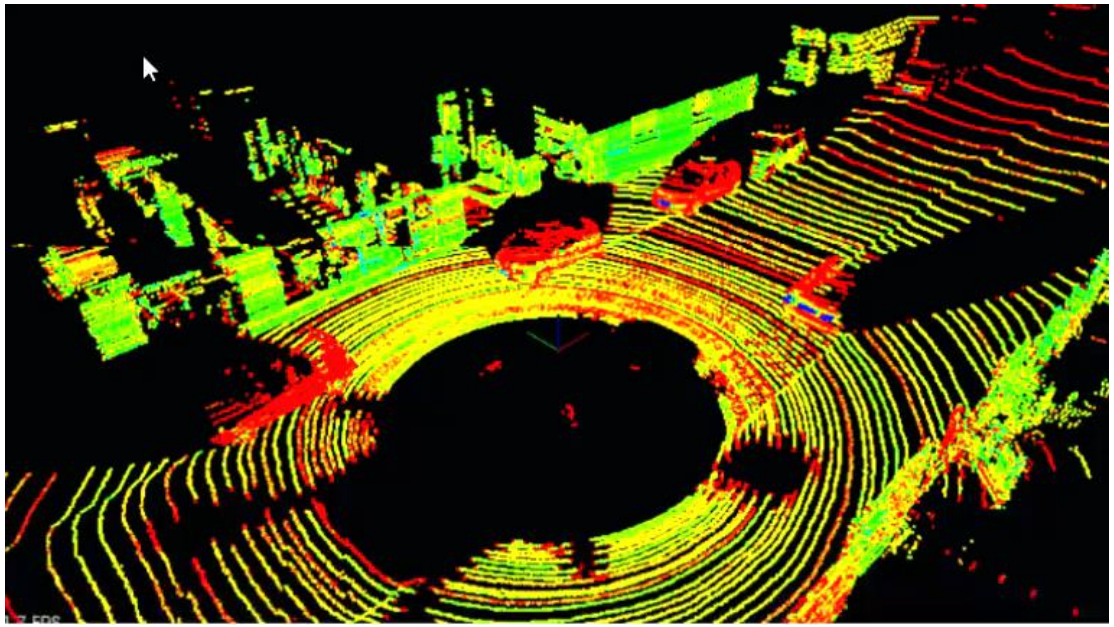
- ❑ C++

Images

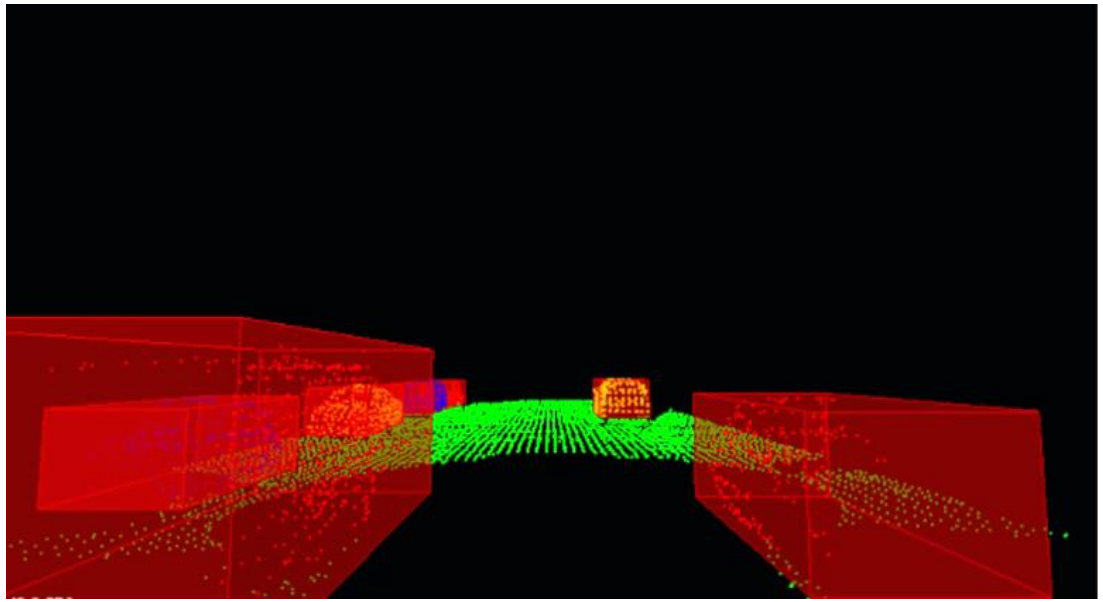


Filtered, segmented and clustered point cloud data

Videos



Sample stream of unprocessed point cloud data



Streaming von processed punktwolken daten

Project Name

Unscented Kalman Filter Project

Duration

May 2020 - May 2020

Objective

The aim of the project was to track multiple cars on a highway using Lidar and Radar

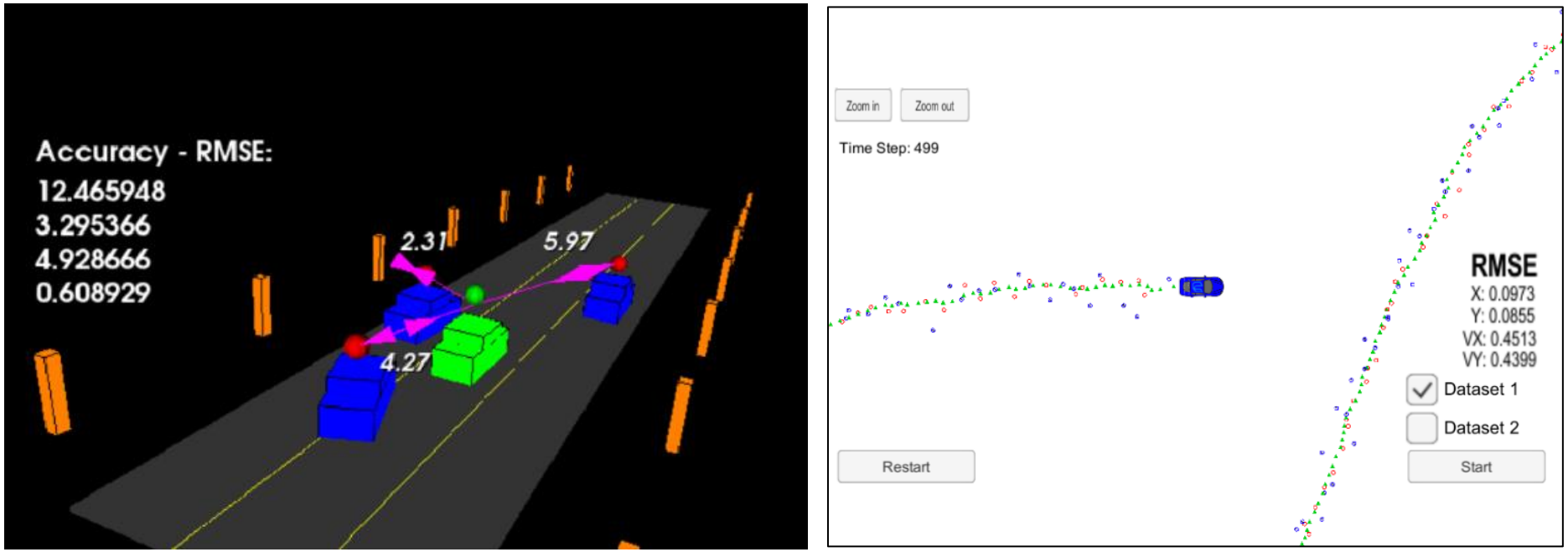
Highlights

- ❑ Each vehicle was tracked using a Constant Turn Rate Velocity motion model
- ❑ Used unscented transformation based on sigma points to approximate the mean and covariance for the real predicted distribution
- ❑ Predicted the position and velocity for the vehicles using Lidar and Radar measurements
- ❑ Checked if the RMS values for the position and velocity were below the threshold

Software used

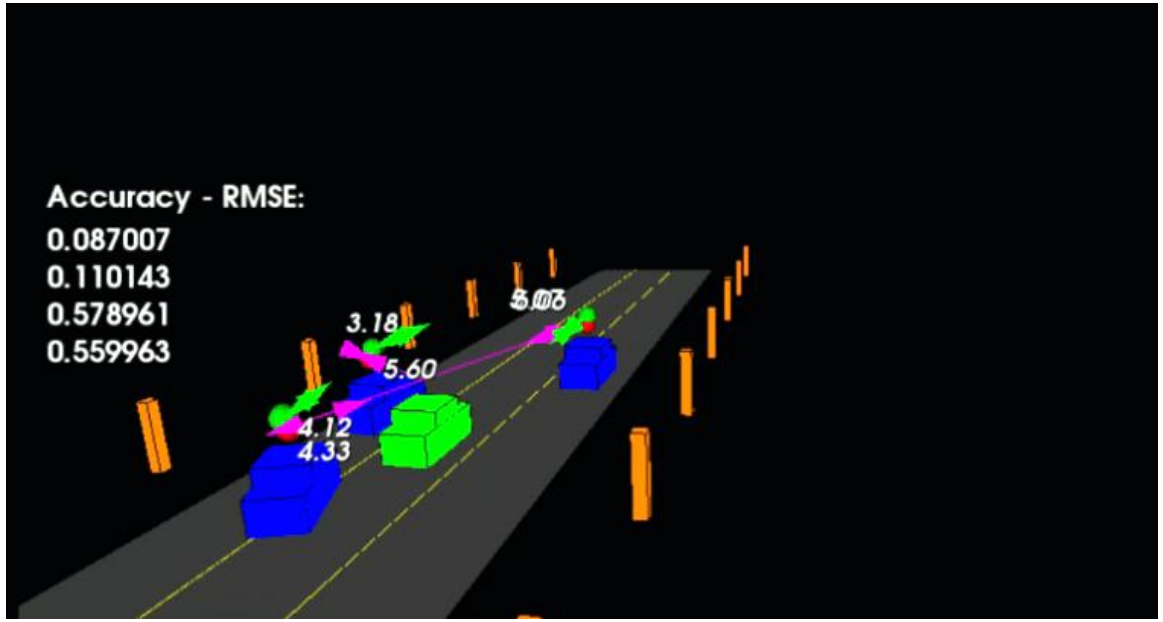
- ❑ C++

Images



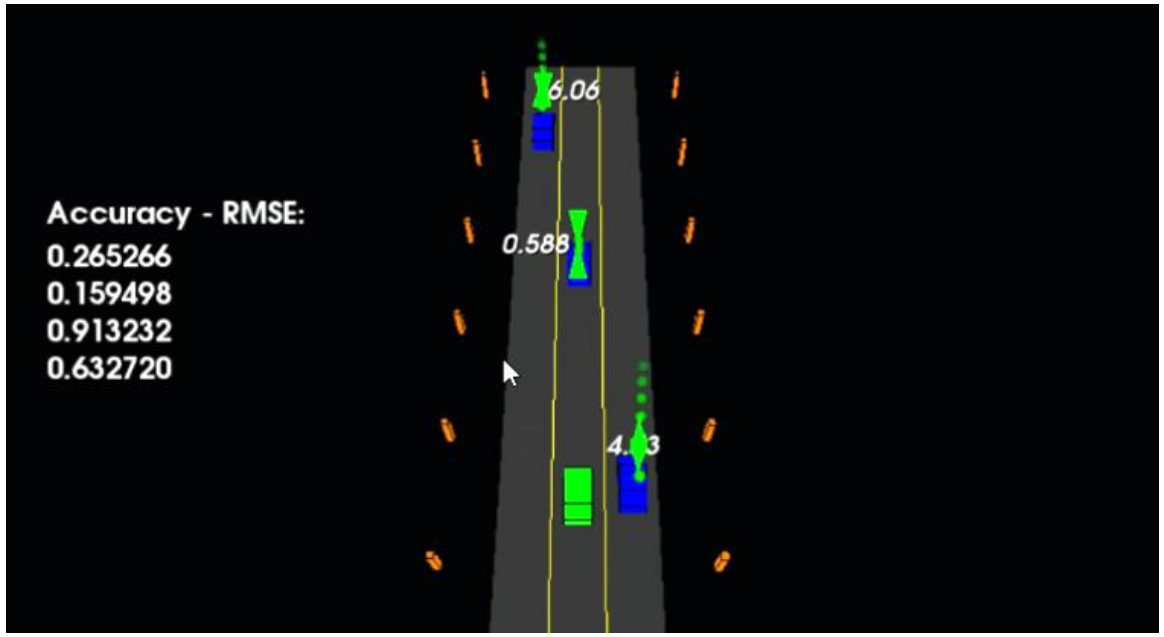
Tracking using Unscented Kalman Filter

Videos



All the cars are considered in the RMSE value so if one car is particularly inaccurate it would affect the overall RMSE in this case. In the animation the 4 RMSE values are X, Y, Vx, Vy from top to bottom.

Assumption:
RMSE should be less than or equal to the values [0.30, 0.16, 0.95, 0.70]



cars do not have constant turning rates you can see the predicted paths swing around and take a while to correct after the car begins moving straight again