Dokumentation

Timo Krehle

Deutsches Zentrum für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft
Institut für Verkehrssystemtechnik
Lilienthalplatz 7
38108 Braunschweig

Tel.: +49 (0)531 295- 3486
Fax: +49 (0)531 295- 3427
E-Mail: mailto:timo.krehle@dlr.de
Info: www.dlr.de/fs

Inhaltsverzeichnis

Inhaltsverzeichnis 2

Radarsimulation 3

Einbinden der Applikation in die JDVE 3
Spezifikation des Radars 3
Datenpaket 4

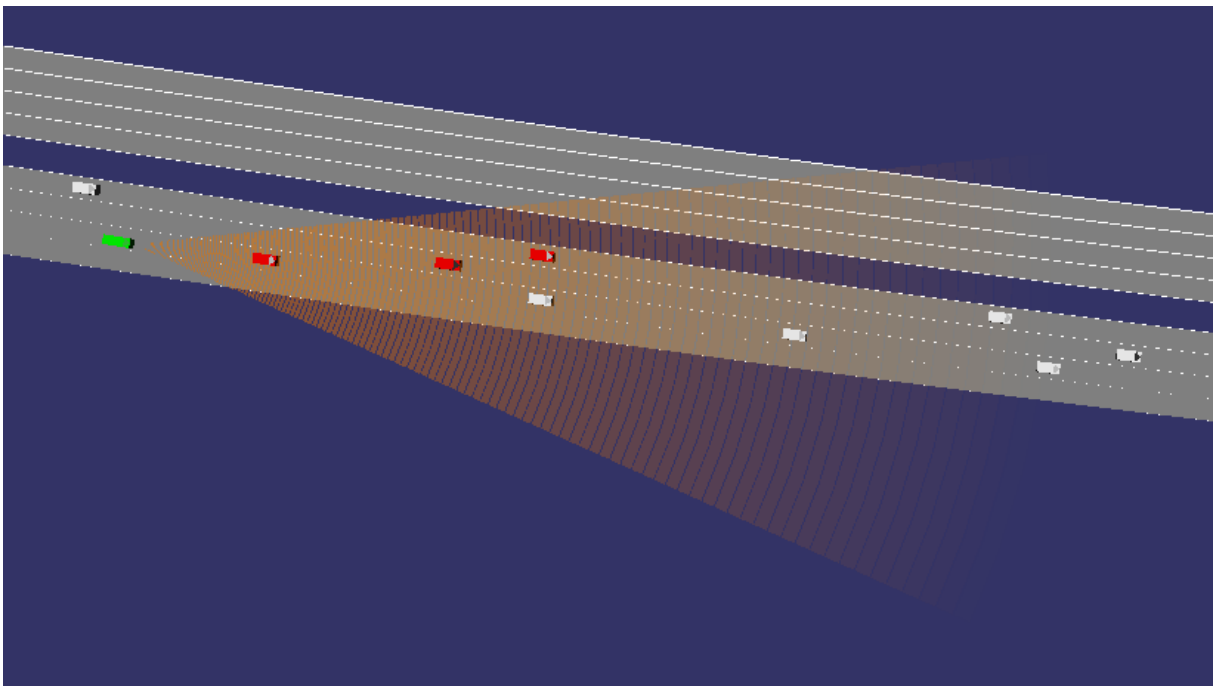
Radarsimulation

Einbinden der Applikation in die JDVE

Um die Applikation DominionED einzubinden müssen die Dateien DominionED_DLL.exe und VehicleModelDB_v3.xml in den bin-Ordner der JDVE-Simulation kopiert werden. Soll diese Applikation dann automatisch mitgestartet werden ist noch die Zeile
start ./DominionED_DLL.exe
in die TestScenarioExpressway.bat einzupflegen.
Sollten alle Applikationen auf einem Rechner laufen würde ich empfehlen vielleicht die Zeile die den bisherigen Viewer startet wieder zu enfernen (*start ./NexGenViewer_DLL.exe*), da DominionED ebenfalls einen Viewer aus der Vogelperspektive bietet und beide Viewer eventuell den Rechner überlastet.

Spezifikation des Radars

Der Radarsensor ist mit einer Reichweite von 150m und einem Öffnungswinkel von 30° eingestellt. Verdeckte Objekte werden nicht erkannt. Erkannte Objekte werden im zugehörigen Viewer rot markiert, sonstige Objekte sind weiß (ist manchmal hilfreich zur Kontrolle). Das Ego-Fahrzeug ist grün dargestellt.



Datenpaket

Die Structure mit dern Datenbeschreibung findet sich in der formalen Beschreibung des Datenkerns (der .xml-Datei) unter dem Namen:

```
<Structure name="DetectedVehicles">
```

Um auf diese Daten in einer eigenen Applikation zugreifen zu können muss folgender Input in der Beschreibung zu der Eurer Applikation hinzugefügt werden:

```
<Input>Environment@SensorData.DetectedVehicles</Input>
```

Es werden maximal 50 Fahrzeuge erkannt.
Ein Zugriff auf die Daten ist innerhalb einer Applikation z.B. folgendermaßen möglich:

```
_output->Environment.SensorData.DetectedVehicles[i].LaneID
_output->Environment.SensorData.DetectedVehicles[i].PositionUTM[0]
```

Das Array der detektierten Fahrzeuge wir von 0 an gefüllt. Nicht erkannte oder nicht vorhandene Fahrzeuge erhalten die CarTrafficID = -1. (z.B. liefert die Abfrage `_output->Environment.SensorData.DetectedVehicles[10].CarTrafficID` eine -1, so sind die nachfolgen Array-Werte für 11–49 auch -1 und enthalten somit keine detektierten Fahrzeuge mehr!

Ein detektiertes Fahrzeug wird mit folgenden Werten gespeichert:

Name	Beschreibung	Datentyp
Type	Beschreibt das Aussehen und die Größe des Fahrzeuges, sollte hier allerdings immer den Wert 4 beinhalten, da nur ein Fahrzeugtyp verwendet wird.	int
CarTrafficID	Einem Fahrzeug zugewiesene eindeutige ID	int
LaneID	Die eindeutige ID des Fahrspurobjektes das das Fahrzeug befährt	int
PositionUTM	Koordinatenarray (Absolutkoordinaten) PositionUTM[0] = X-Koordinate des Fahrzeugemittelpunktes PositionUTM[1] = Y-Koordinate des Fahrzeugemittelpunktes PositionUTM[2] = Z-Koordinate des Fahrzeugemittelpunktes PositionUTM[3] = Roll-Winkel (immer 0) PositionUTM[4] = Pitch-Winkel (immer 0) PositionUTM[5] = Heading-Winkel	float
Velocity	Geschwindigkeit (Absolut)	float
Length	Länge des Fahrzeuges (m)	float
Width	Breite des Fahrzeuges (m)	float