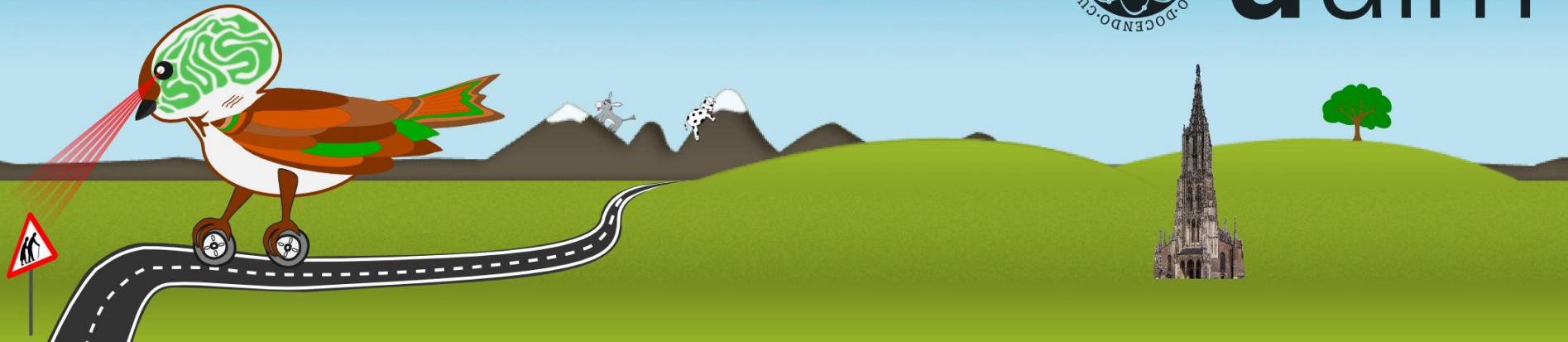


Team Spatzenhirn



universität
uulm



Carolo-Cup Fahrzeugpräsentation

Patrick Hatzelmann
Marcel Debout

Braunschweig, 5. Februar 2013

Zentrale Frage:

Kann euer Spatz
fliegen?

Gliederung

Der Spatz

Rundkurs

Hindernisfahrt

Einparken

Gliederung

Der Spatz

Rundkurs

Hindernisfahrt

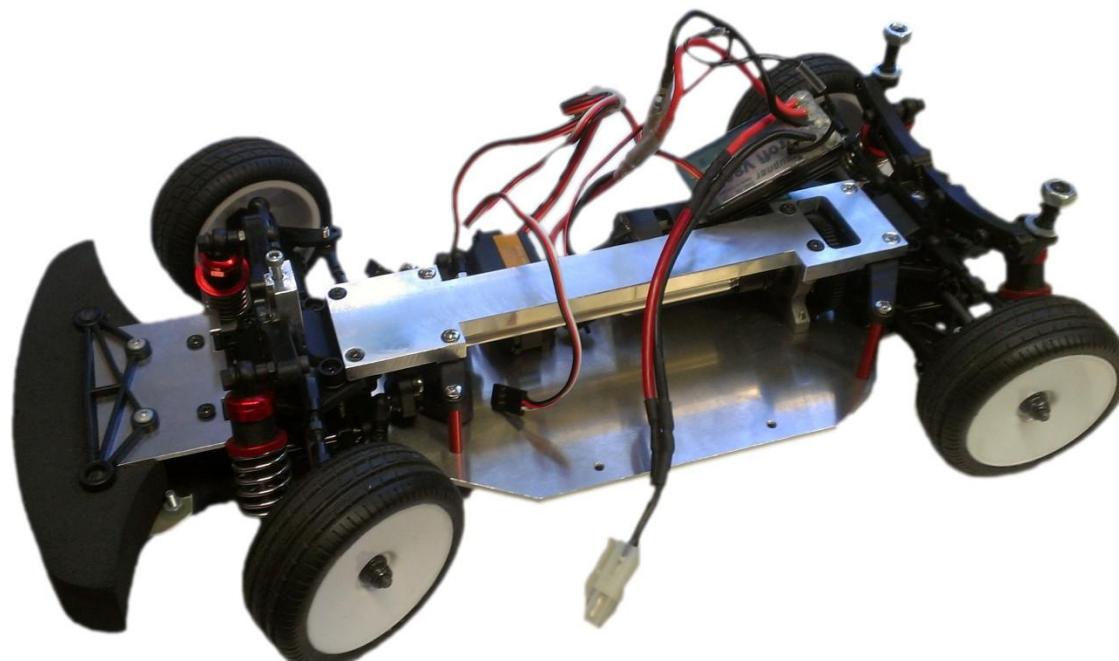
Einparken

Hardwarekonzept

Hardwarekonzept

Fahrzeugaufbau

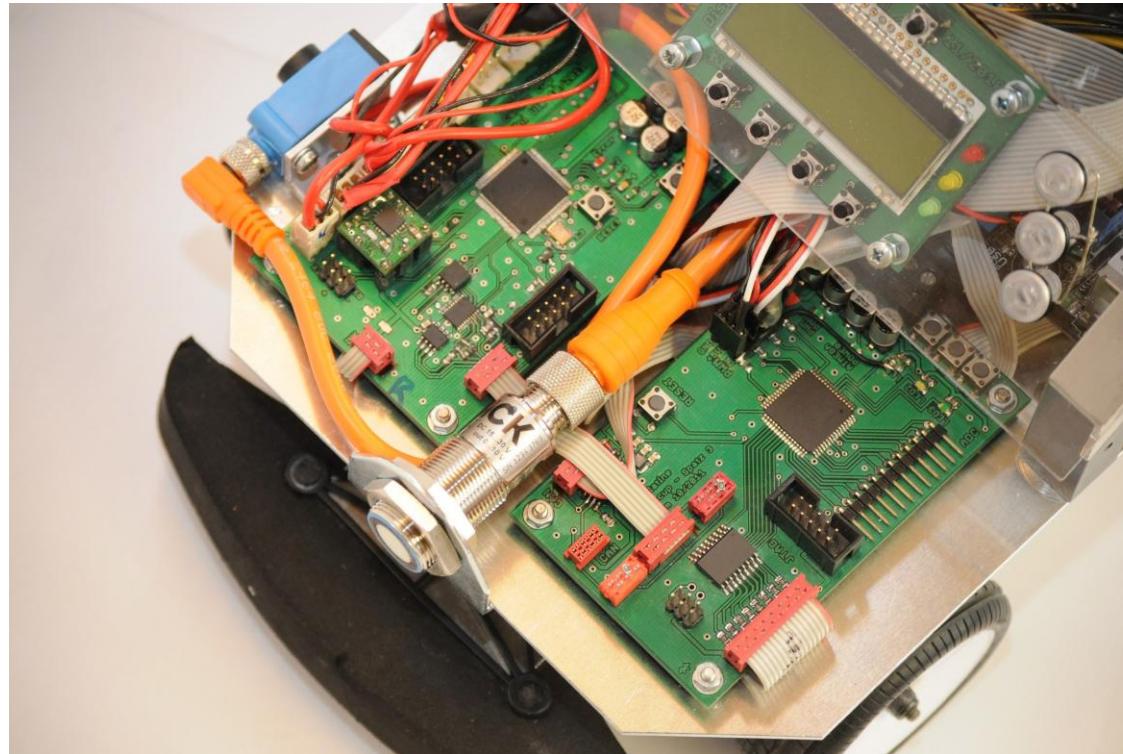
- Eigenentwickeltes Chassis aus Aluminium
- Eigenentwickelte Karosserie aus Plexiglas
- Allradgetrieben



Hardwarekonzept

Platinen

- Eigenentwickelte Aktor-, Sensor- und Energieplatine
- Kommunikation mittels CAN-Bus



Hardwarekonzept

Recheneinheit

- Mainboard: Mini-ITX
- CPU: Intel Core i7 3770T, 4 x 2.50 GHz
- GPU: nVidia Quadro 600, 96 Kerne



Sensorkonzept

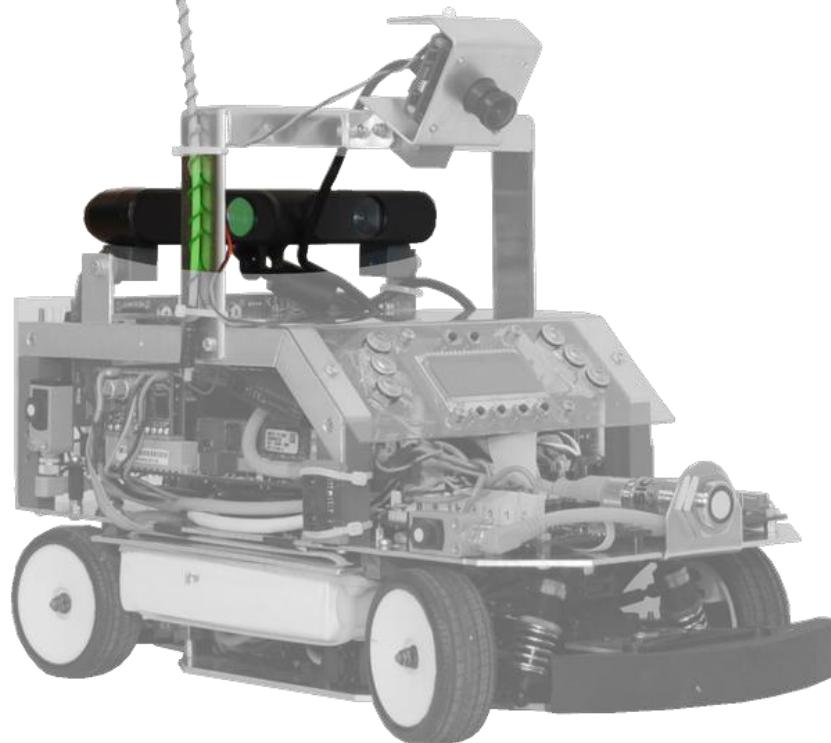


Sensorkonzept



- **VGA-Kamera:** Fahrspurdetektion, Stoppliniendetektion
- Tiefenbildkamera: Hinderniserkennung, Parklückenvermessung
- Ultraschall und Infrarot: Hinderniserkennung, Parklückenvermessung

Sensorkonzept



- VGA-Kamera: Fahrspurdetektion, Stoppliniendetektion
- **Tiefenbildkamera:** **Hinderniserkennung, Parklückenvermessung**
- Ultraschall und Infrarot: Hinderniserkennung, Parklückenvermessung

Sensorkonzept



- VGA-Kamera: Fahrspurdetektion, Stoppliniendetektion
- Tiefenbildkamera: Hinderniserkennung, Parklückenvermessung
- **Ultraschall und Infrarot: Hinderniserkennung, Parklückenvermessung**

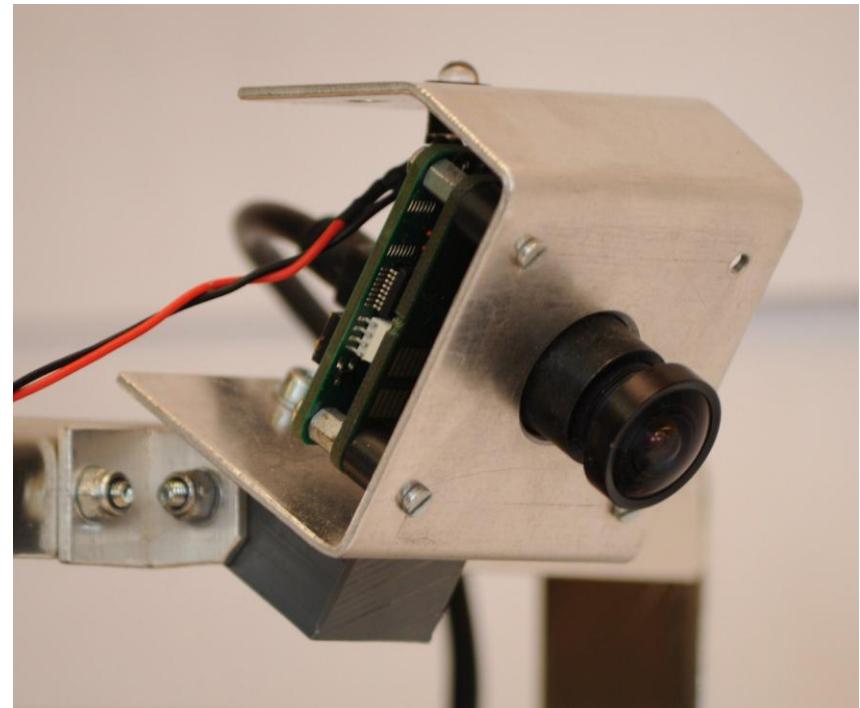
VGA-Kamera

Kamera

- Graubild-Kamera
- Auflösung: 752 x 480
- max. Framerate: 60 Hz

Linse

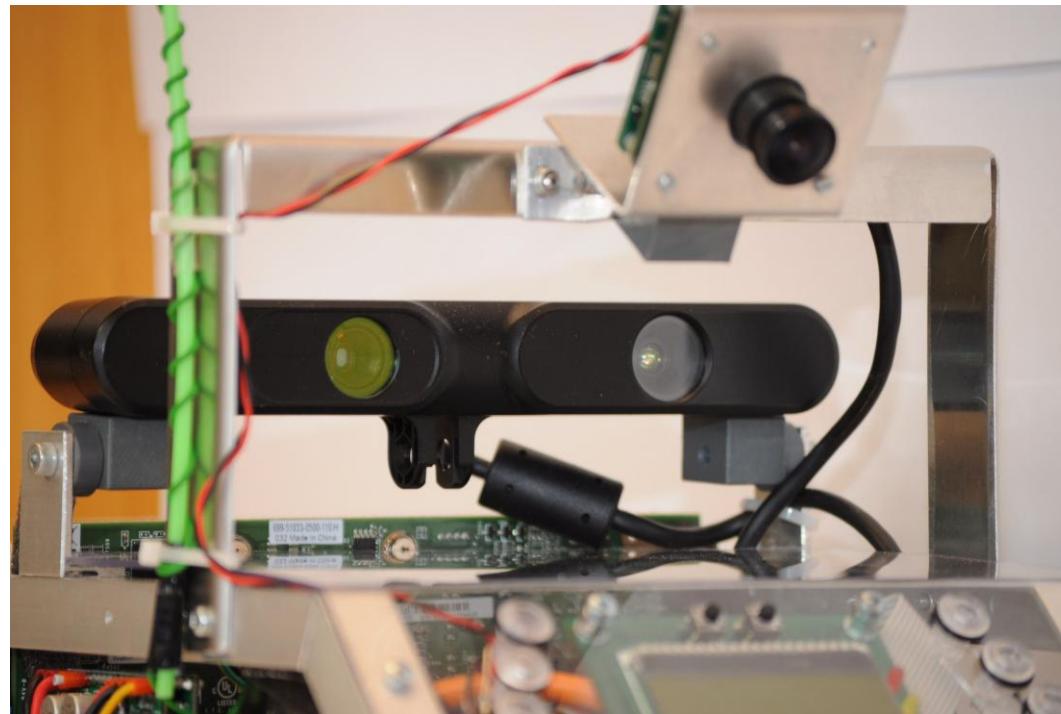
- Weitwinkel (151° Öffnungswinkel, diagonal)
- softwareseitig auf GPU entzerrt



Tiefenbildkamera

Asus Xtion Pro

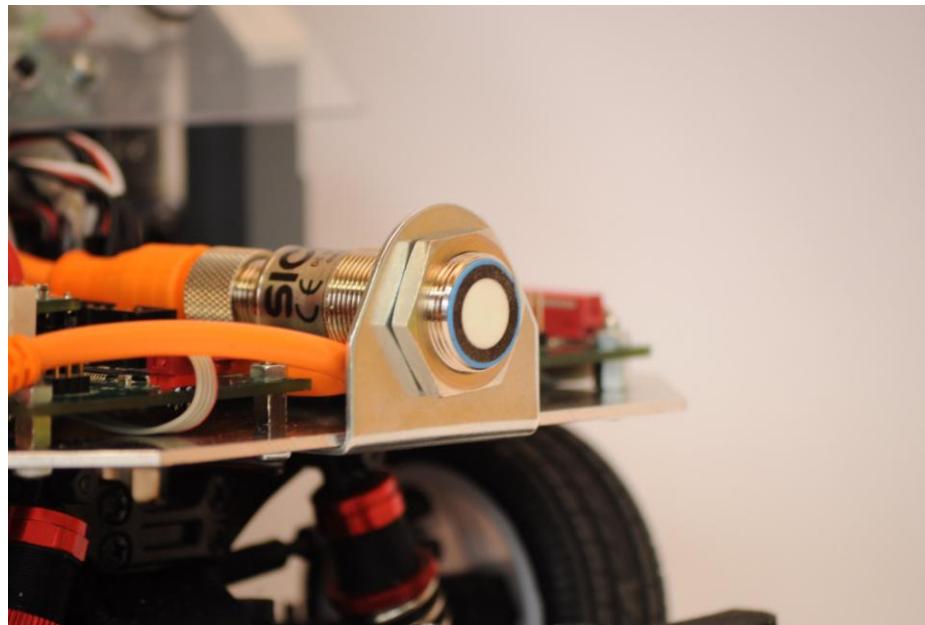
- Öffnungswinkel: 58° horizontal, 45° vertikal
- Auflösung: 320 x 240 bei max. 60 Hz
- Reichweite: 60 – 350 cm



Ultraschall- und Infrarotsensoren

Ultraschall (Front)

- Analoger Sensor
- Reichweite: 12 – 130 cm
- Framerate: 25 Hz



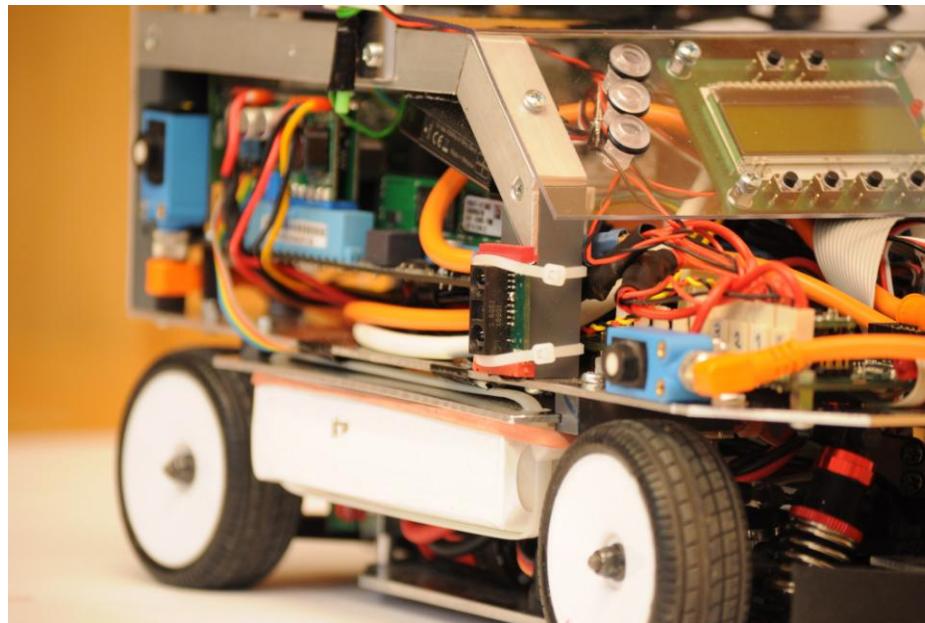
Ultraschall- und Infrarotsensoren

Ultraschall (Seite)

- Digitale Sensoren
- Reichweite: bis 25 cm
- Framerate: 33 Hz

Infrarot (Seite)

- Analoger Sensor
- Reichweite: 15 – 100 cm
- Framerate: 50 Hz



Alternative Sensorik

Laserscanner

Hohe Reichweite

Größe und Gewicht

Preis

Geringe Auflösung

Radar

Messung der Geschwindigkeit

Größe und Gewicht

Preis

Papphindernisse schlecht detektierbar

Softwarekonzept

Betriebssystem

- Ubuntu 12.04
- 32 Bit

Softwarekonzept

Betriebssystem

- Ubuntu 12.04
- 32 Bit

Framework: ADTF

- „**A**utomotive **D**ata and **T**ime-Triggered **F**ramework“
- Umfangreiche Entwicklungs- und Debug-Möglichkeiten
- Entwicklung der Algorithmen als „**T**oolboxen“
 - Hohe Wiederverwertbarkeit des Codes
 - Unabhängiges Arbeiten der einzelnen Teams
- Schnittstellen: ADTF-Filter
 - Einfache Anbindung der Toolboxen
 - Einfache Anbindung der Hardware

Alternative: eigenentwickeltes Framework

Materialkosten

Fahrzeug	500 €
Motor, Servo	260 €
Fernsteuerung, Empfänger	130 €
Fahr-, Bordakkumulator	110 €
Sensorik	665 €
Kamera	470 €
Tiefenbildkamera	110 €
Infrarotsensoren	10 €
Inkrementalgeber	35 €
Ultraschallsensoren	40 €
Sonstiges	725 €
Elektronik-Bauteile	150 €
Stromversorgung	100 €
Mainboard	310 €
Speichermedien	45 €
Grafikkarte	120 €
Gesamtkosten	2690 €

Fertigungskosten*

Platinenfertigung	140 €
Energieplatine, Display	40 €
Sensorplatine	25 €
Aktorplatine	25 €
Mechanik	660 €
Verkleidung (Plexiglas)	230 €
Rahmenaufbau	200 €
Kalibrierung der Sensorik	110 €
Grundplatte	90 €
Akkuhalterung	30 €

* angenommener Stundensatz: 40 €

Energiebilanz

Verbrauch*

Mainboard + CPU + Grafikkarte (typisch)	65 W
Motor	15 W
Servolenkung	5 W
Energieplatine, Display	2,7 W
Sensorplatine	1,7 W
Aktorplatine	0,8 W
Funkempfänger	0,08 W
Gesamtverbrauch	~ 95 W

Versorgung

Motorakku (NiMH)	30 Wh
Boardakku (LiPo)	50 Wh

→ 40 Minuten Laufzeit \triangleq 4.800m Strecke

* Rundkurs, Durchschnittsgeschw.: 2 m/s

Energieeinsparung

Umgesetzt

Aktivierung der Algorithmen nur bei Bedarf (Framework)

Aktivierung der Sensoren nur bei Bedarf

Überwachung der Spannungen und Ströme → Optimierung im Entwicklungsprozess

Möglich

Nutzung von FPGA statt PC

Gewichtsreduzierung

Gliederung

Der Spatz

Rundkurs

Hindernisfahrt

Einparken

Fahrstreifendetektion - Alternative Algorithmen

- **Canny-Kantendetektion** im gesamten Bild
→ Hough-Transformation

zuverlässig

sehr rechenaufwendig

- **Ableitungs-Filter** in Teilausschnitten (Boxen) des Bildes
→ Lineare Regressionen

weniger rechenaufwendig

wenig robust

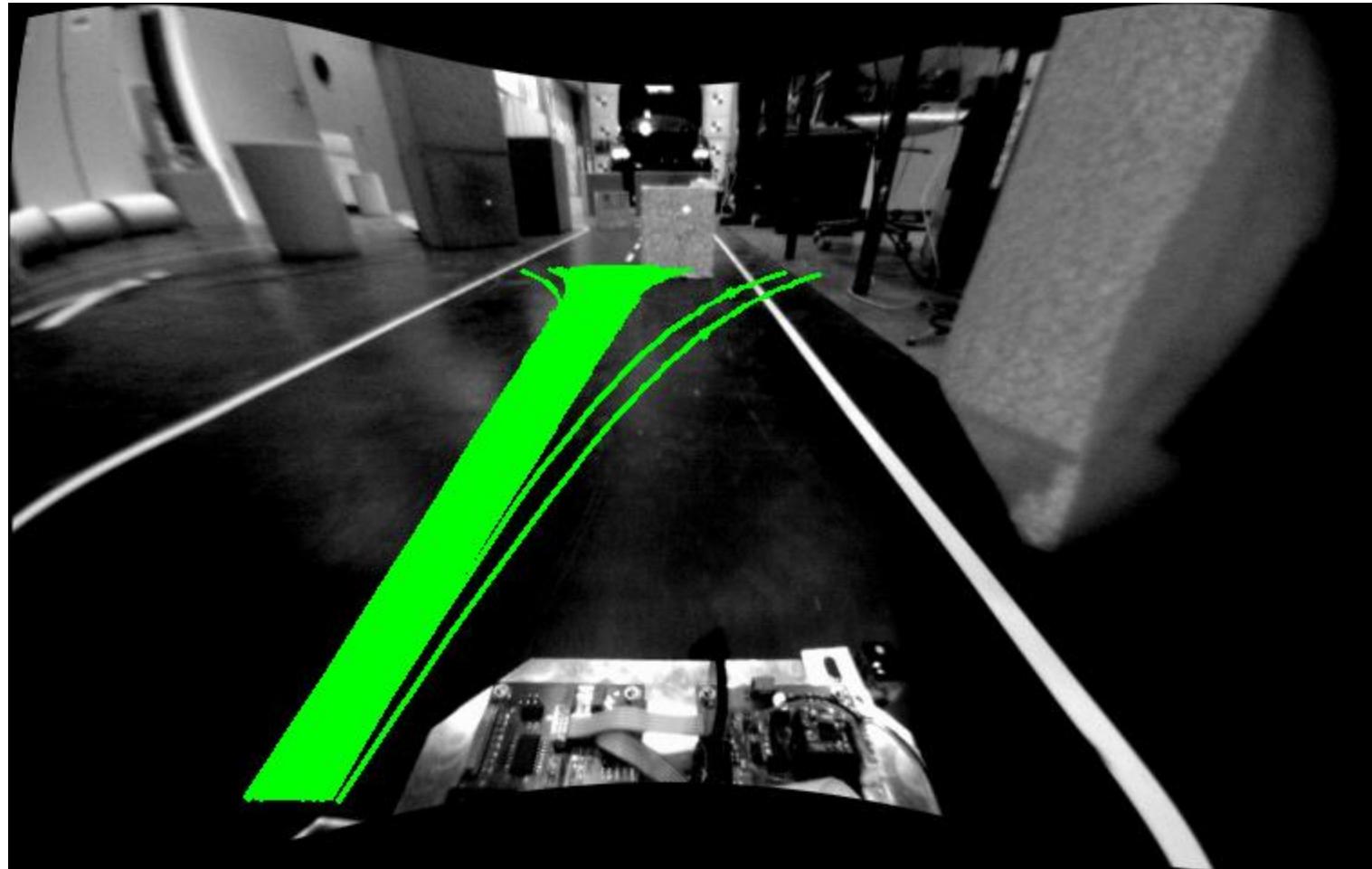
- Generelles Problem: anfällig auf Bildrauschen und Fehlstellen
- Ansatz: Fahrstreifen tracken

Fahrstreifendetektion und -tracking

→ Partikelfilter zum Tracken der Mittellinie

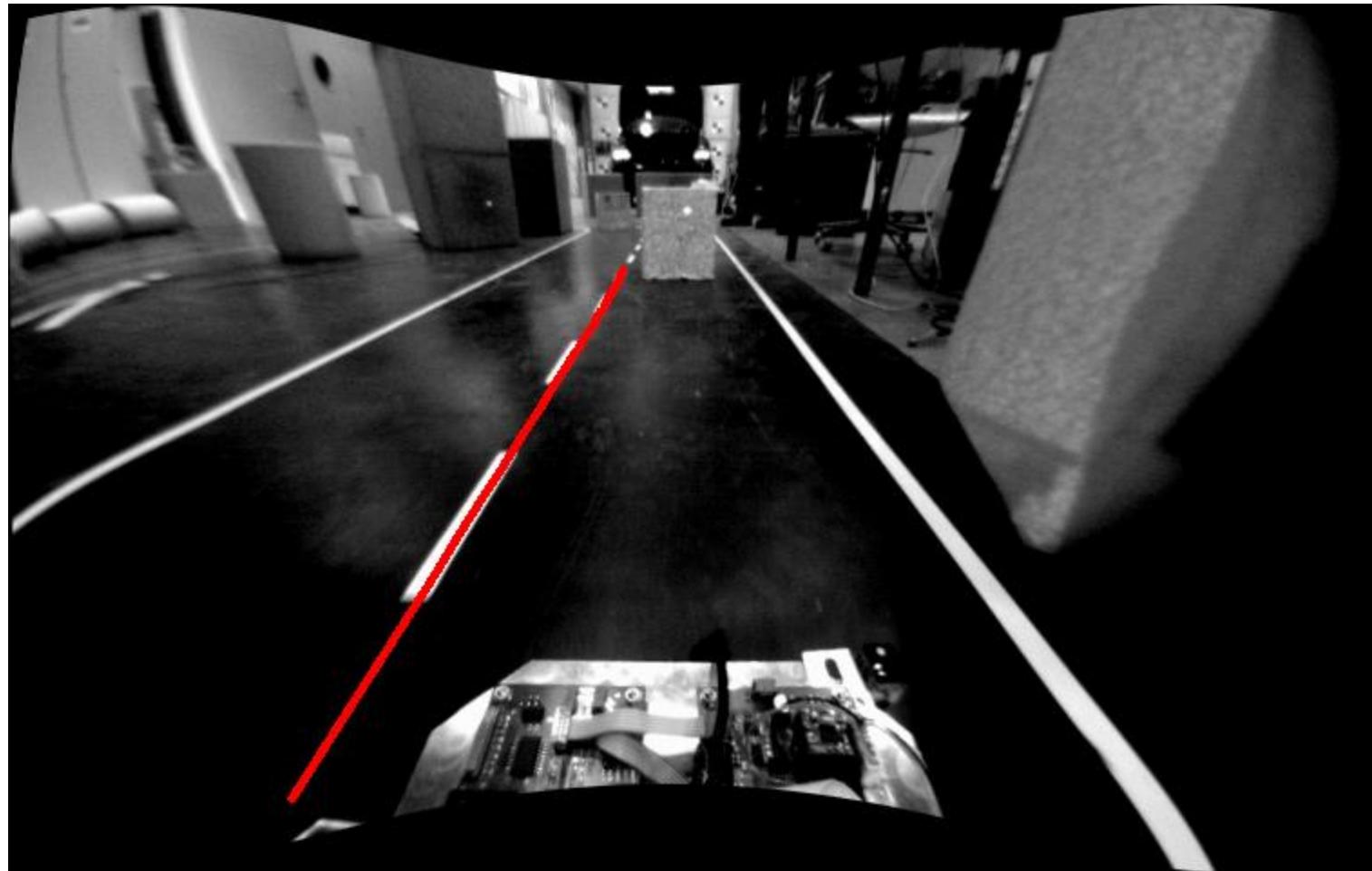
- Implementiert in CUDA auf GPU
- Berücksichtigung der Eigenbewegung zur Prädiktion
- Streuen von 500 Partikeln
- Jedes Partikel repräsentiert möglichen Zustand des Straßenverlaufs

Fahrsteifendetektion und -tracking



500 gestreute Partikel

Fahrsteifendetektion und -tracking



Höchstgewichtetes Partikel

Fahrsteifendetektion und -tracking

→ Gewichtsberechnung durch alle drei Fahrbahnmarkierungen

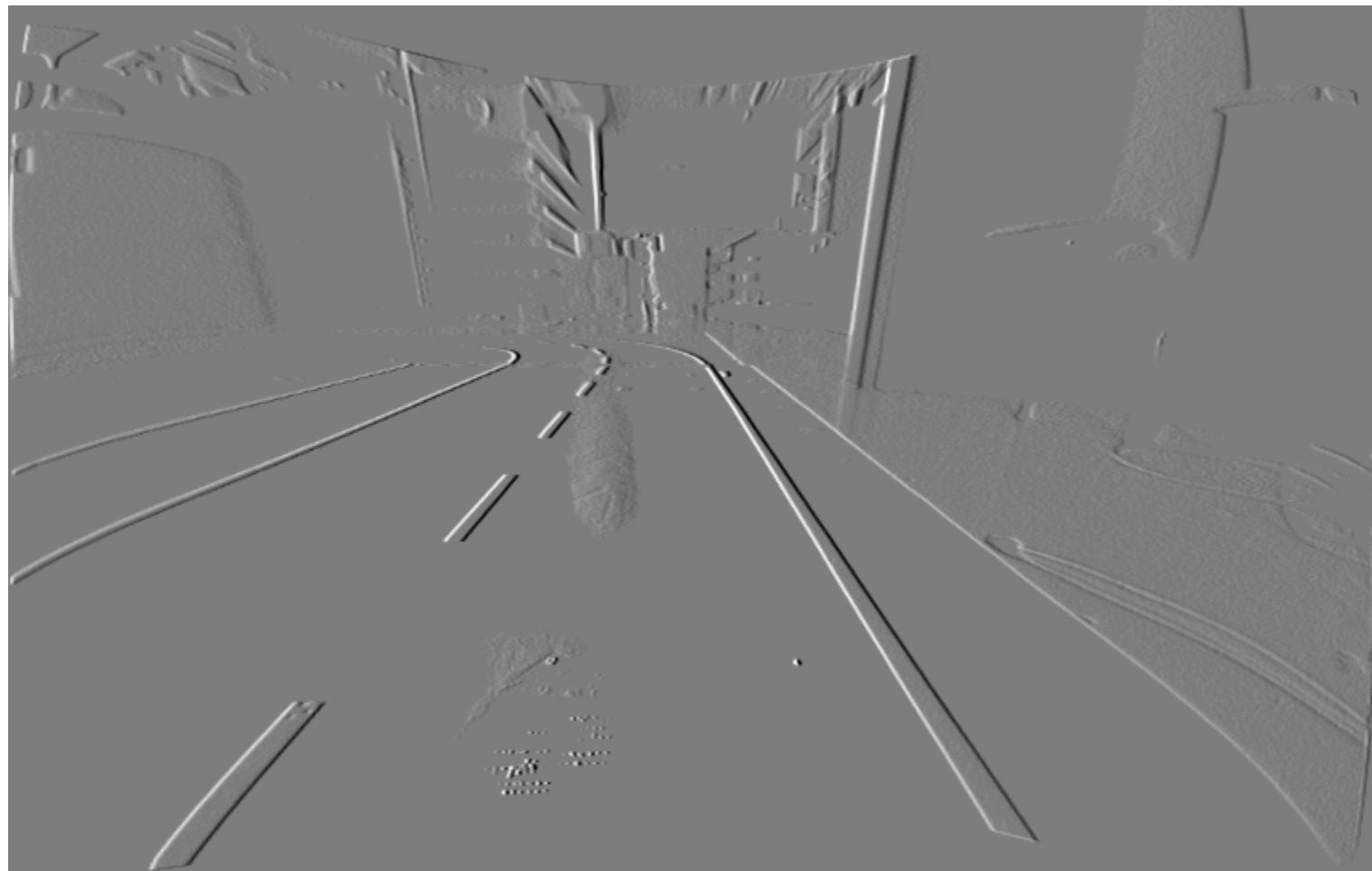
Verwendete Features

- Gradientenbild (horizontal)
- Strukturtensor



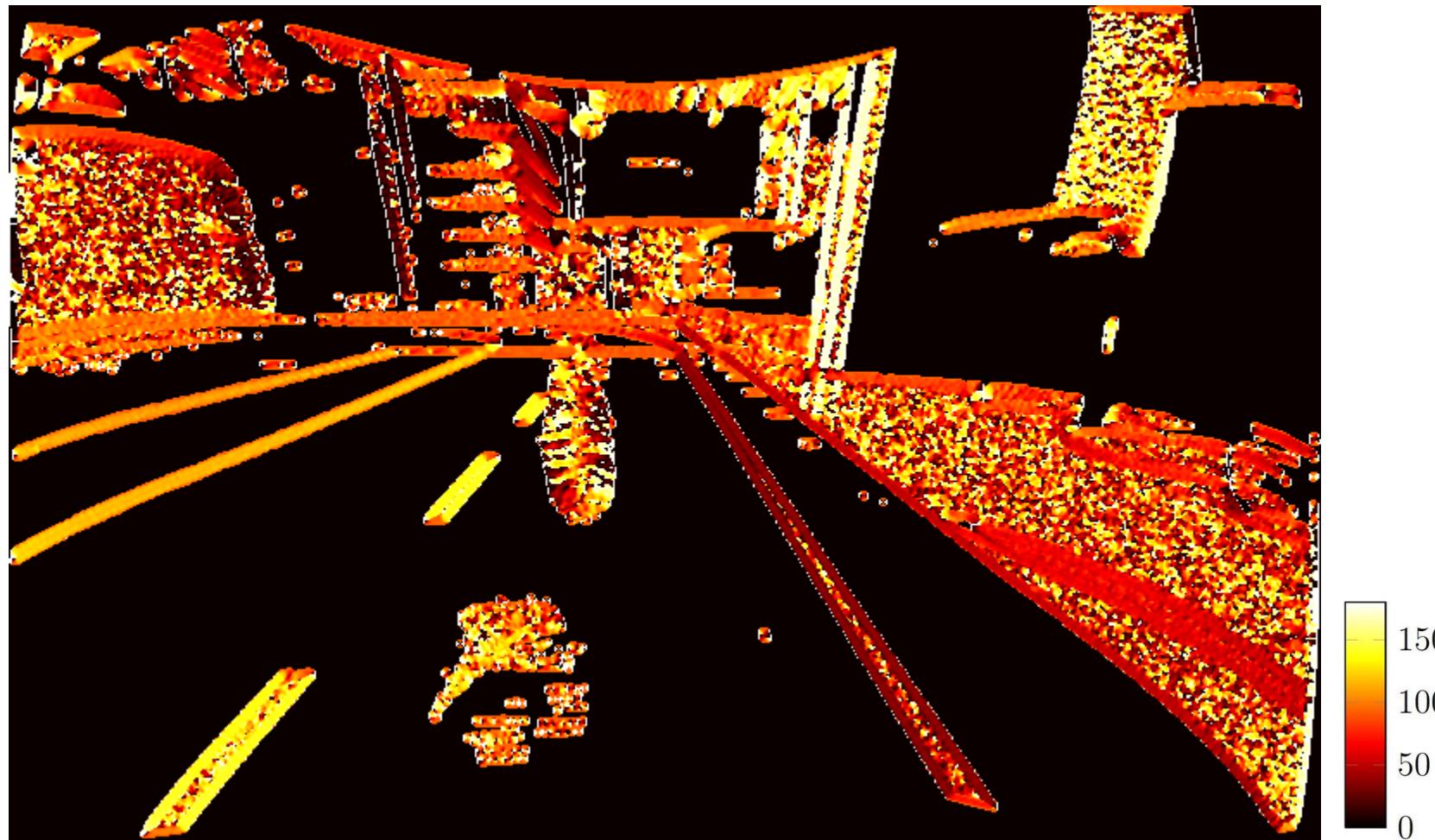
Entzerrtes Originalbild

Gradientenfeature



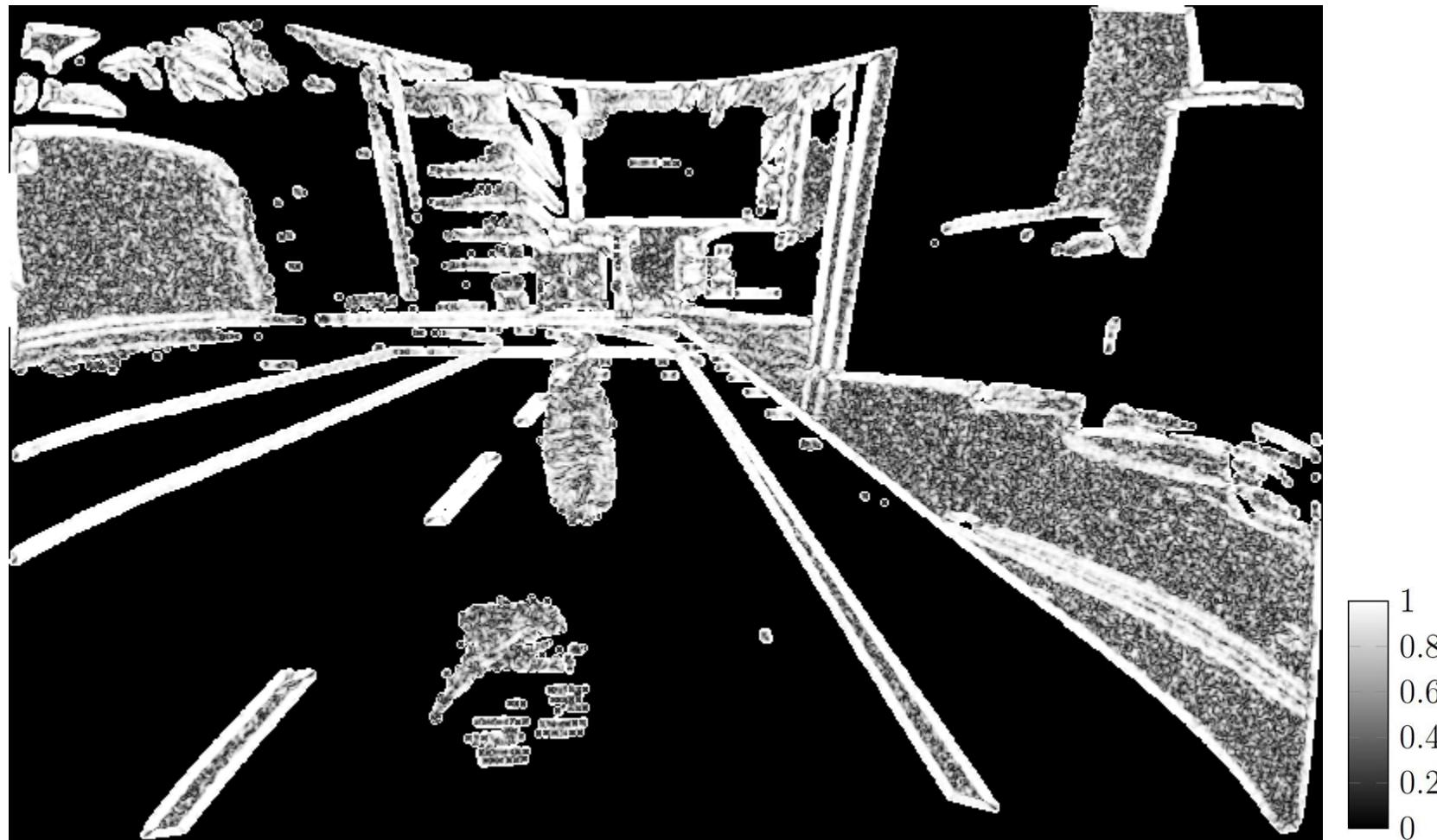
Gradientenbild (horizontal)

Tensorfeature



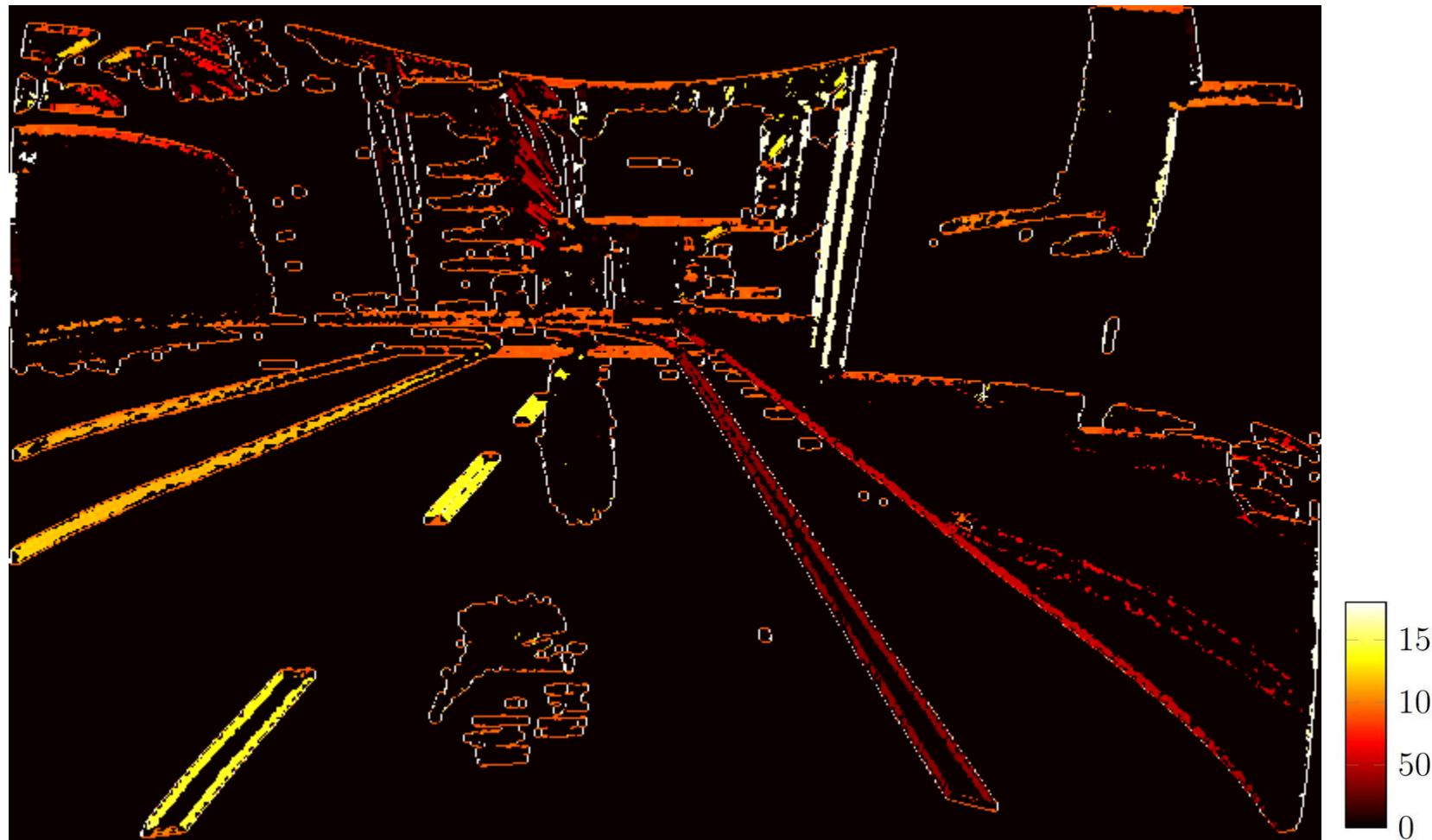
Winkelbild (in Grad) des Strukturtensors

Tensorfeature



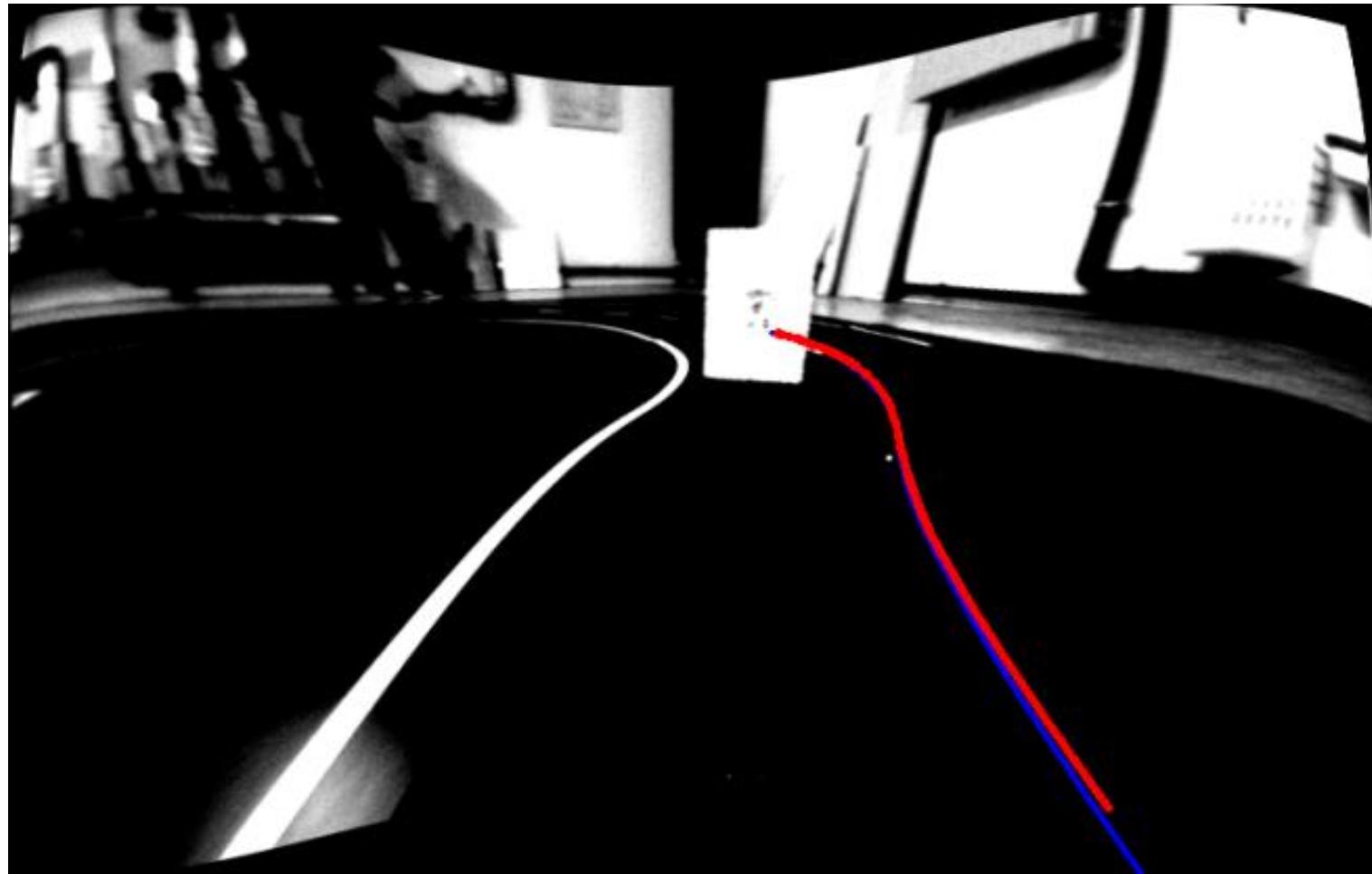
Kohärenzwerte

Tensorfeature



Winkelwerte (in Grad) des Strukturtensors

Performance



→ Detektion arbeitet sehr robust mit bis zu 35 FPS

Regelung

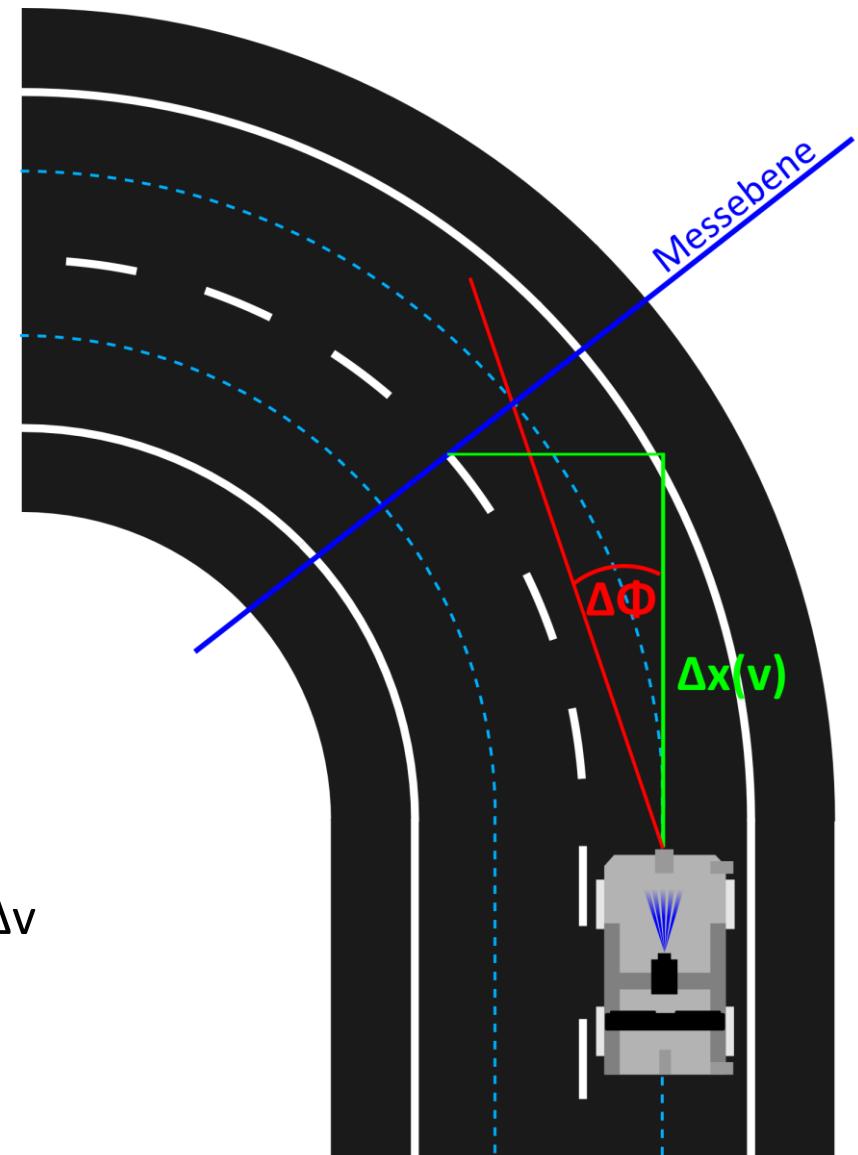
Berechnung der Ideallinien auf Basis von getrackter Mittellinie

Querregler:

- Einhalten der Ideallinie
- Regelgröße: Gierwinkeldifferenz $\Delta\Phi$
- Lösung: PID-Regler

Längsregler:

- Einhalten der Soll-Geschwindigkeit
- Regelgröße: Geschwindigkeitsdifferenz Δv
- Lösung: PI-Regler mit Vorsteuerung



Fahrdynamische Aspekte

	Bereits implementiert	Möglich	
„ASR“	Begrenzung der max. Beschleunigung abh. von Kurvenradius & Geschw.	Messung der Geschwindigkeit an jedem Rad einzeln	} + Ansteuerung einzelner Räder: ESP
„ABS“	Begrenzung der max. Bremskraft abh. von Kurvenradius & Geschw.	Messung der Geschwindigkeit an jedem Rad einzeln	

Gliederung

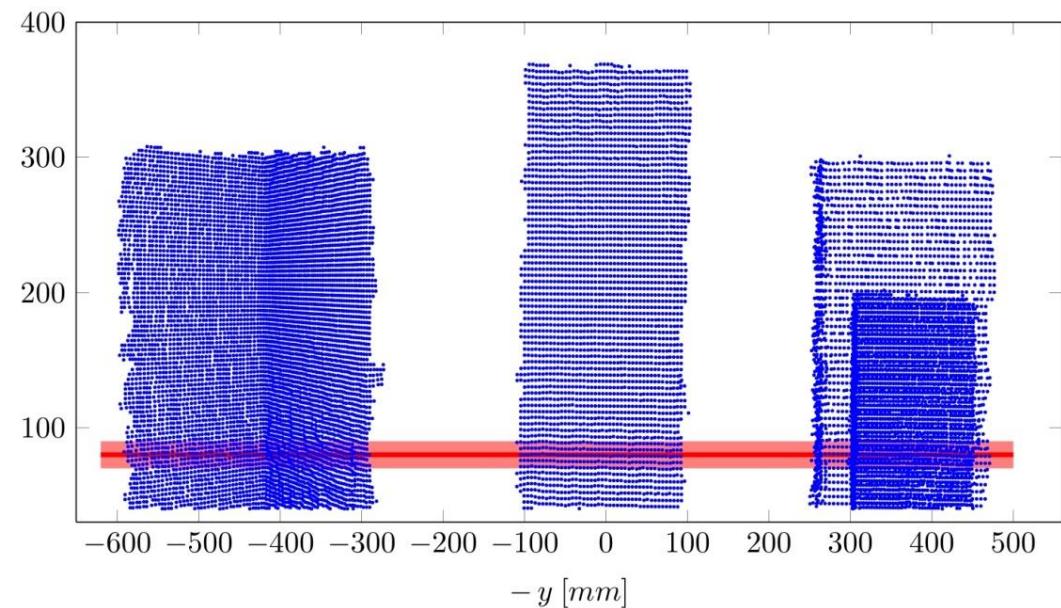
Der Spatz

Rundkurs

Hindernisfahrt

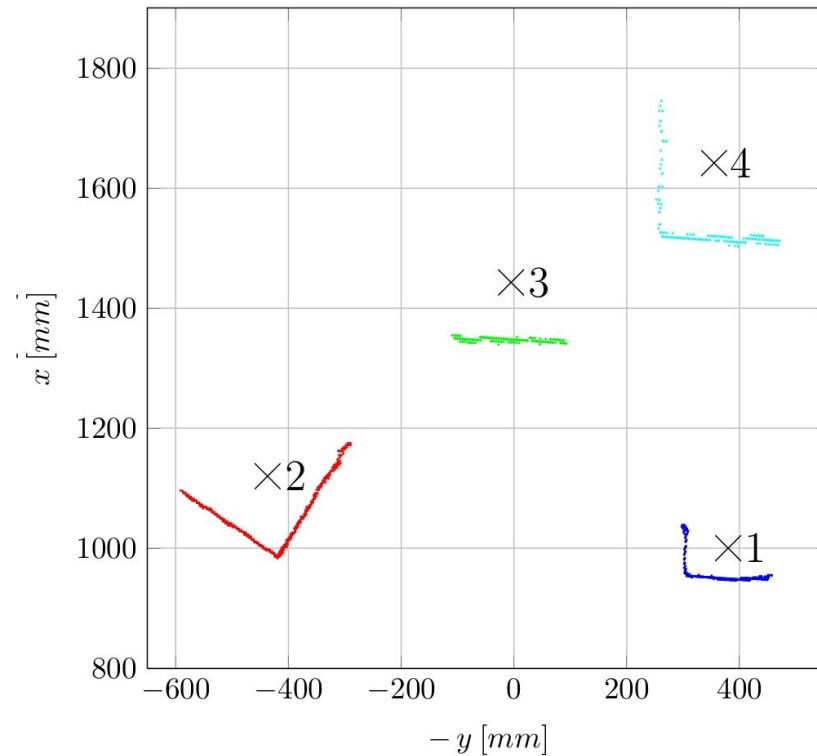
Einparken

Detektion – Algorithmus



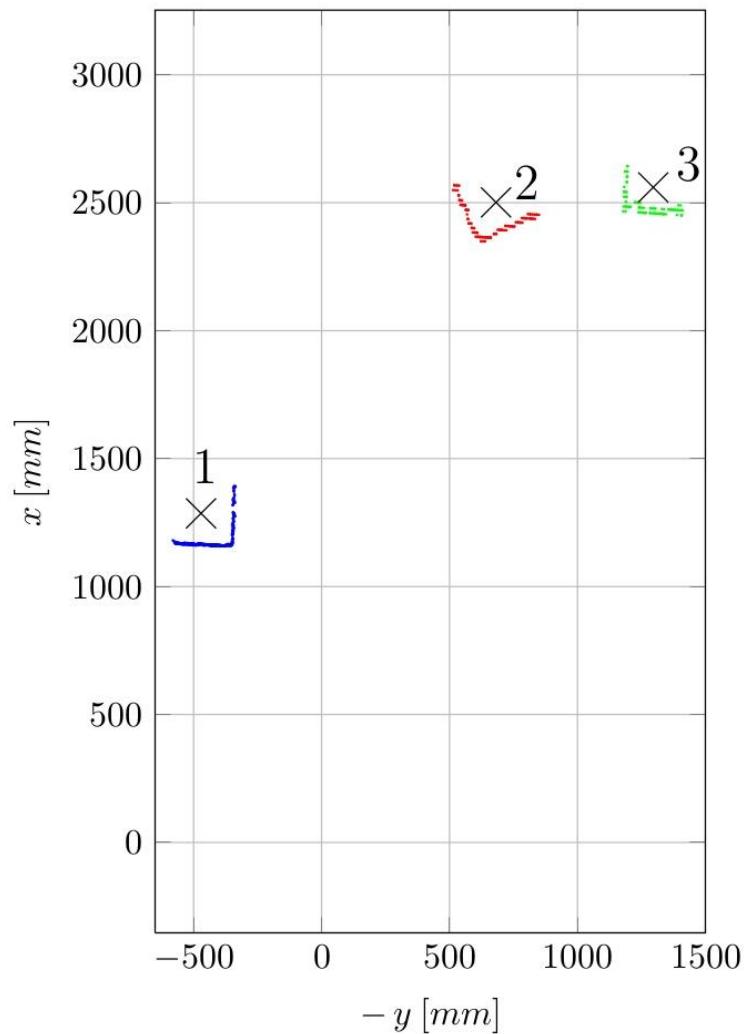
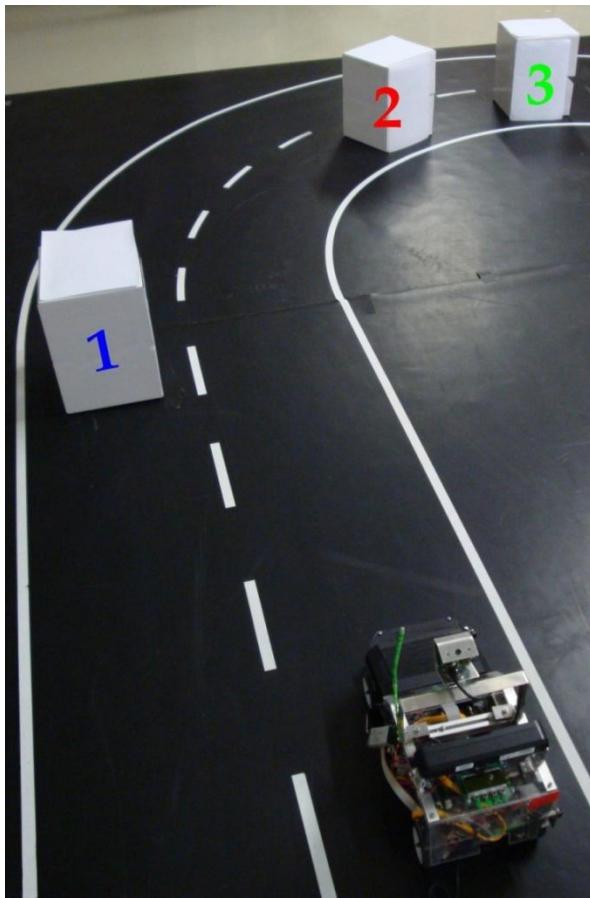
- Xtion liefert Tiefenbild
- Trafo in Fahrzeugkoordinaten (exakte Kalibrierung notwendig)
- Konstruktion einer virtuellen Ebene → parallel zum Untergrund
- Eliminierung aller Punkte mit Abstand d zur virtuellen Ebene
- Projektion auf 2D-Karte → Reduktion der Datenmenge um 95%

Detektion – Algorithmus



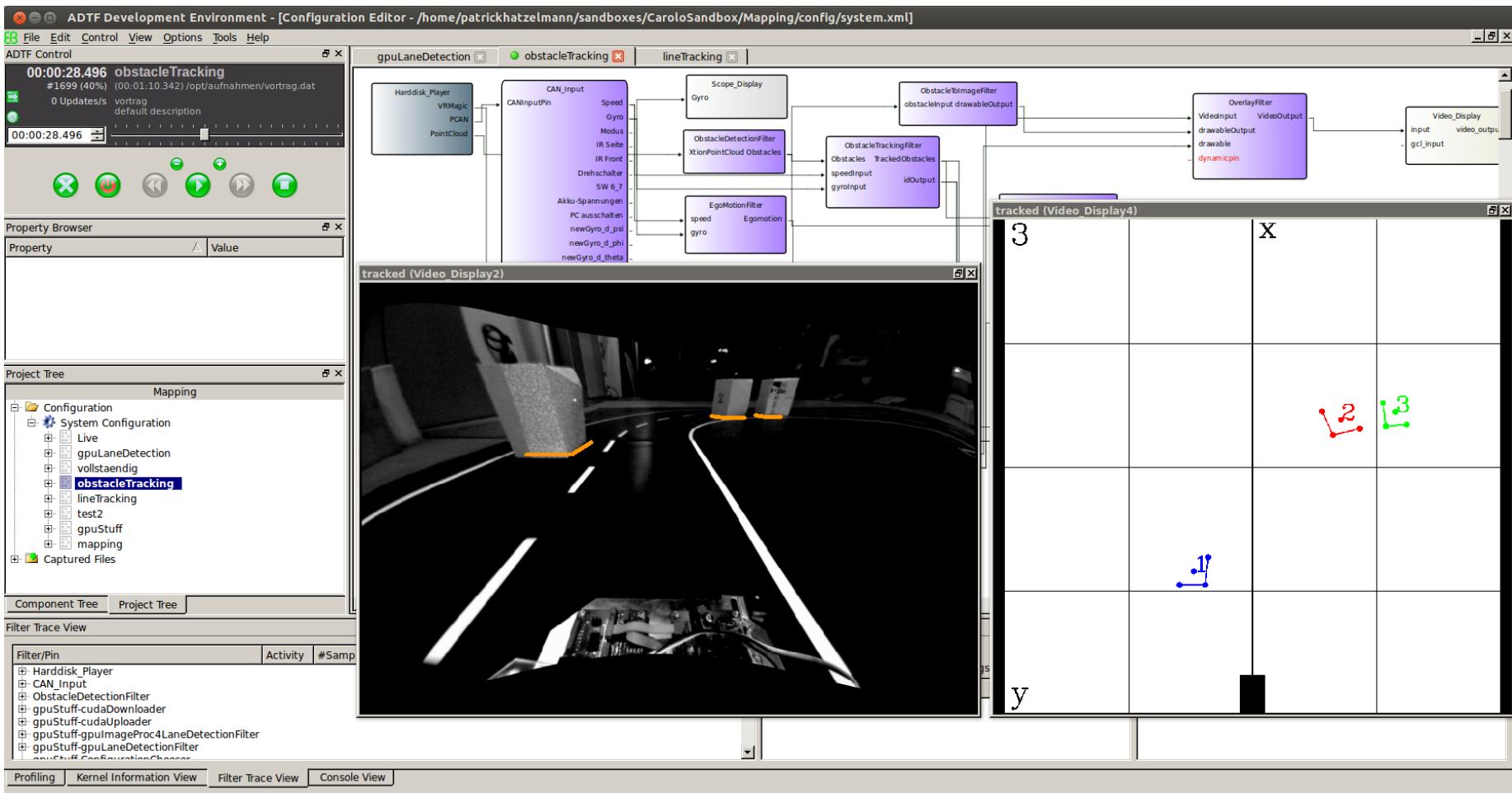
- 2D Karte
- Clustering mittels DBGridSCAN
- Bestimmung der Form (L- oder I-Shape)
- Ermittlung der Mittelpunkte

Detektion - Ergebnisse



- Performance: 45 FPS
- Lokalisierungsfehler: max. 5cm bei 3m

Detektion - Tracking



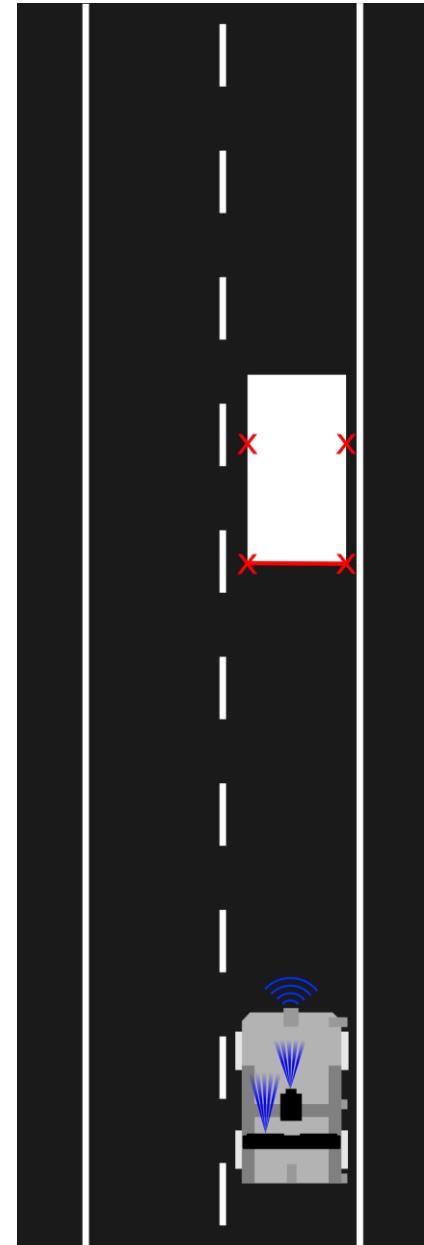
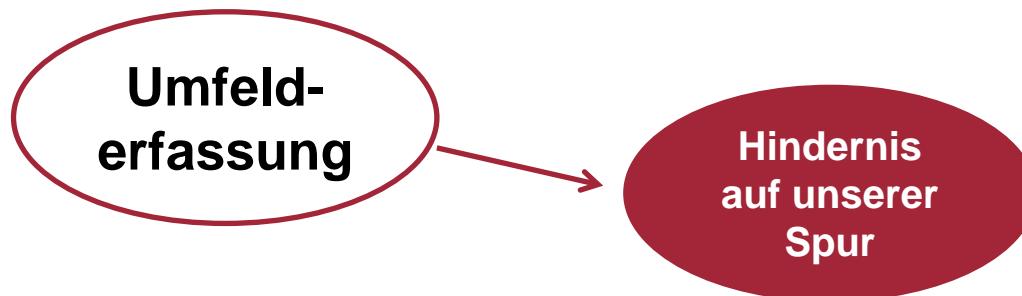
- Multi-Objekt-Tracking mit Kalman Filter
- Eliminierung des Rauschens der Xtion
- Ermittlung der Geschwindigkeit der Hindernisse

Ausweichmanöver

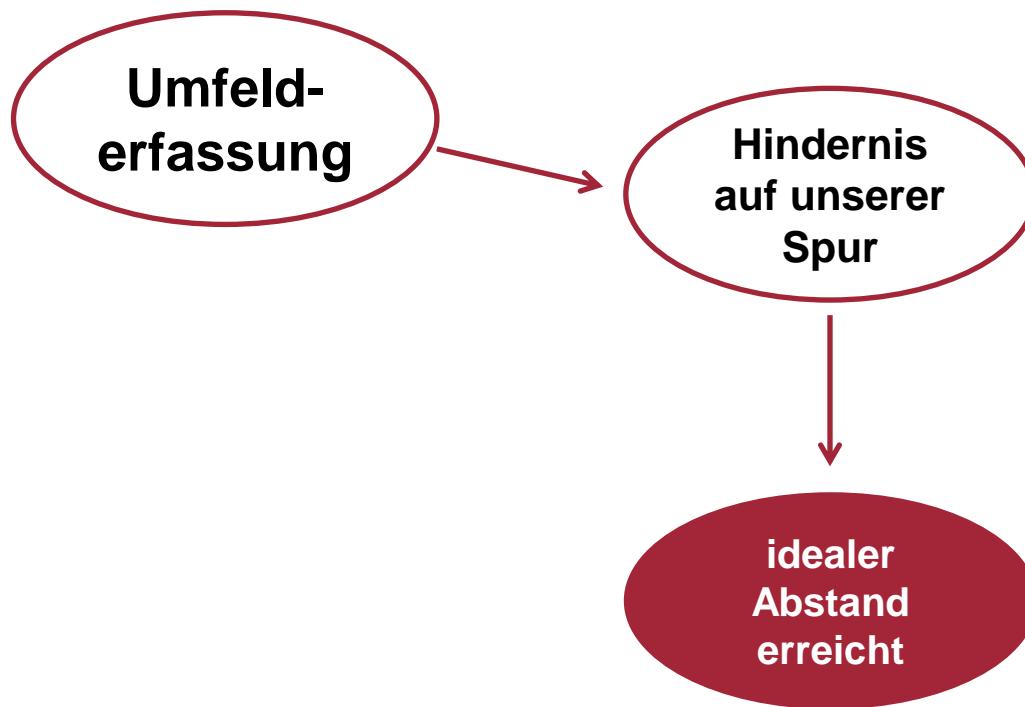
Umfeld-
erfassung



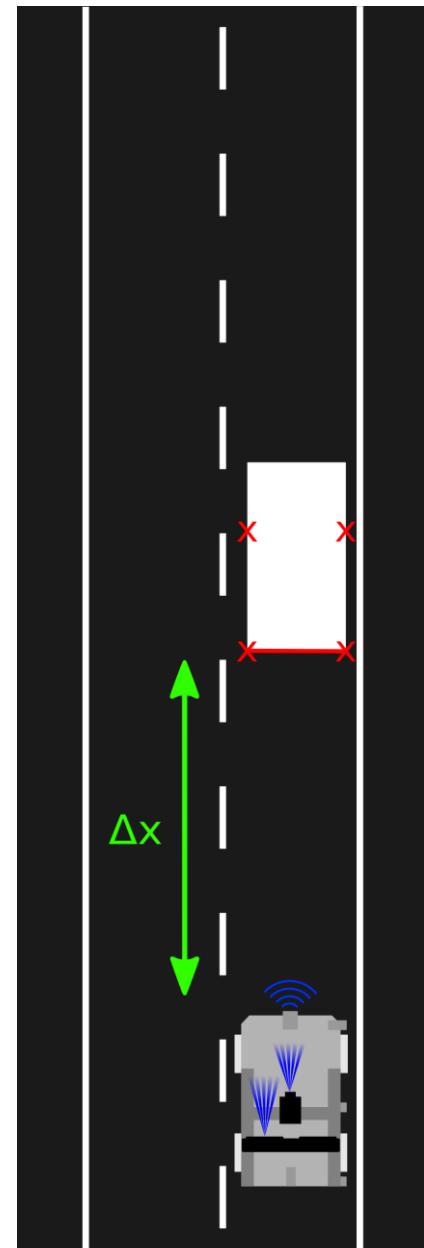
Ausweichmanöver



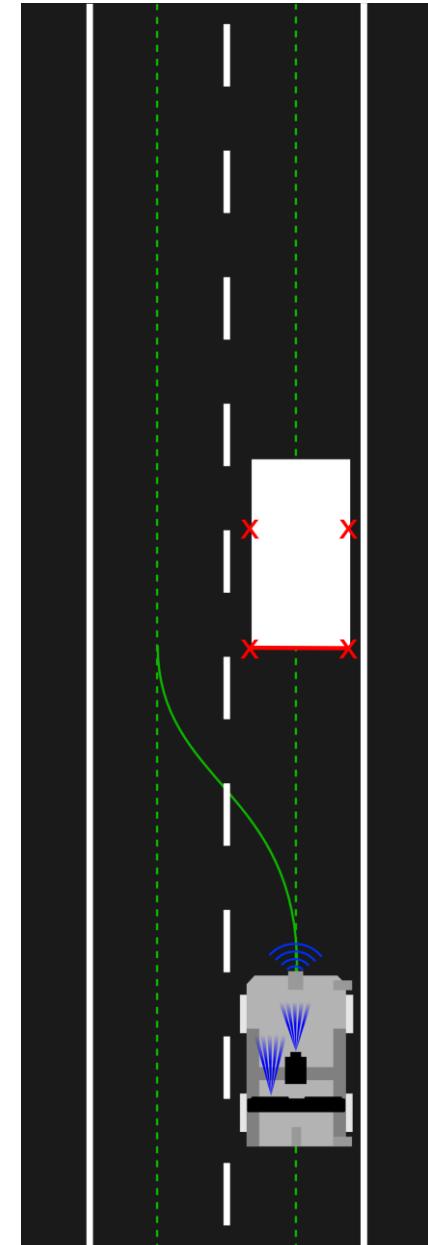
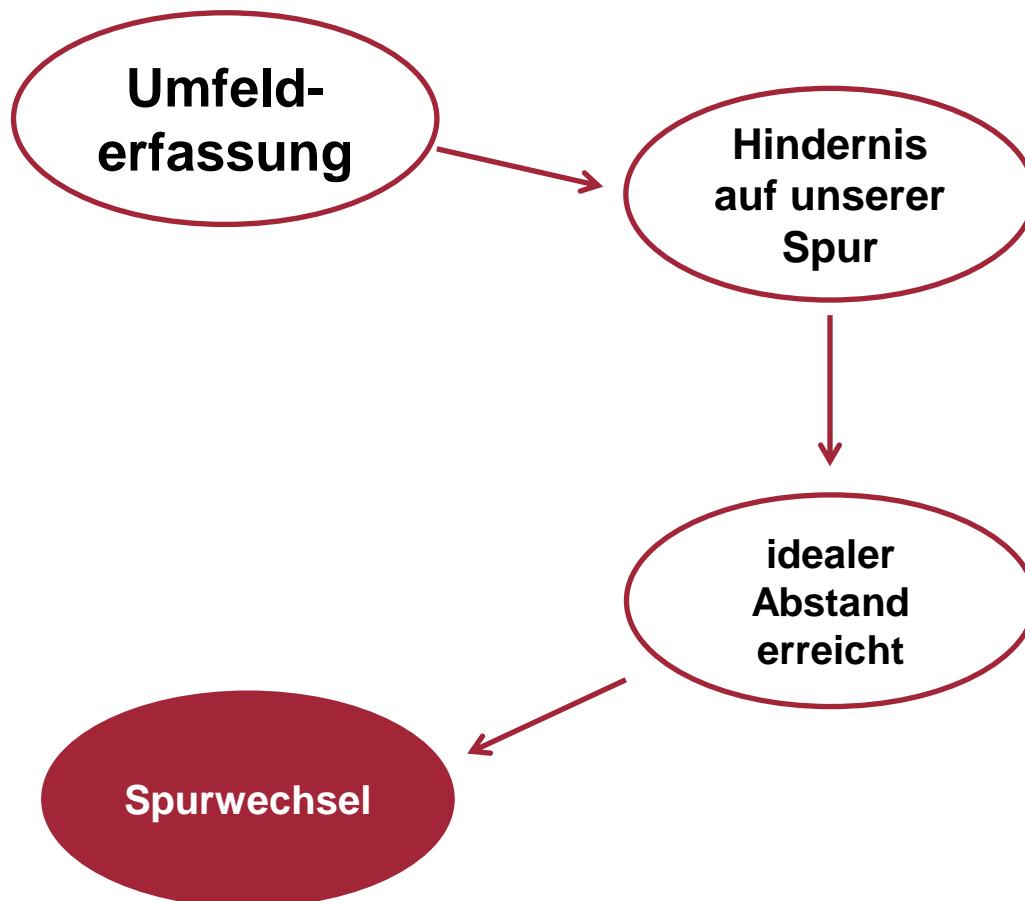
Ausweichmanöver



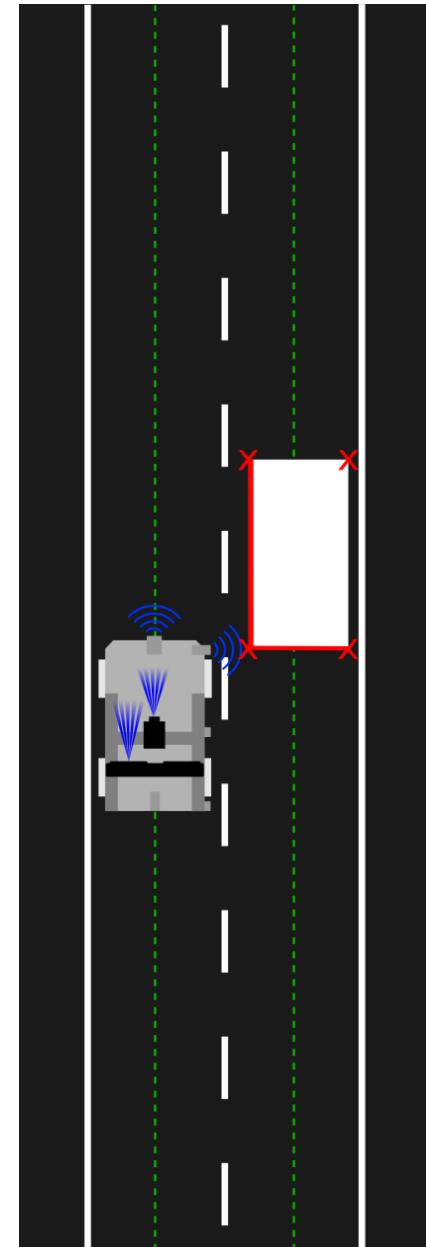
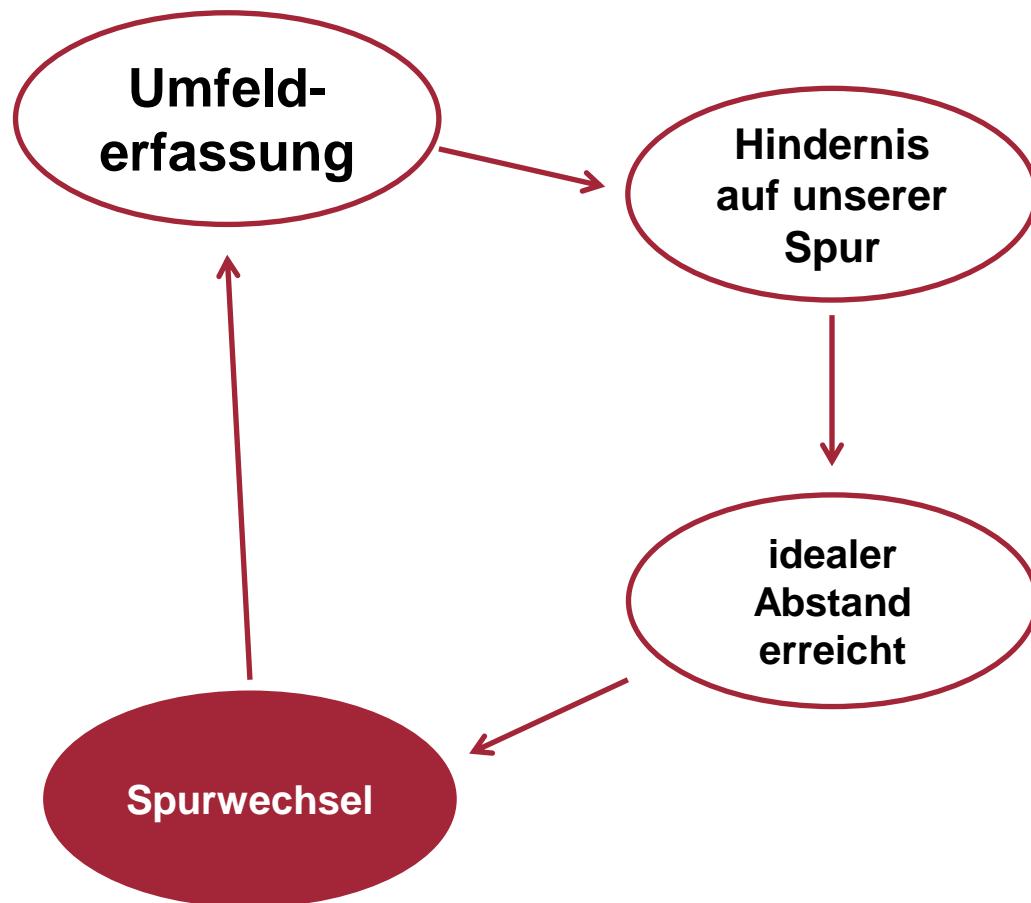
$\Delta x(v)$ = Minimaler Ausweichabstand



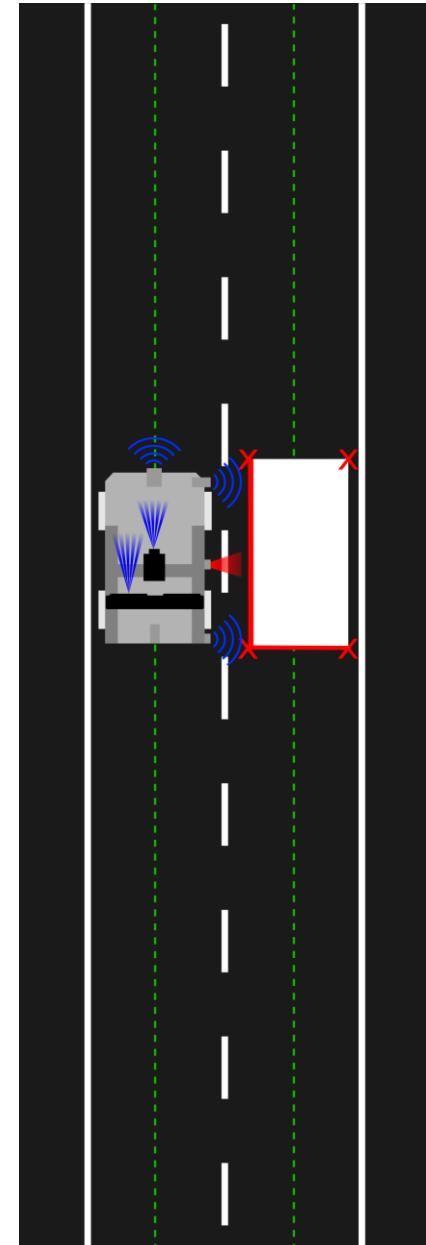
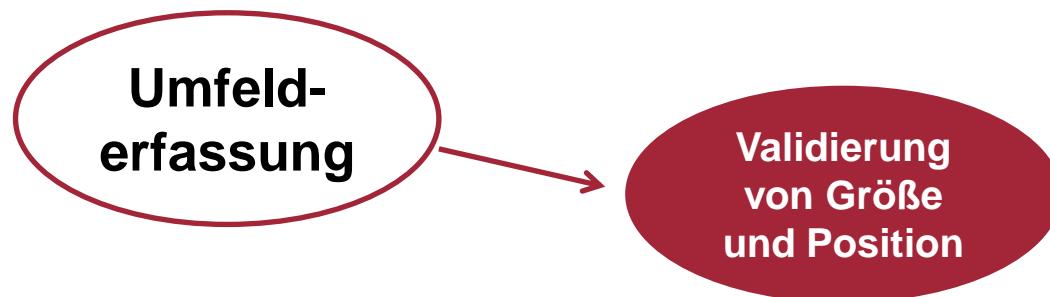
Ausweichmanöver



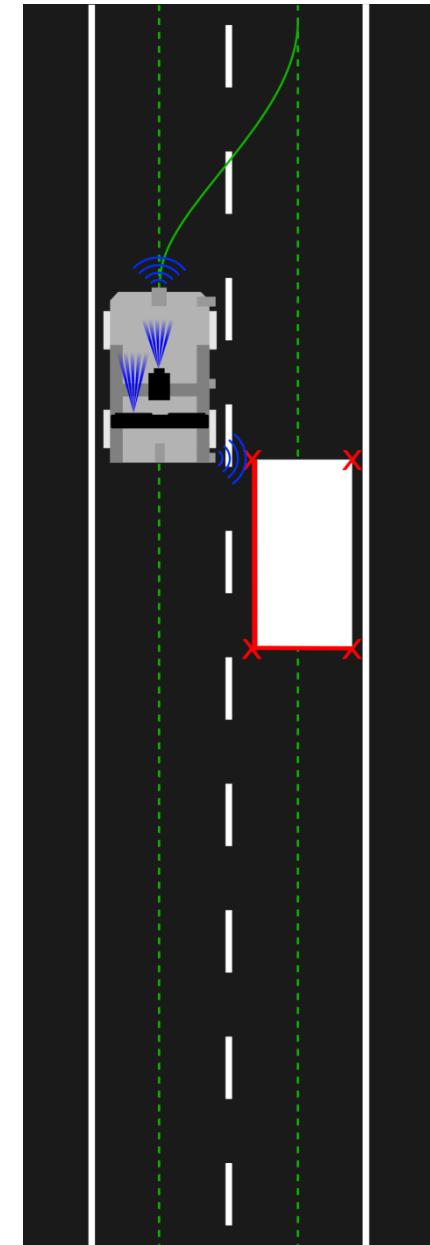
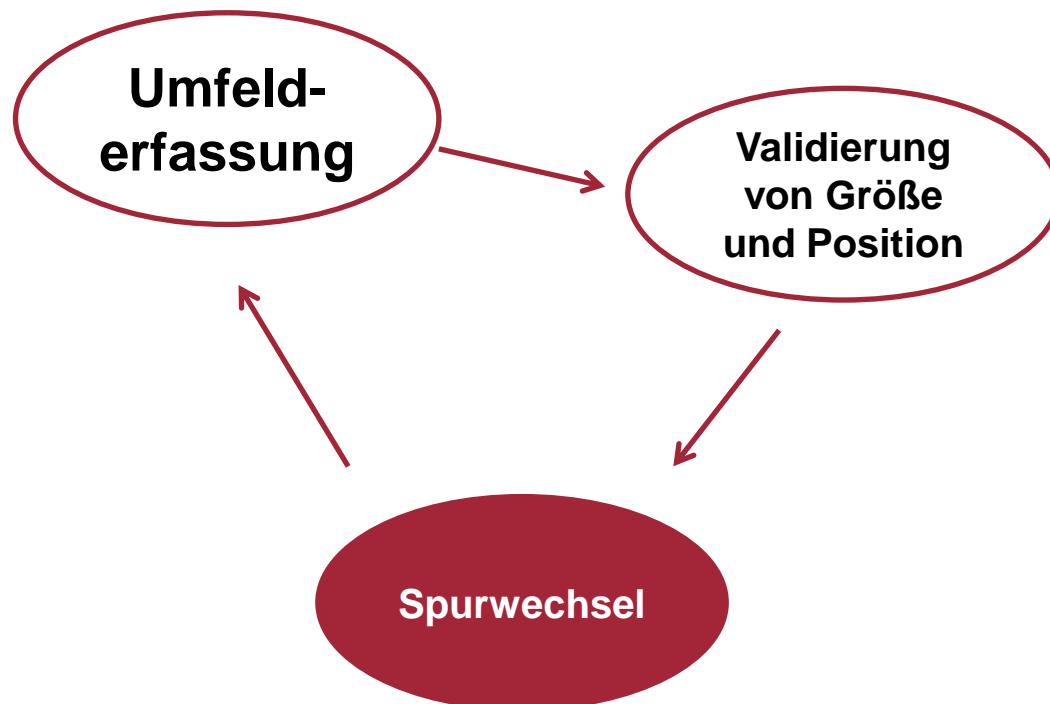
Ausweichmanöver



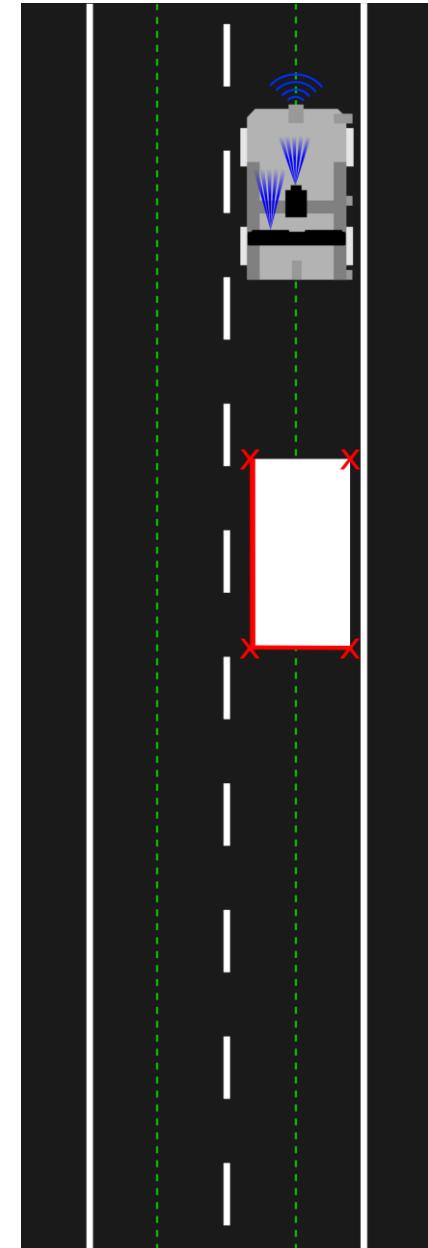
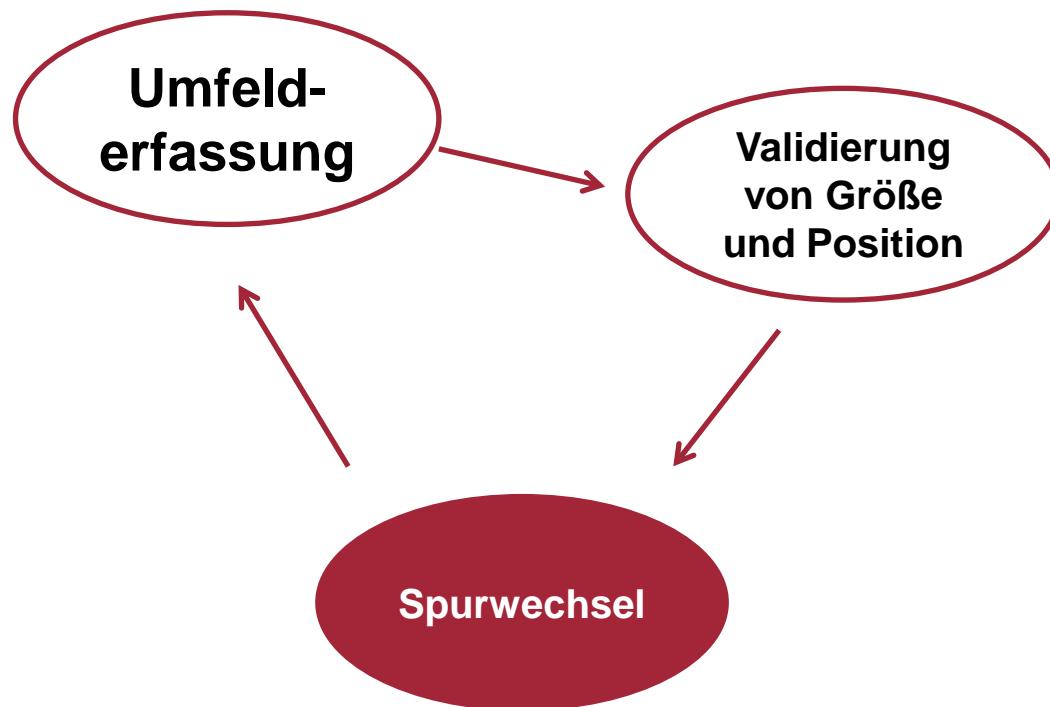
Ausweichmanöver



Ausweichmanöver



Ausweichmanöver



Besonderheiten

Hindernis auf der Gegenfahrbahn

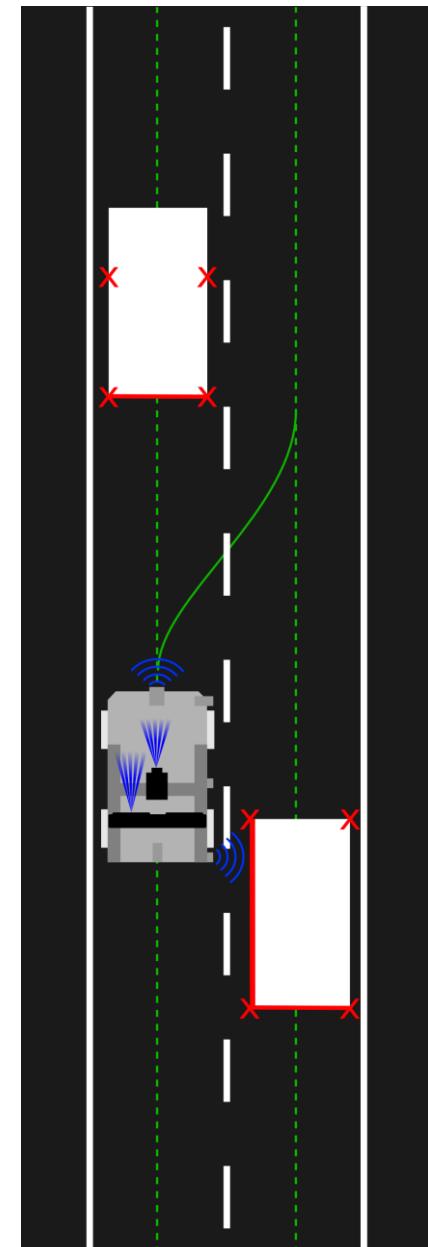
- Hindernis wird frühzeitig erkannt
- Geschwindigkeit drosseln
- Spurwechsel rechtzeitig möglich

Dynamisches Hindernis

- Ermittlung der Geschwindigkeit durch Tracking
- Beschleunigung nach Ausscheren

Notfallstrategien / Redundanz

- Ausfall des Hauptsensors → Alternativsensorik
- Ausfall aller Sensoren → Spatz hält an → Reset

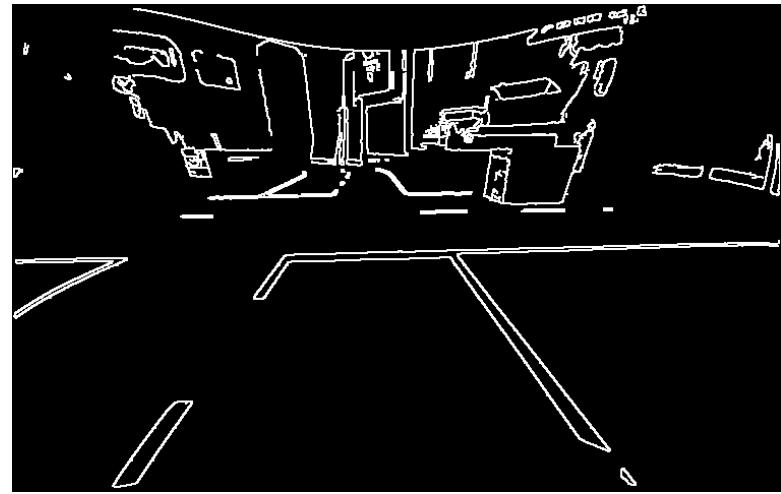


Kreuzungskonzept - Linienerkennung



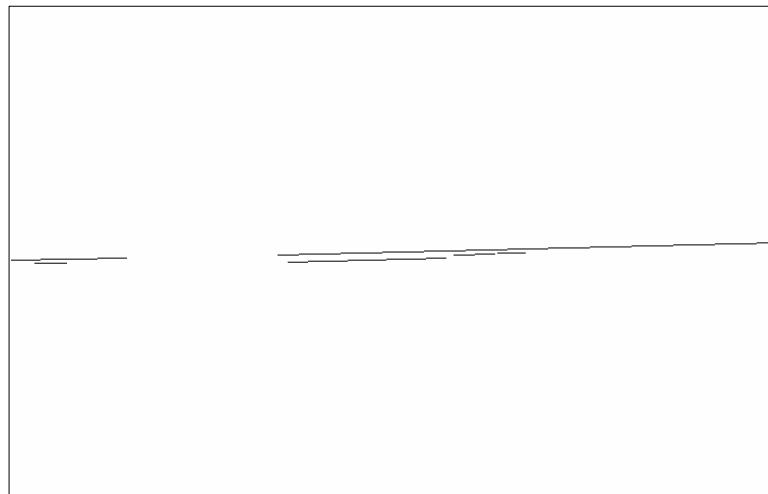
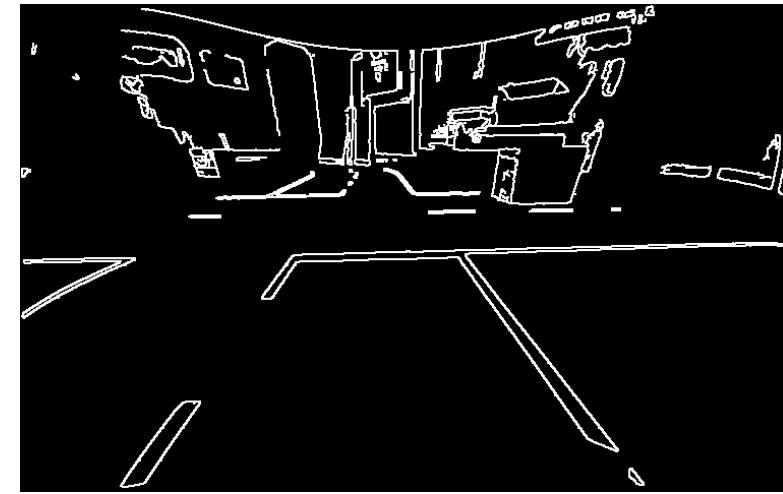
Entzerrtes Kamerabild

Kreuzungskonzept - Linienerkennung



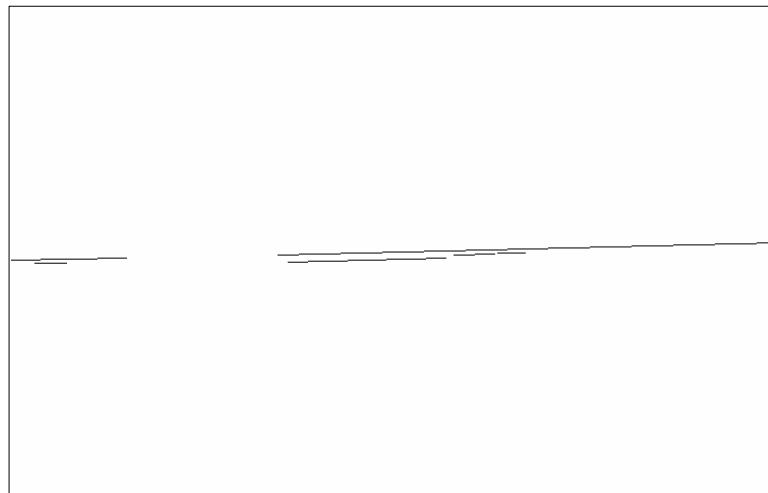
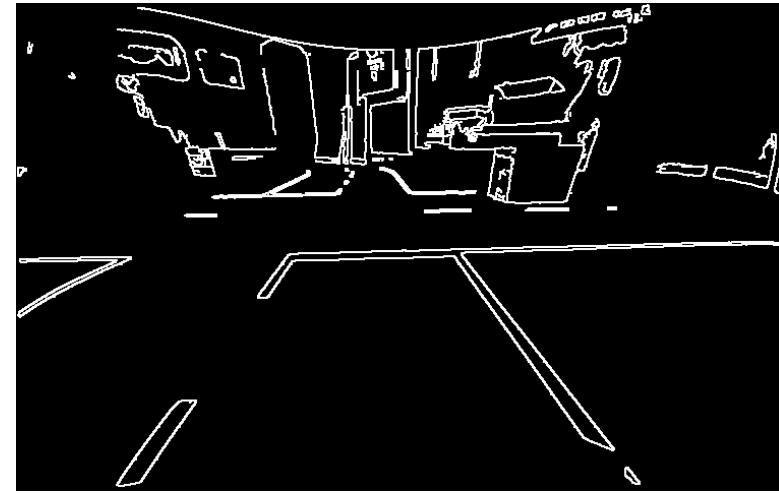
Canny Algorithmus sowie Dilatation

Kreuzungskonzept - Linienerkennung



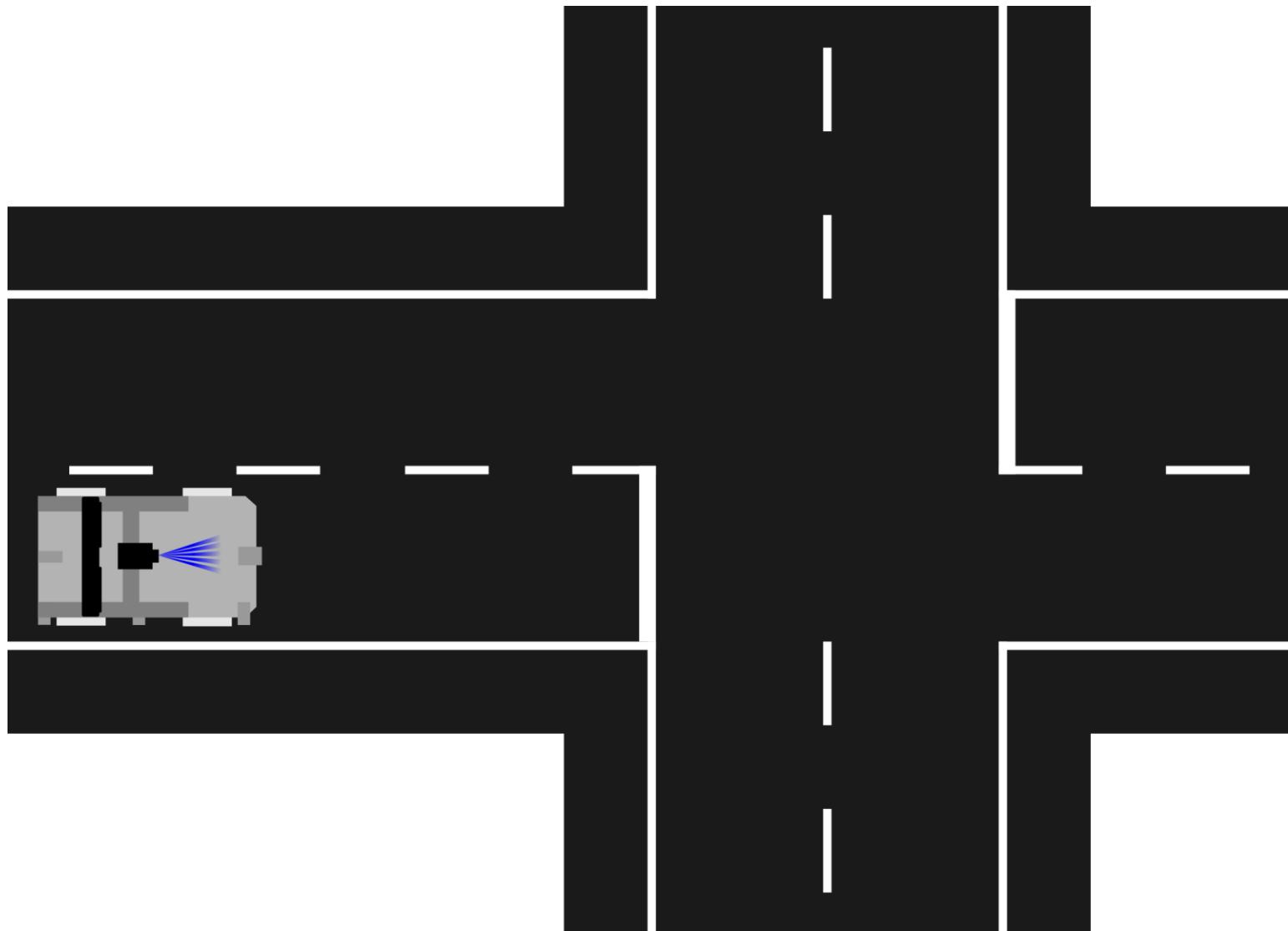
Line Segmentation Algorithm → Validierung über Winkel und Länge

Kreuzungskonzept - Linienerkennung

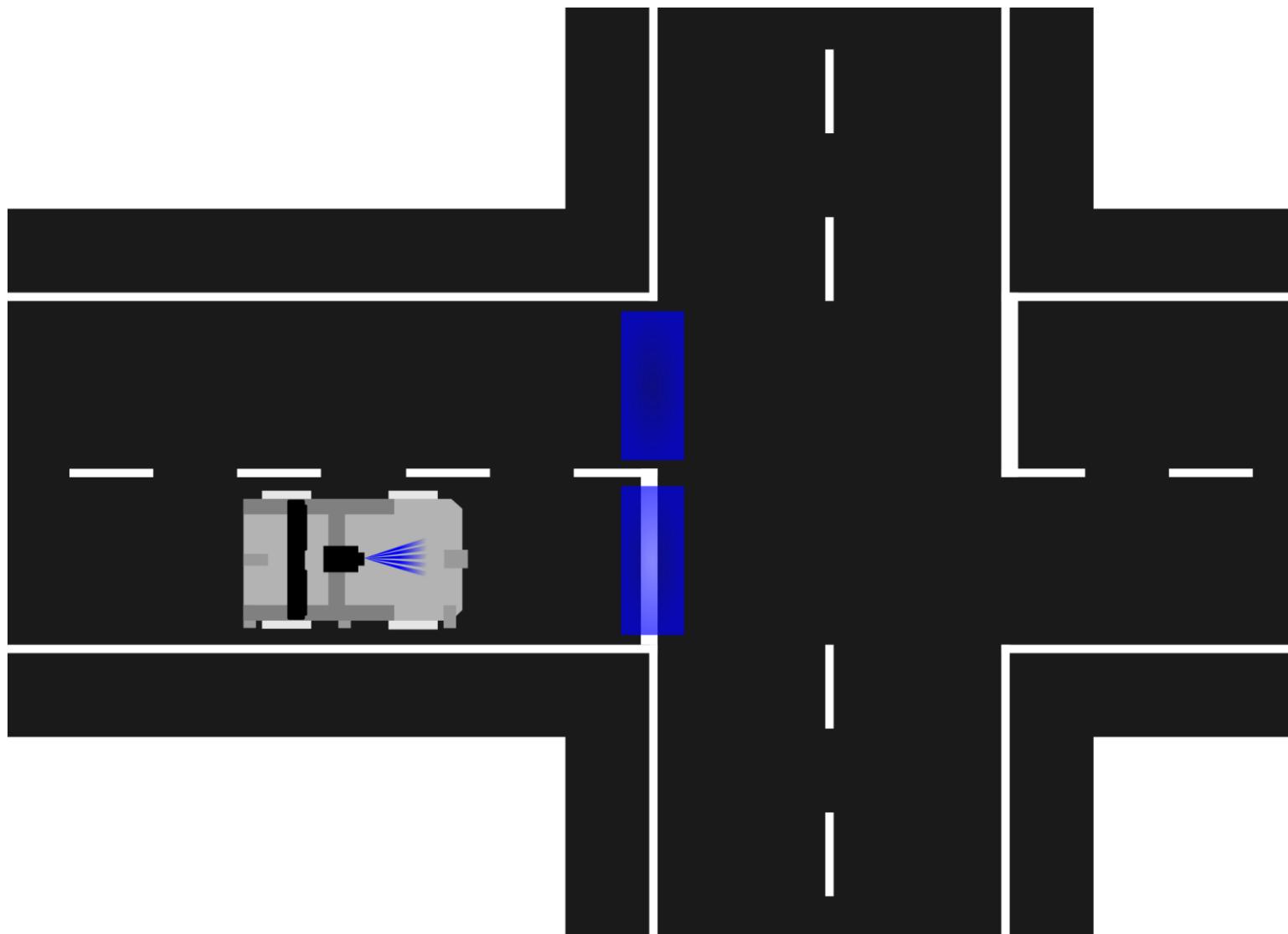


Erkannte Haltelinie

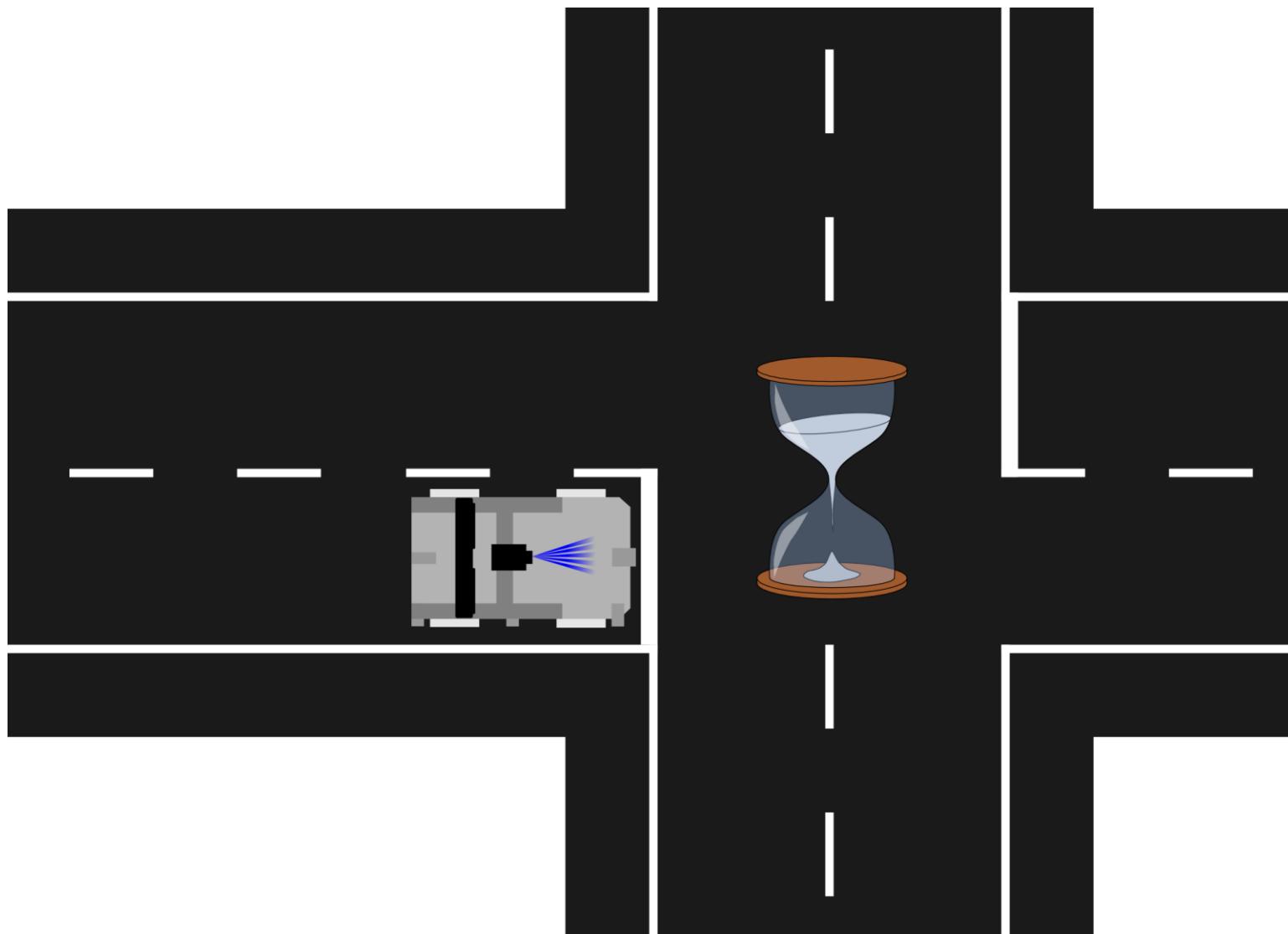
Kreuzungskonzept - Vorfahrtsberechtigte Fahrzeuge



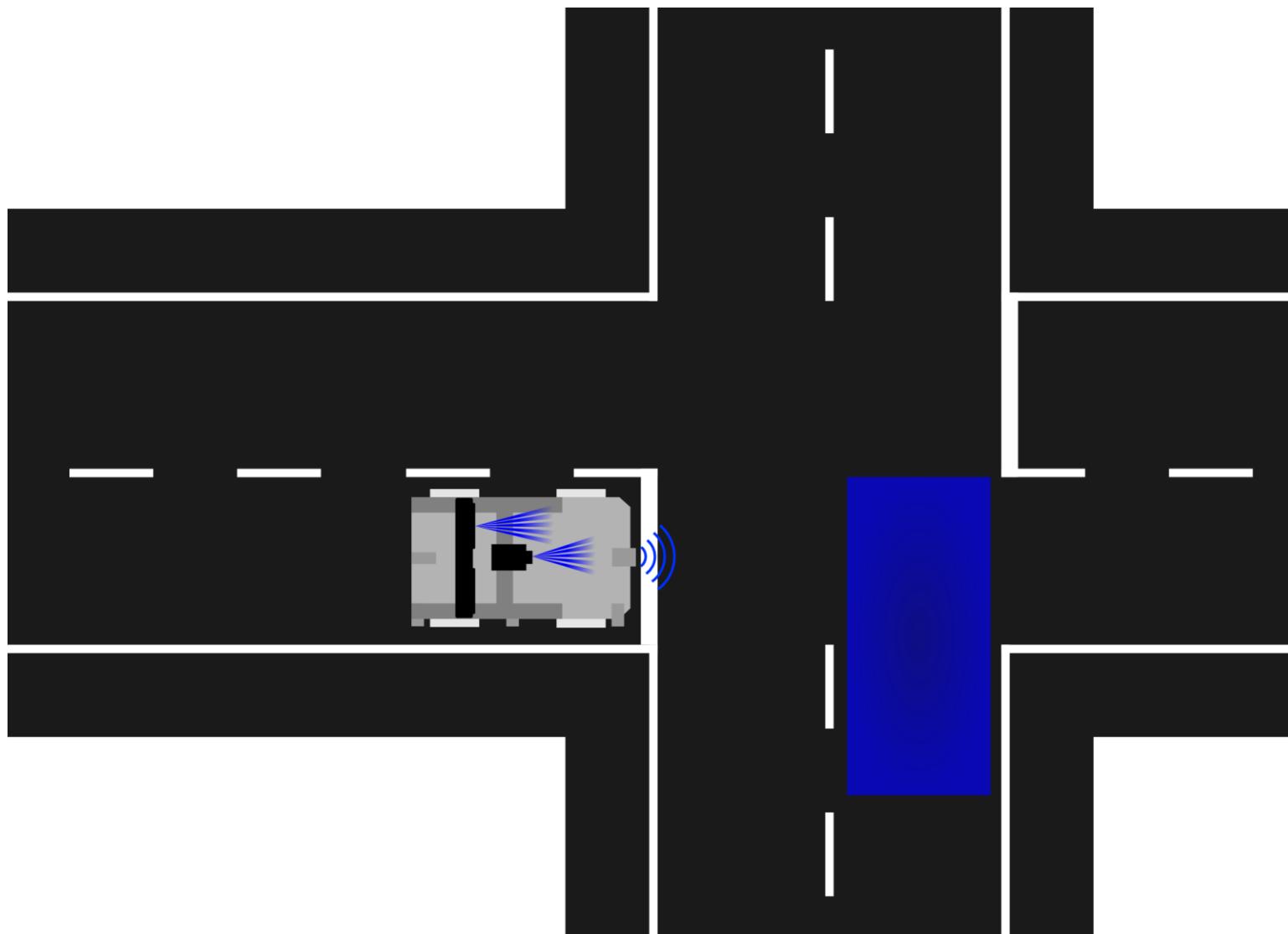
Kreuzungskonzept



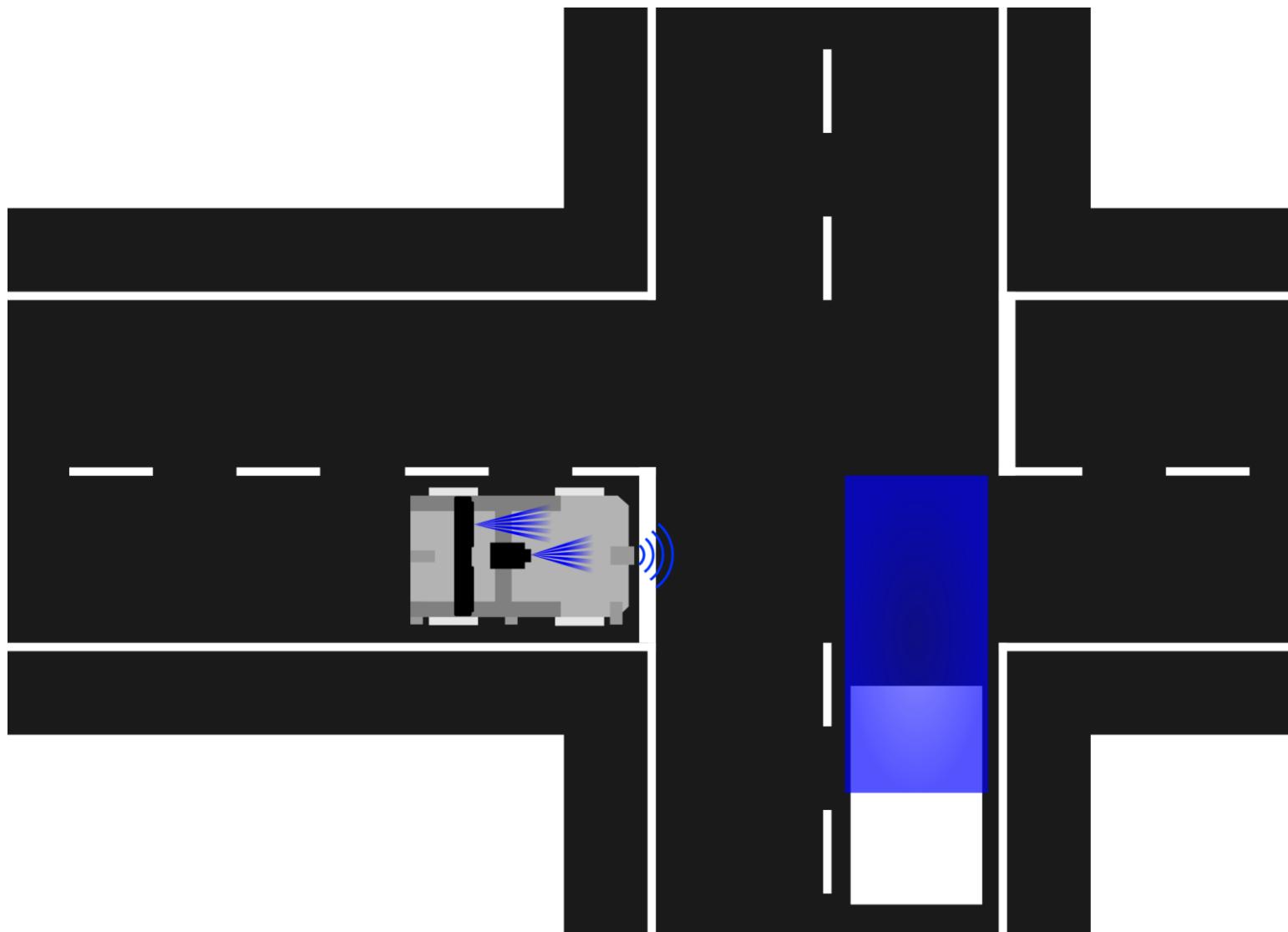
Kreuzungskonzept



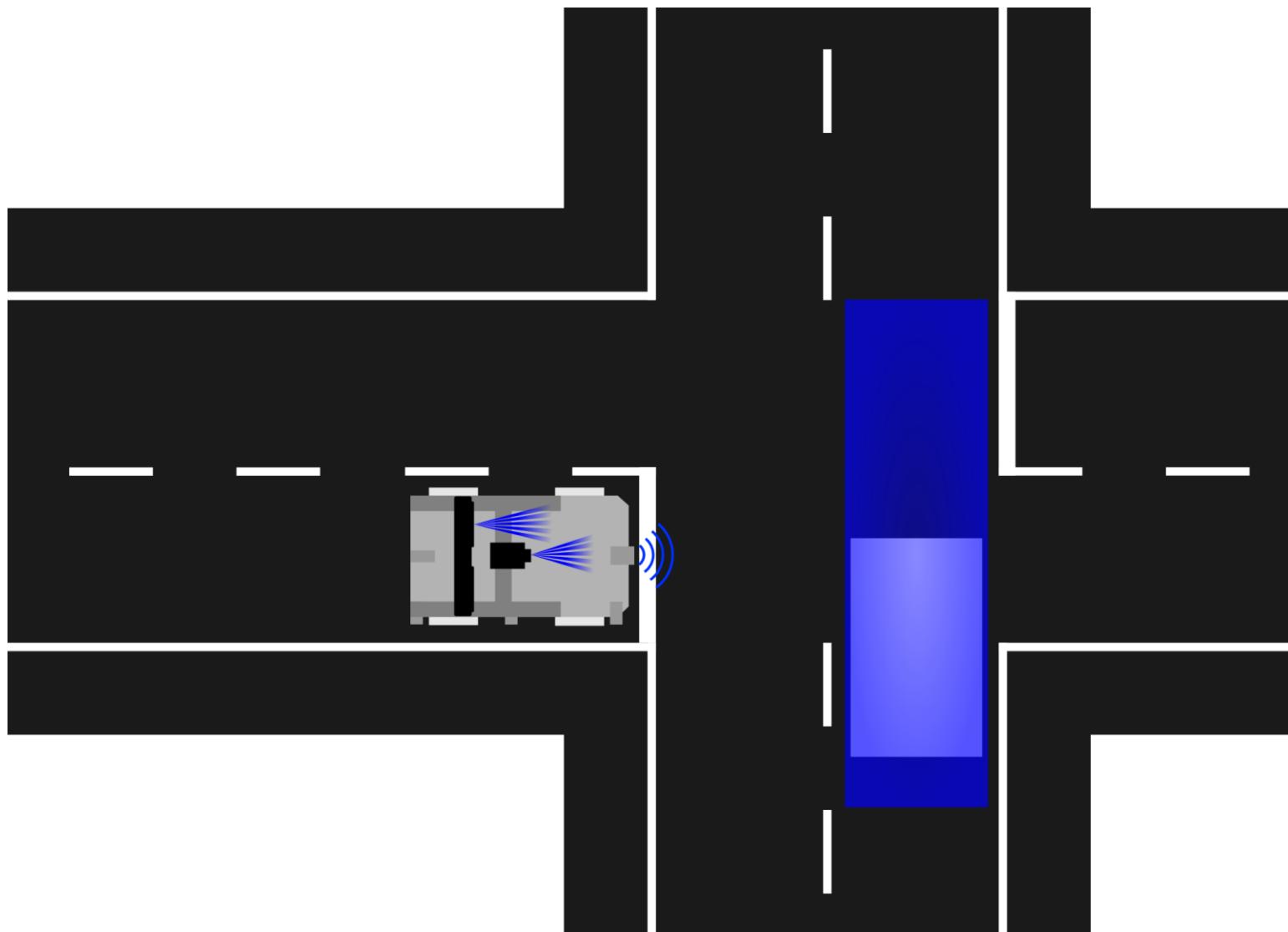
Kreuzungskonzept



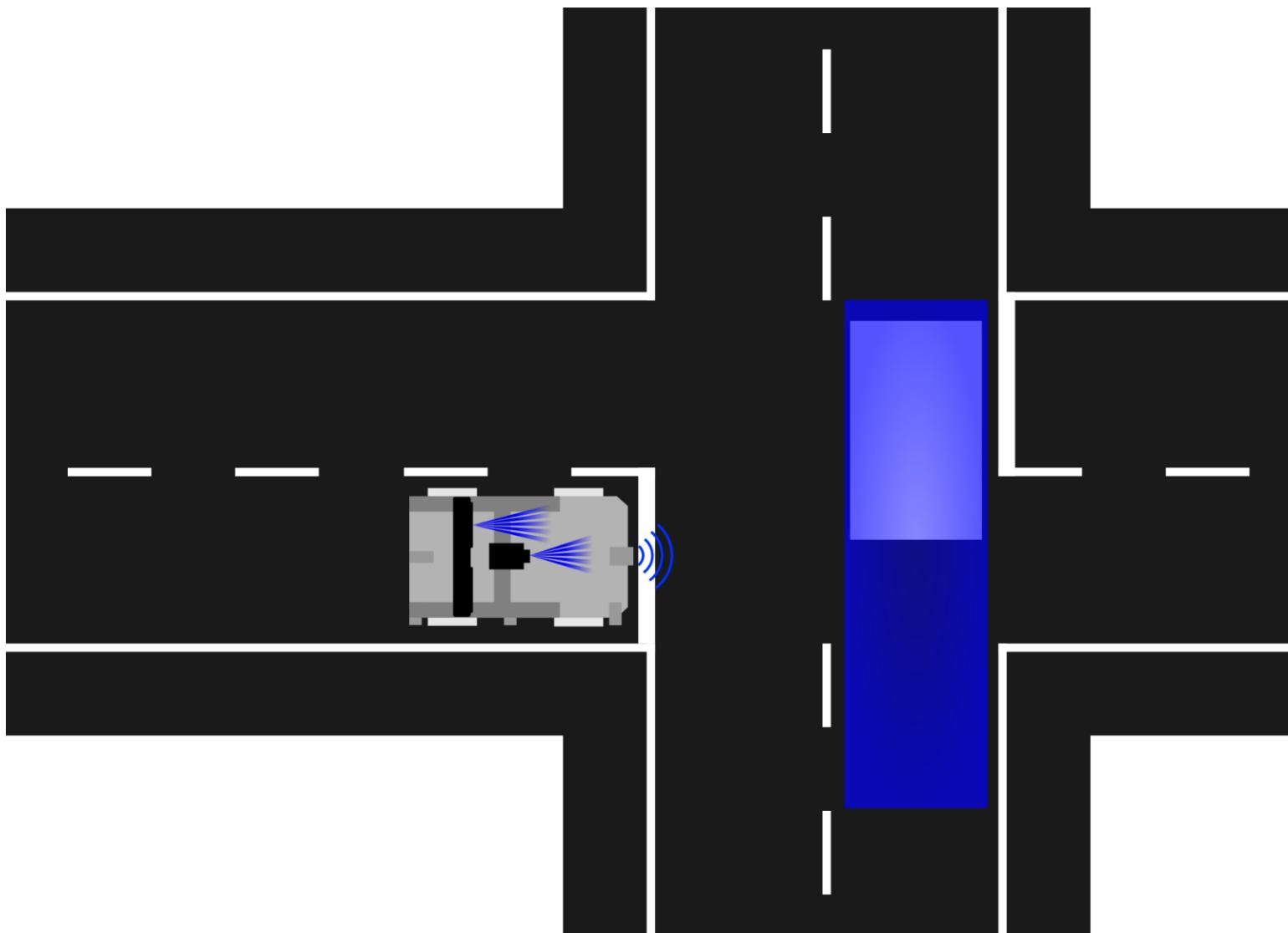
Kreuzungskonzept



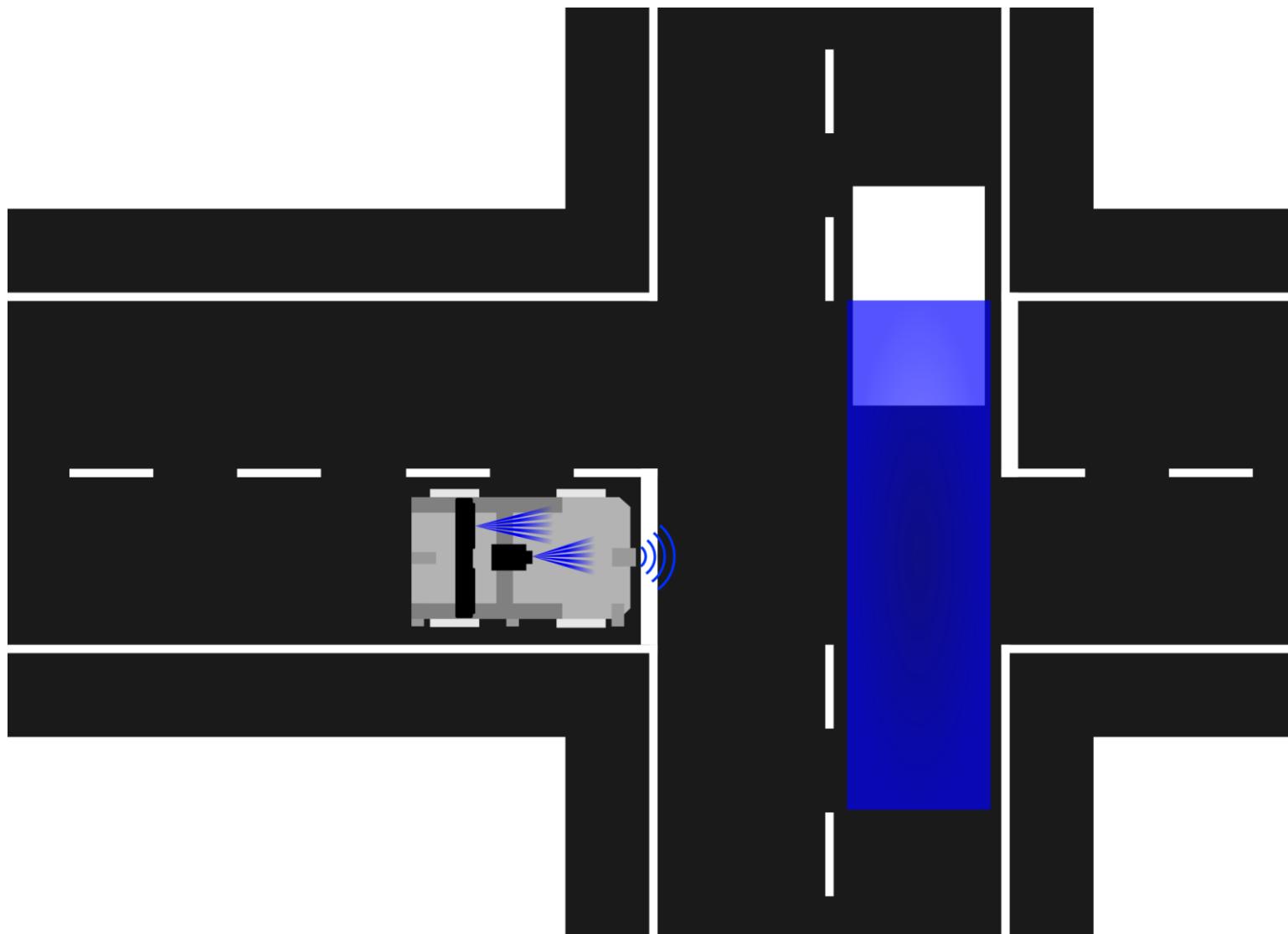
Kreuzungskonzept



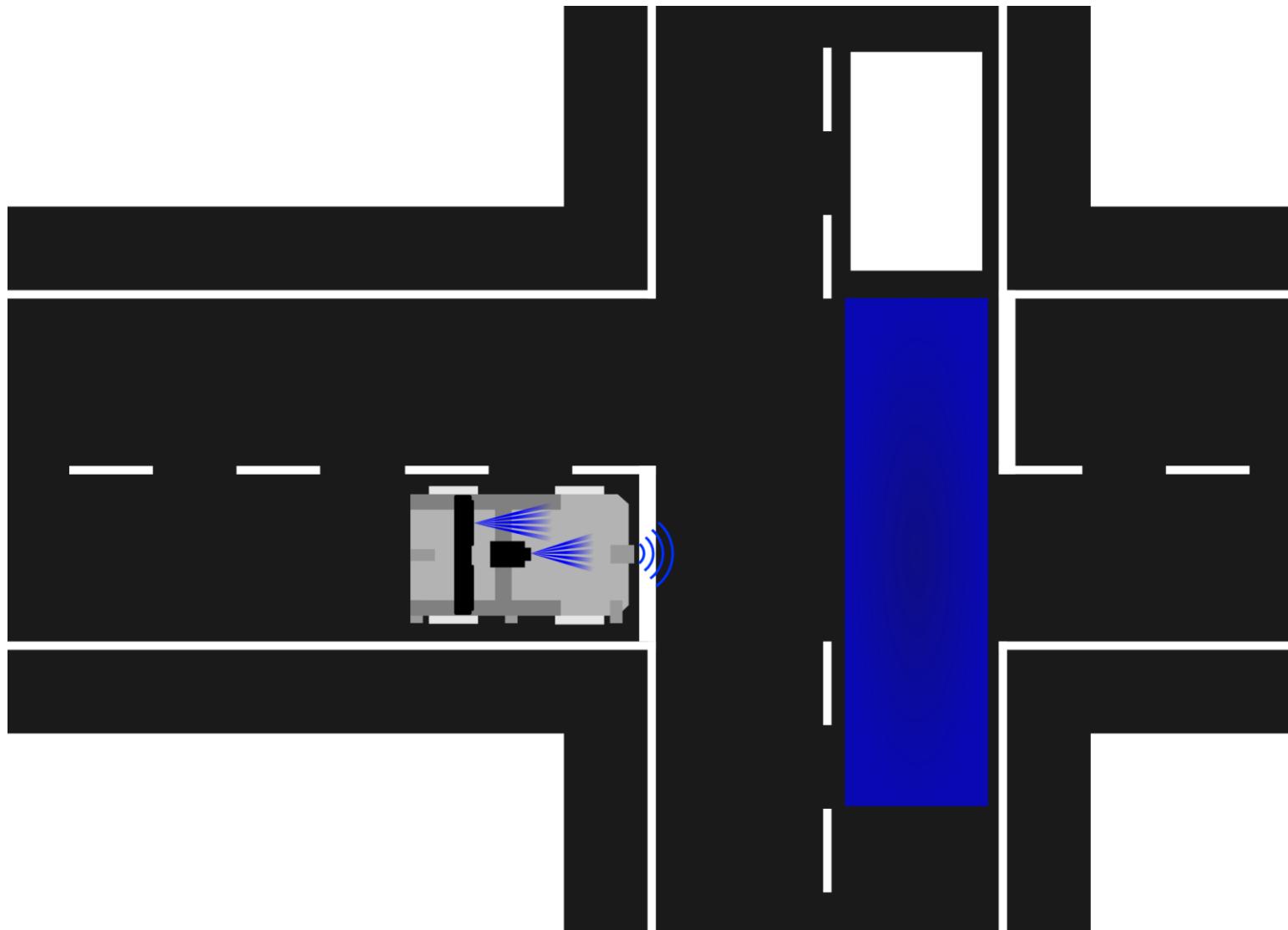
Kreuzungskonzept



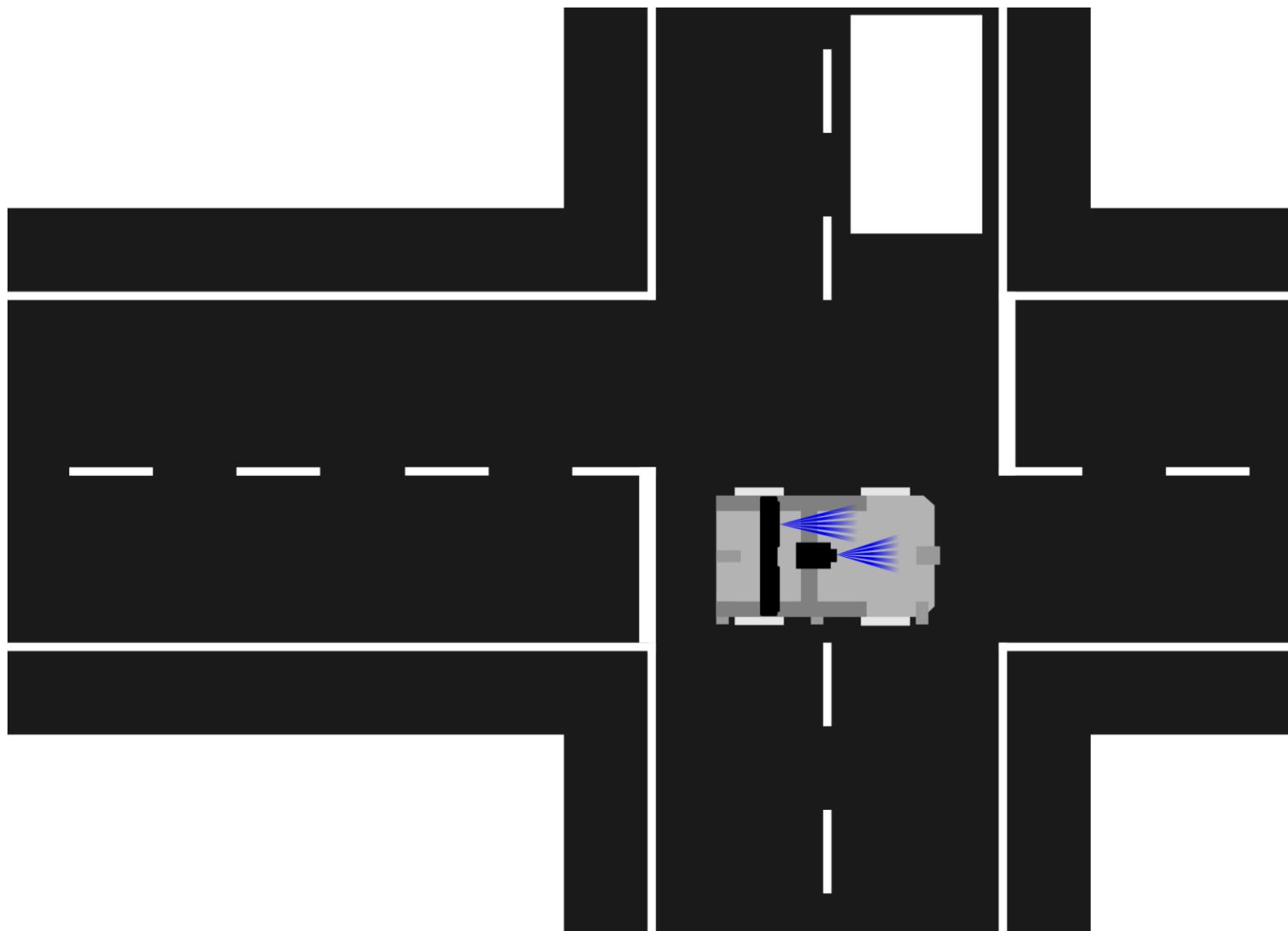
Kreuzungskonzept



Kreuzungskonzept



Kreuzungskonzept



Gliederung

Der Spatz

Rundkurs

Hindernisfahrt

Einparken

Sensoren zur Lückenfindung

Xtion

Exakte
Lokalisierung

Hohe Framerate

Probleme mit
kleinem Hindernis

Ultraschall

Einfache
Ansteuerung

Messungenauigkeit

Winkelabhängigkeit

Infrarot

Geringe
Abmessungen

Hohe Tastrate

Reflexionen

Lückenfindungskonzept

Kartierung mit „Gridmap“

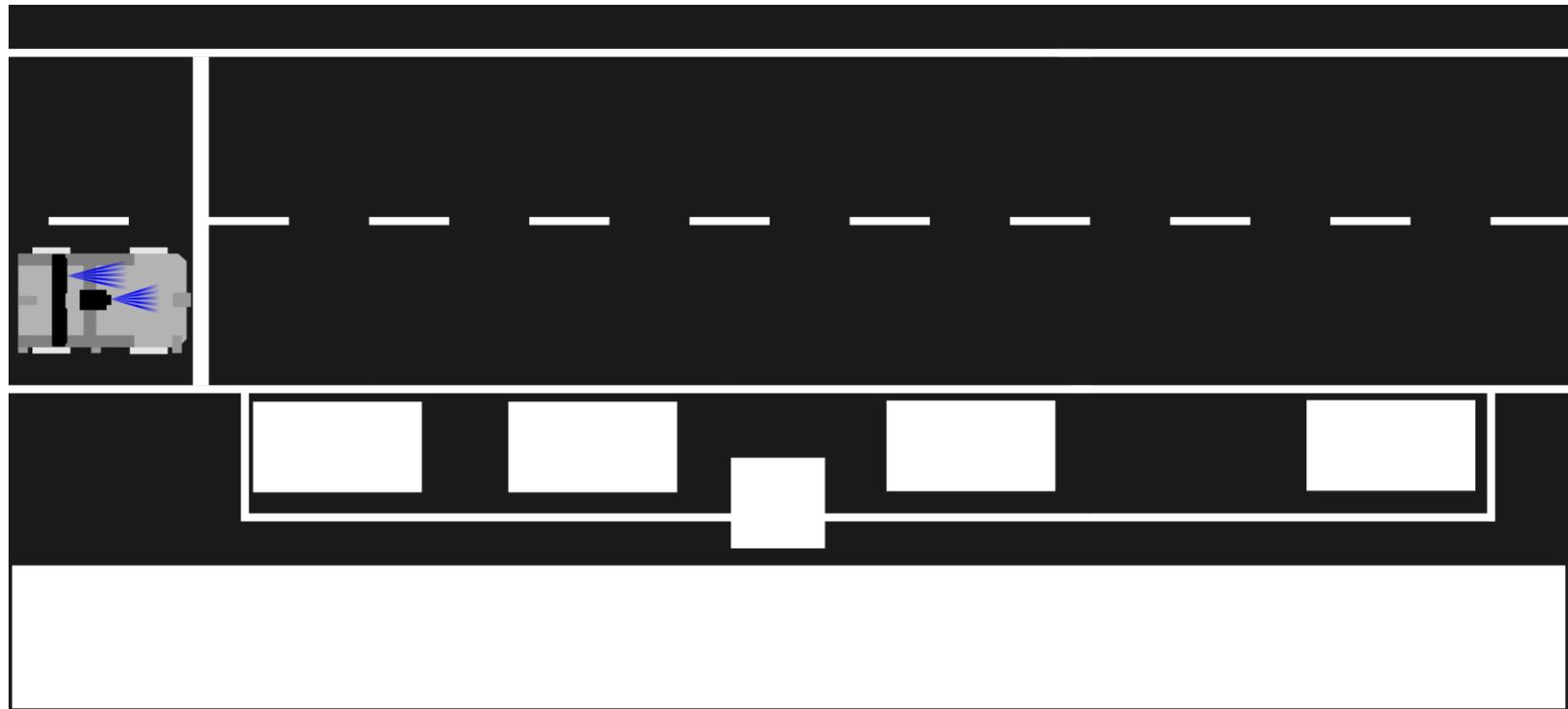
Sensorfusion

- Erstellung der „Gridmap“ mit allen Sensoren
- Erweiterung um zusätzliche Sensoren leicht realisierbar

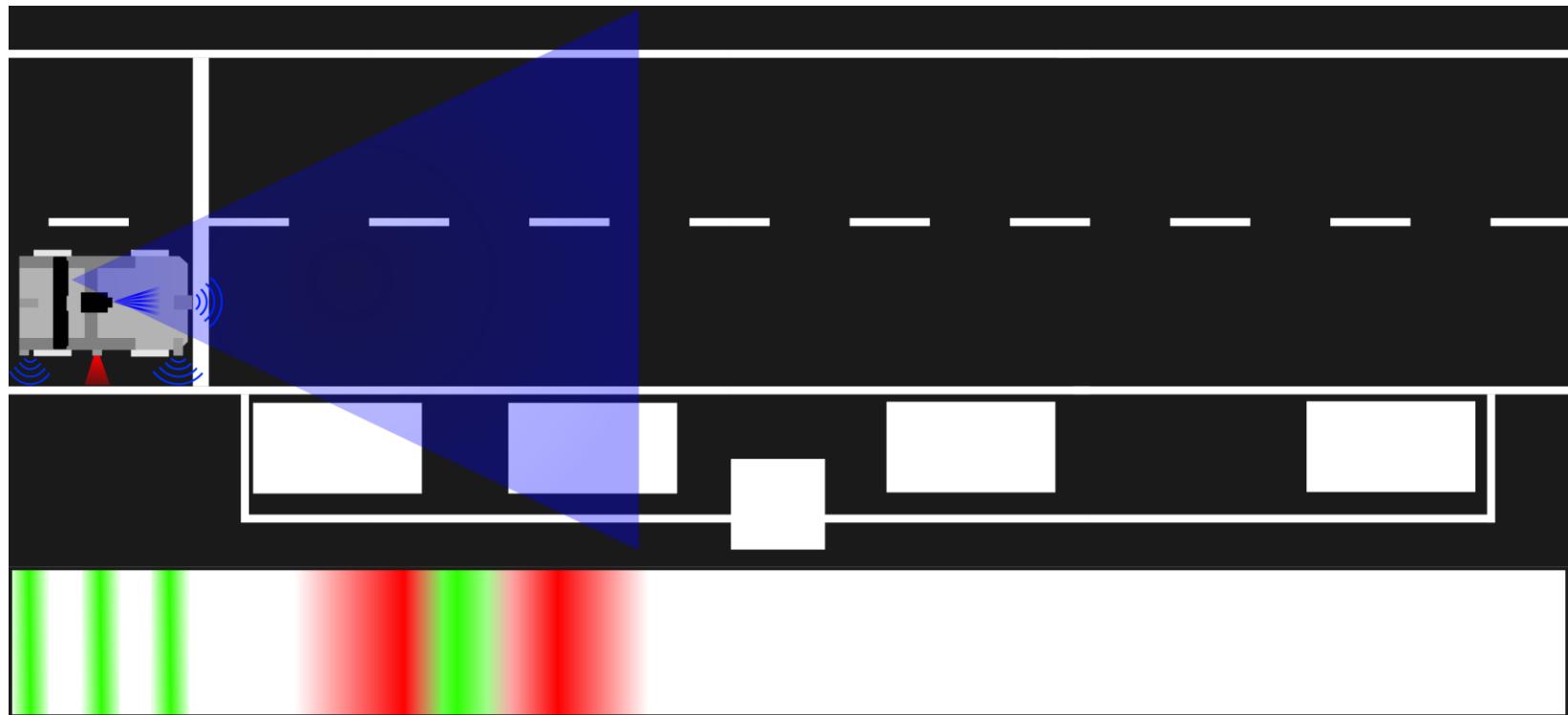
Vorteile

- Unsicherheit sinkt mit Anzahl der Sensoren
- „Gridmap“ eindimensional, da gerade Strecke
- Geringer Speicherbedarf, da eindimensional
- Robustheit

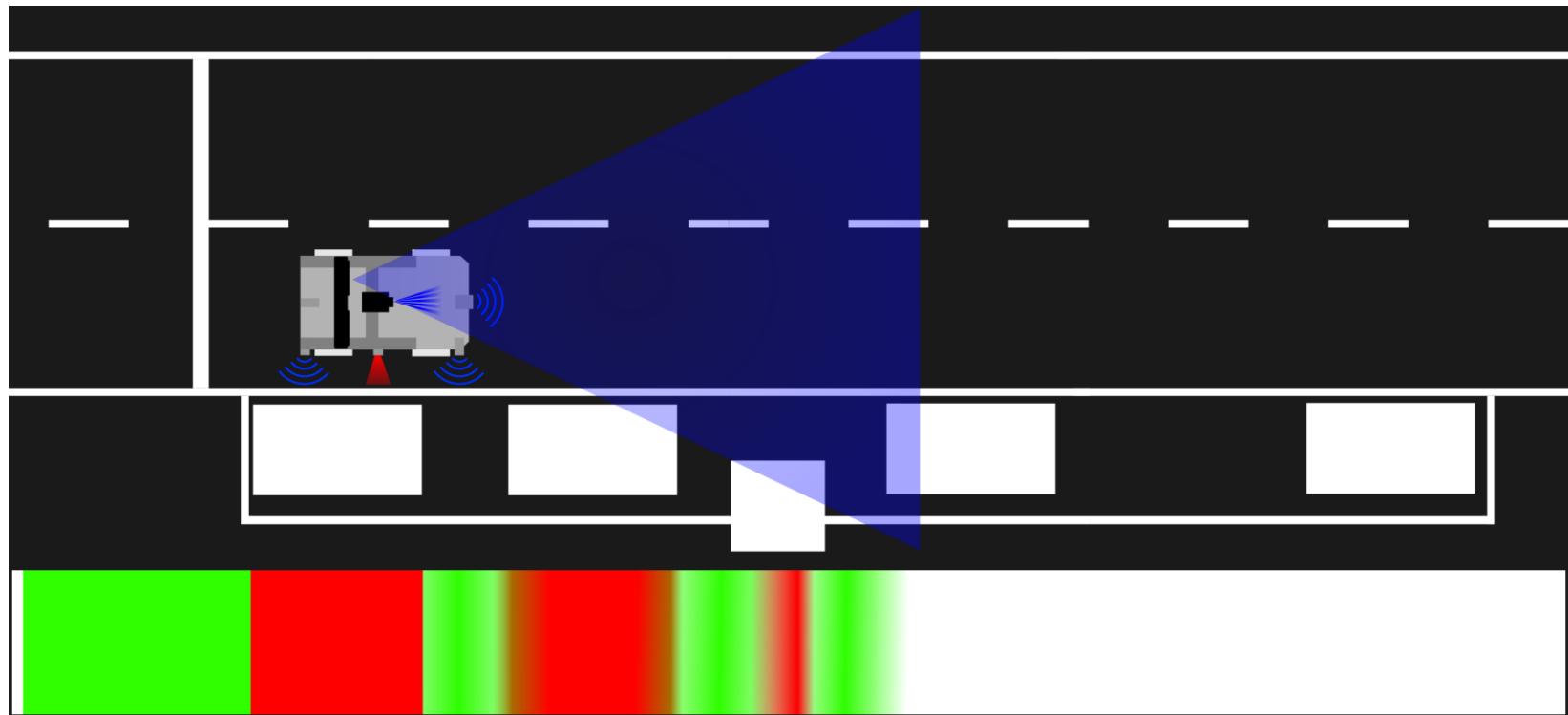
Entstehung der Gridmap



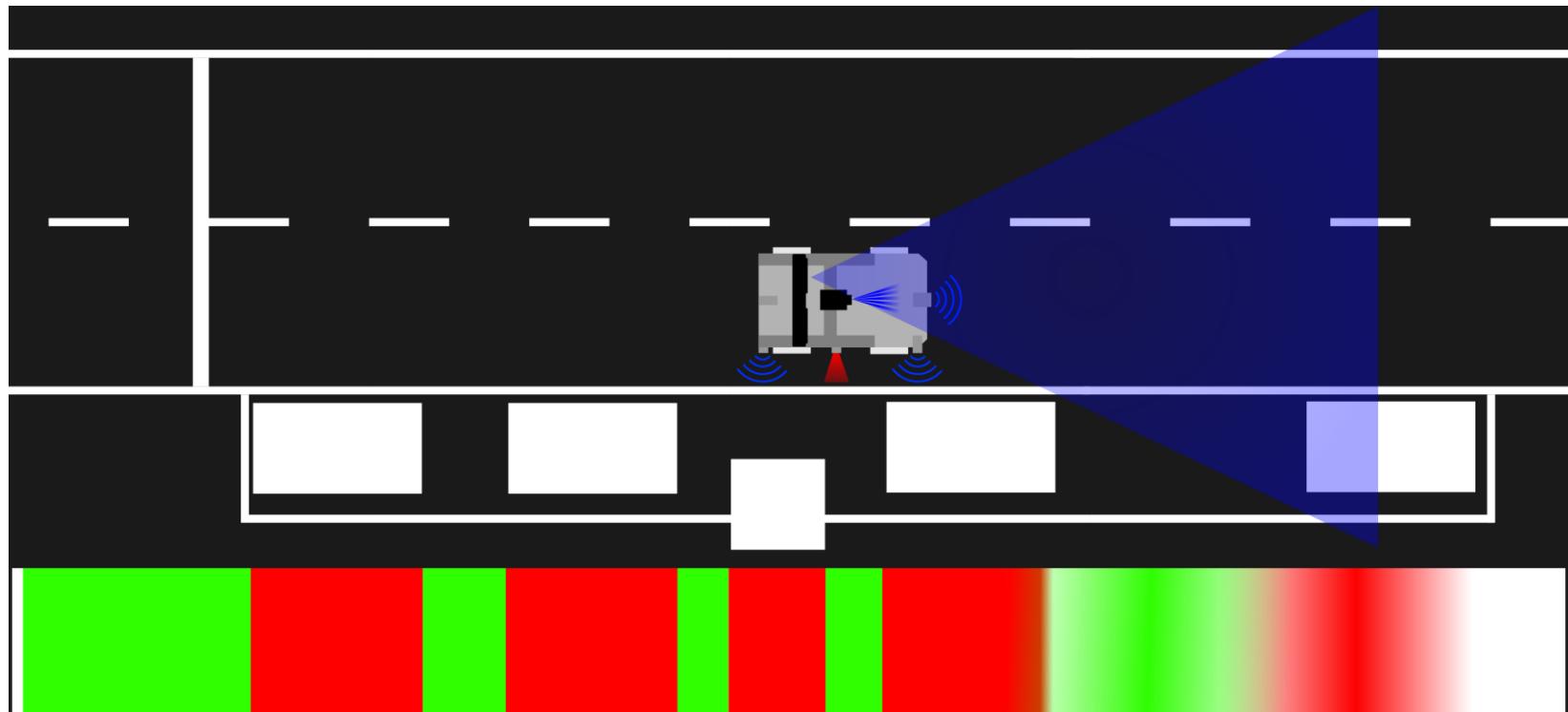
Entstehung der Gridmap



Entstehung der Gridmap



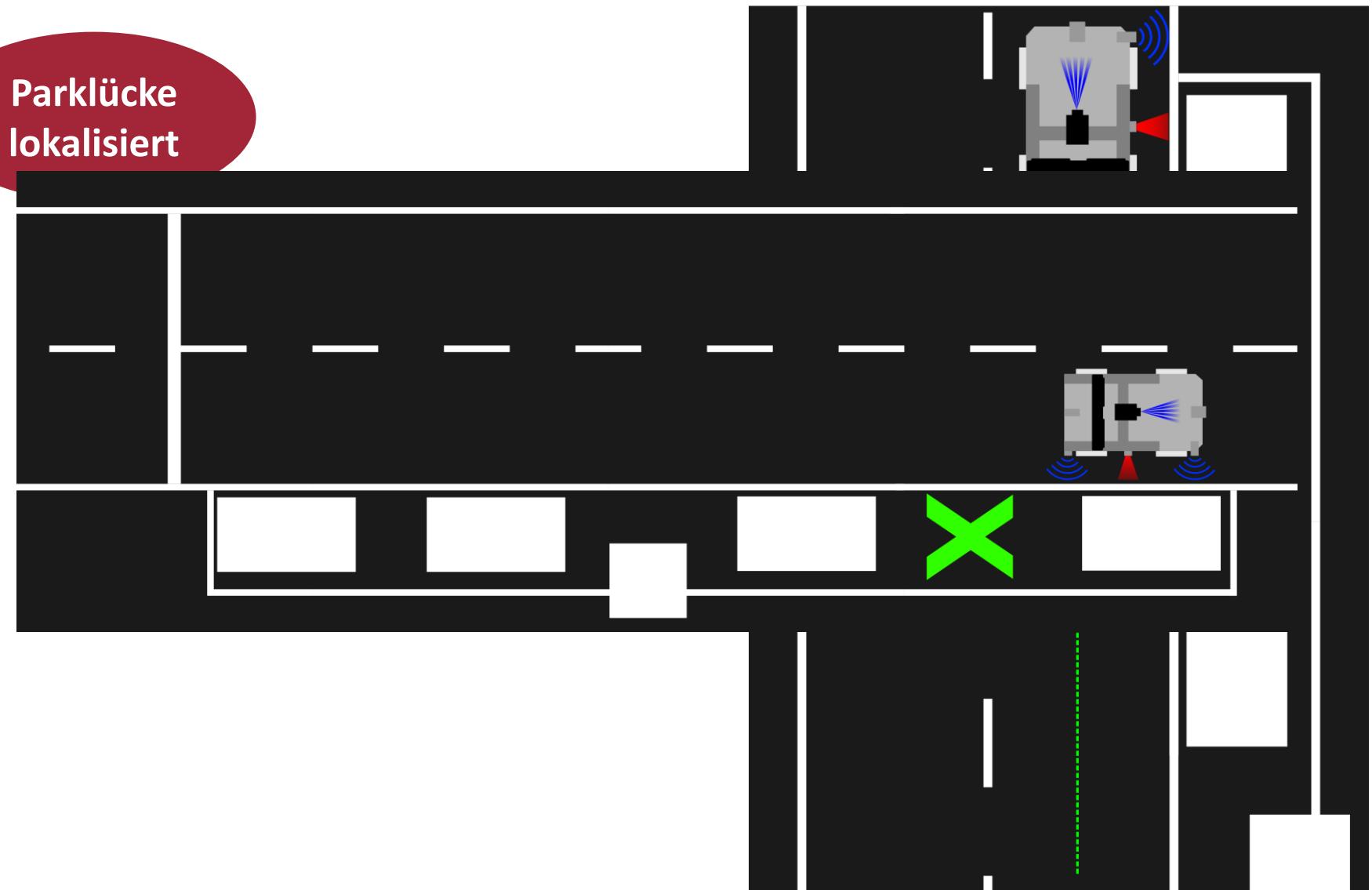
Entstehung der Gridmap



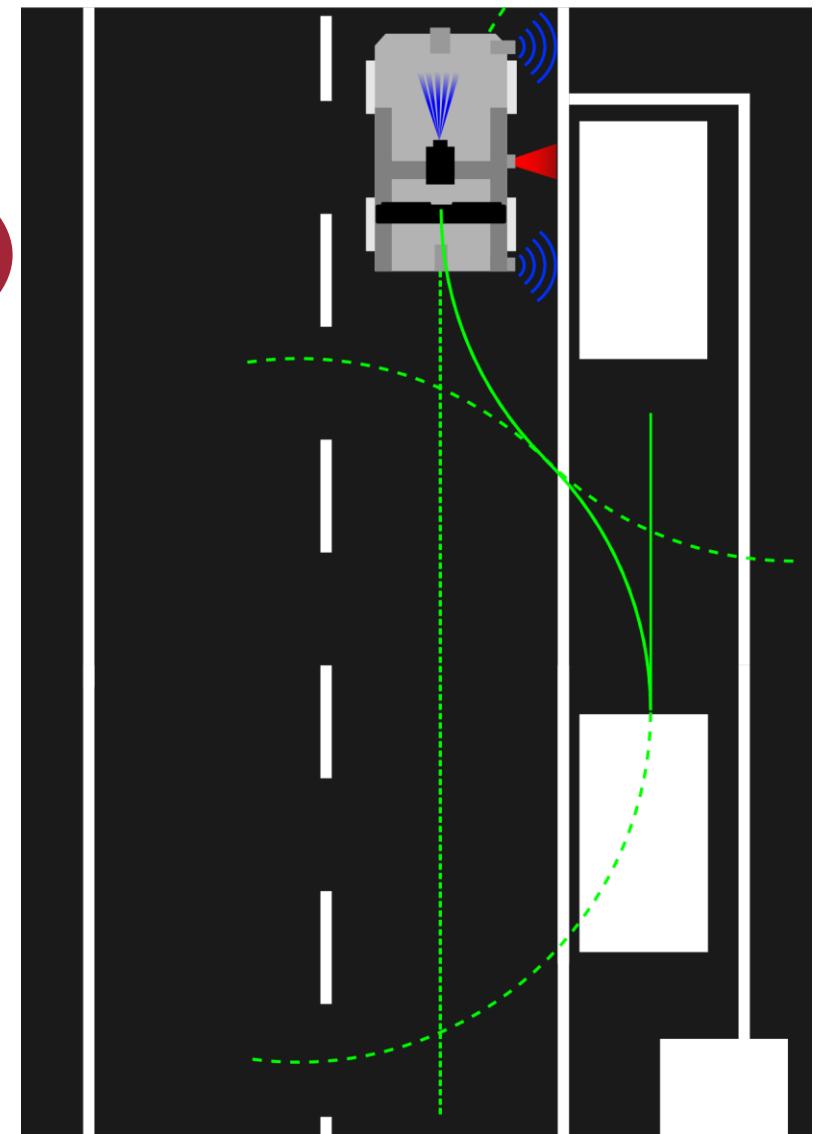
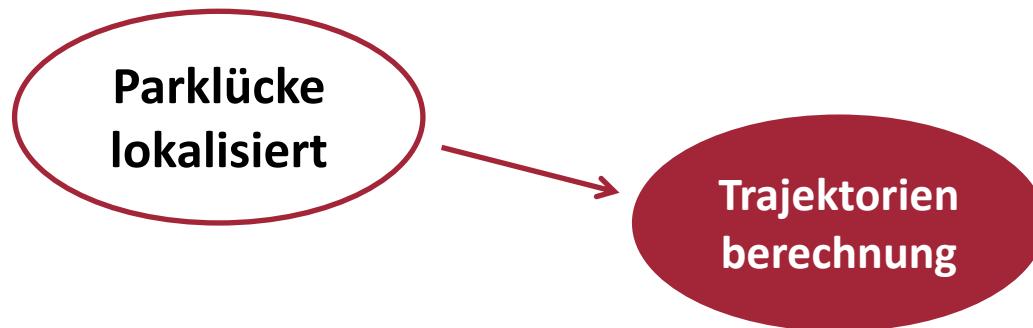
Entstehung der Gridmap



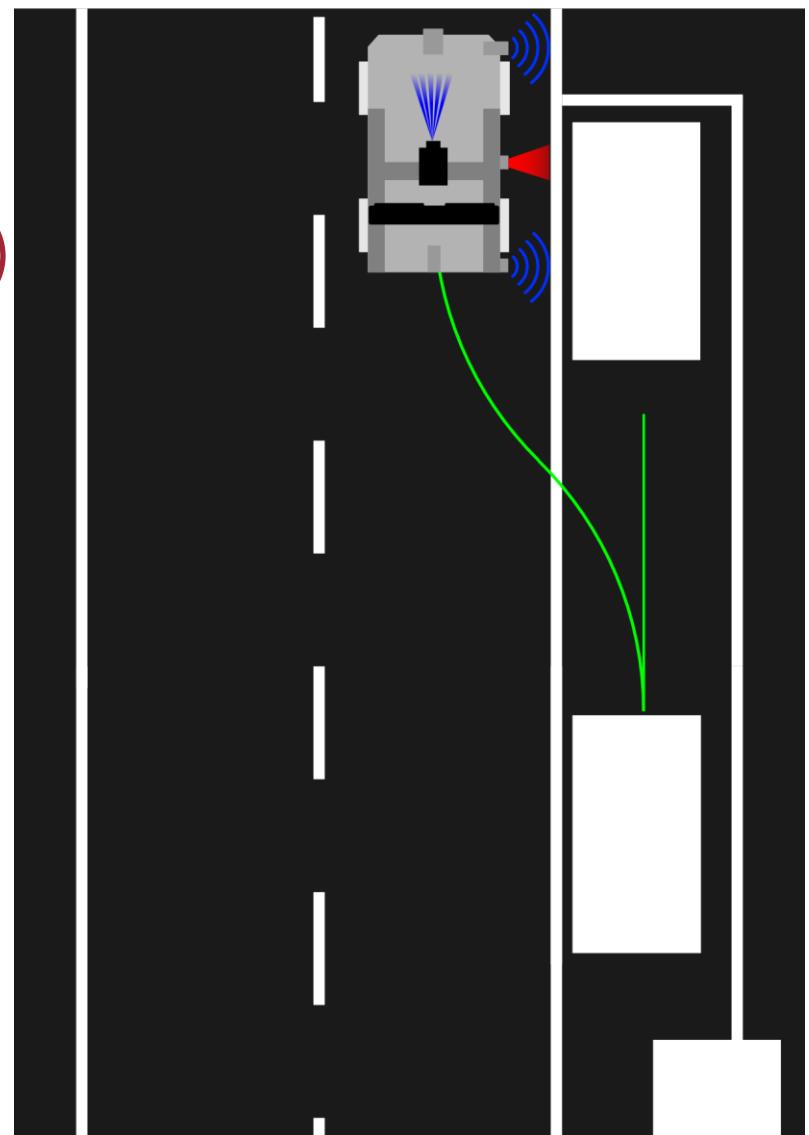
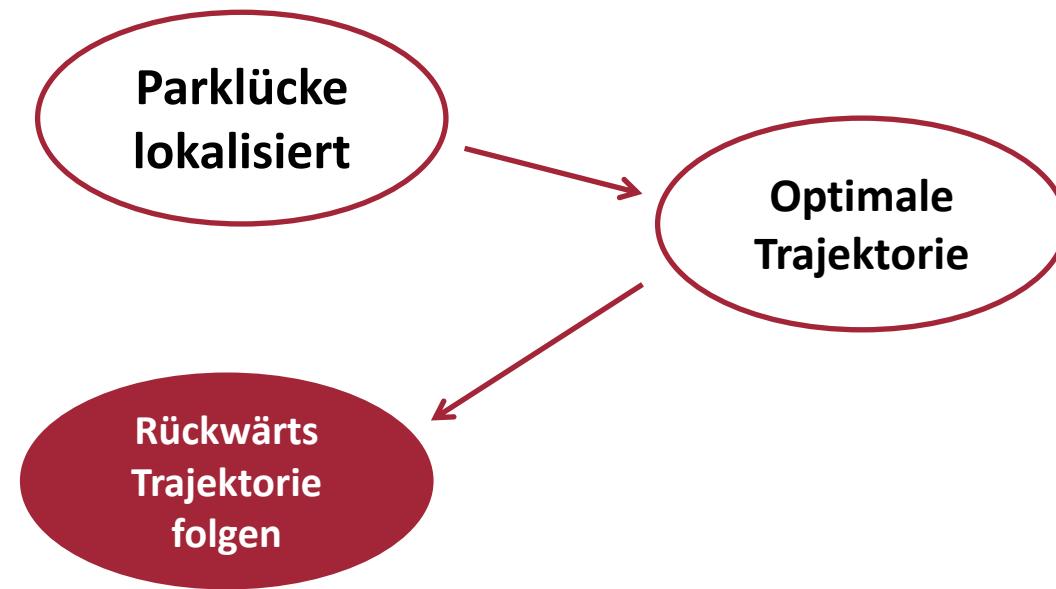
ErfolgsErfolg Gridmap



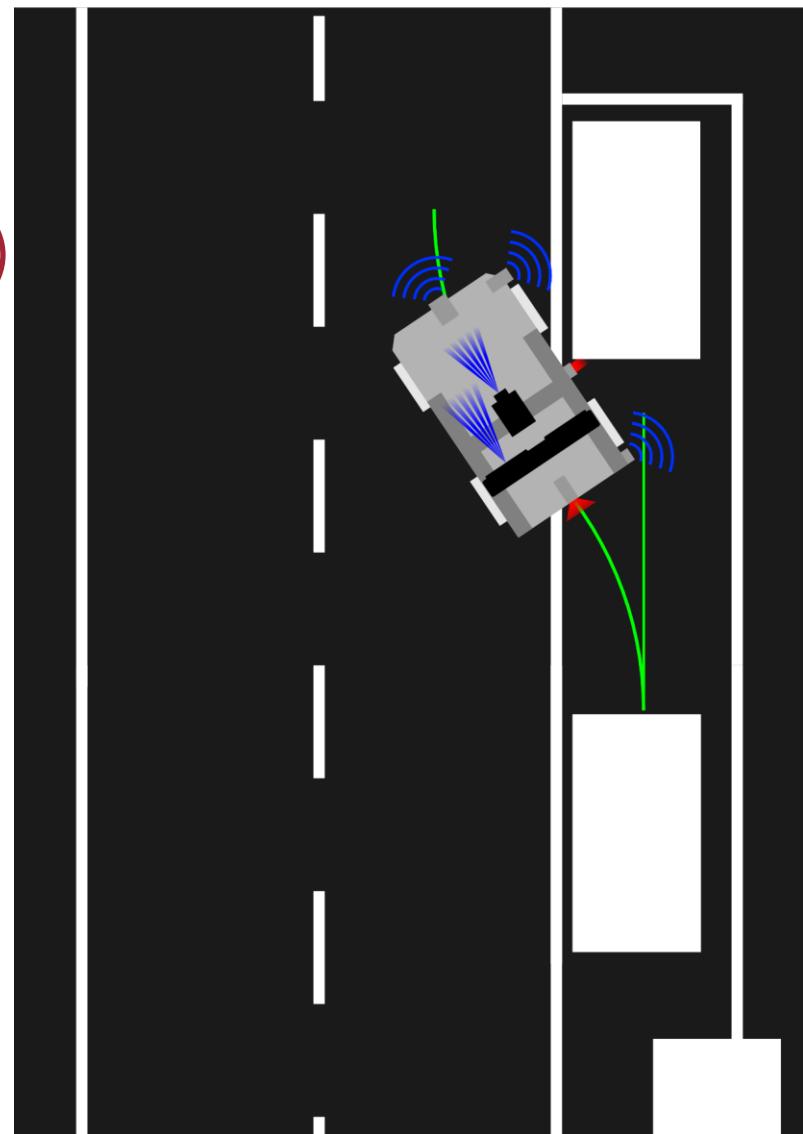
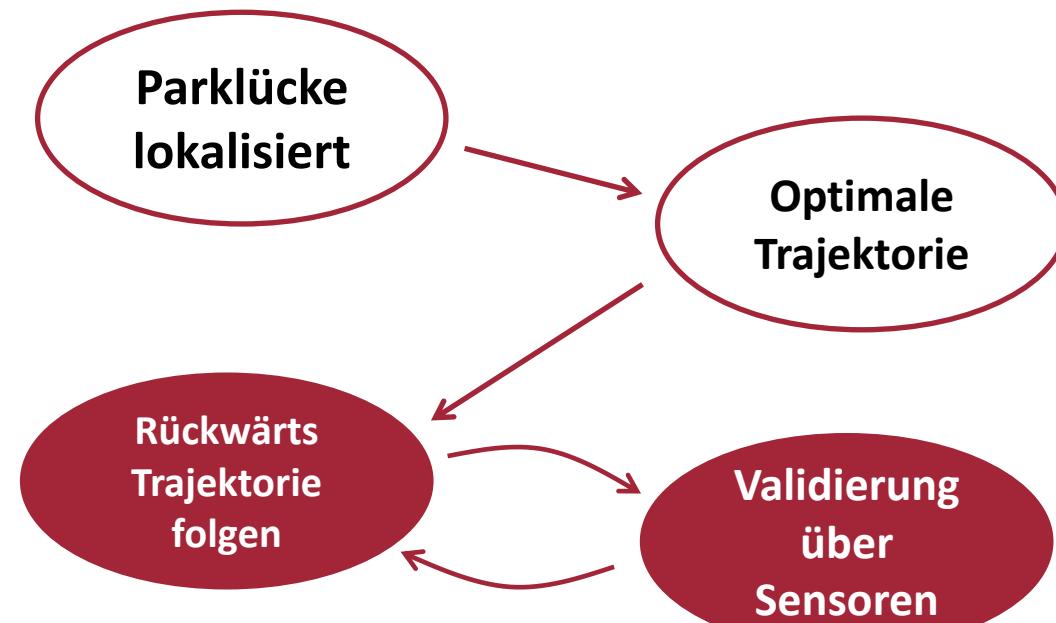
Der Einparkzug



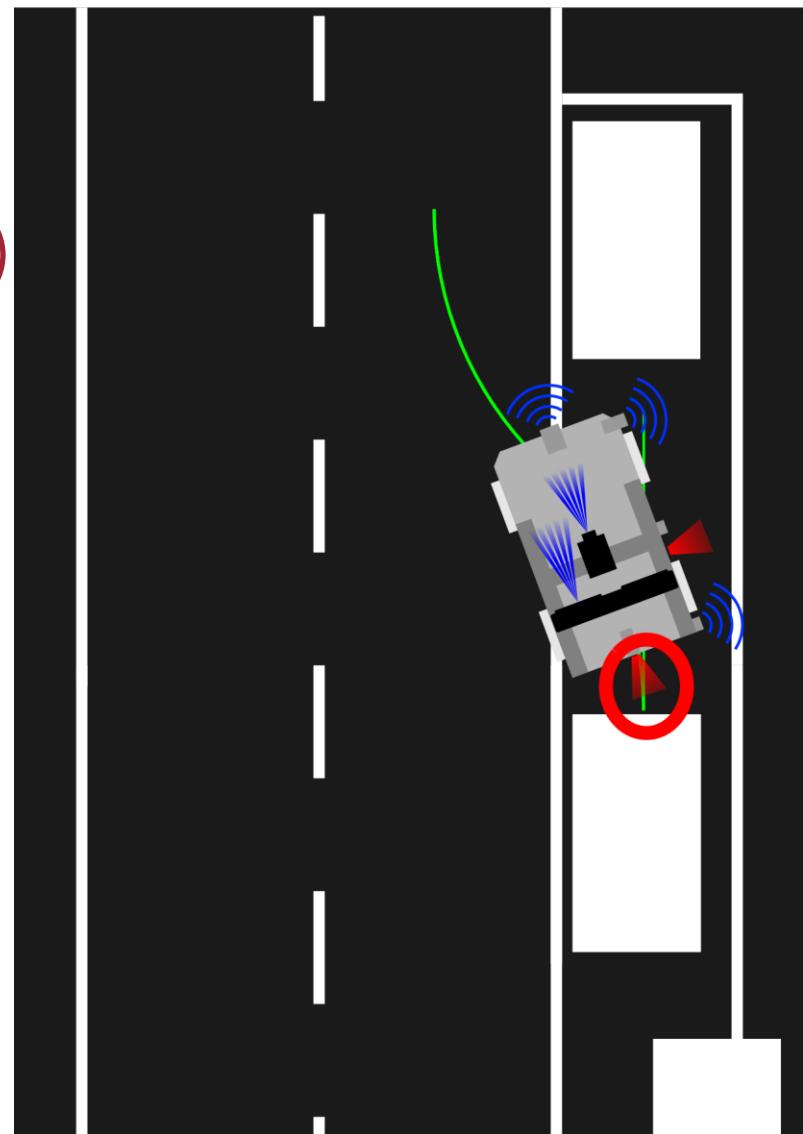
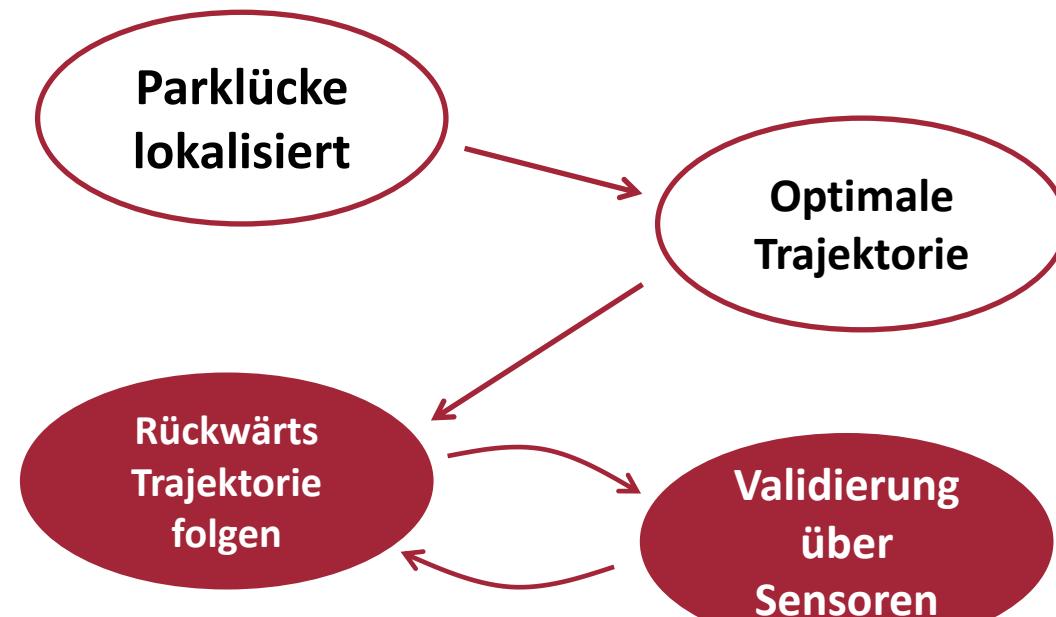
Der Einparkzug



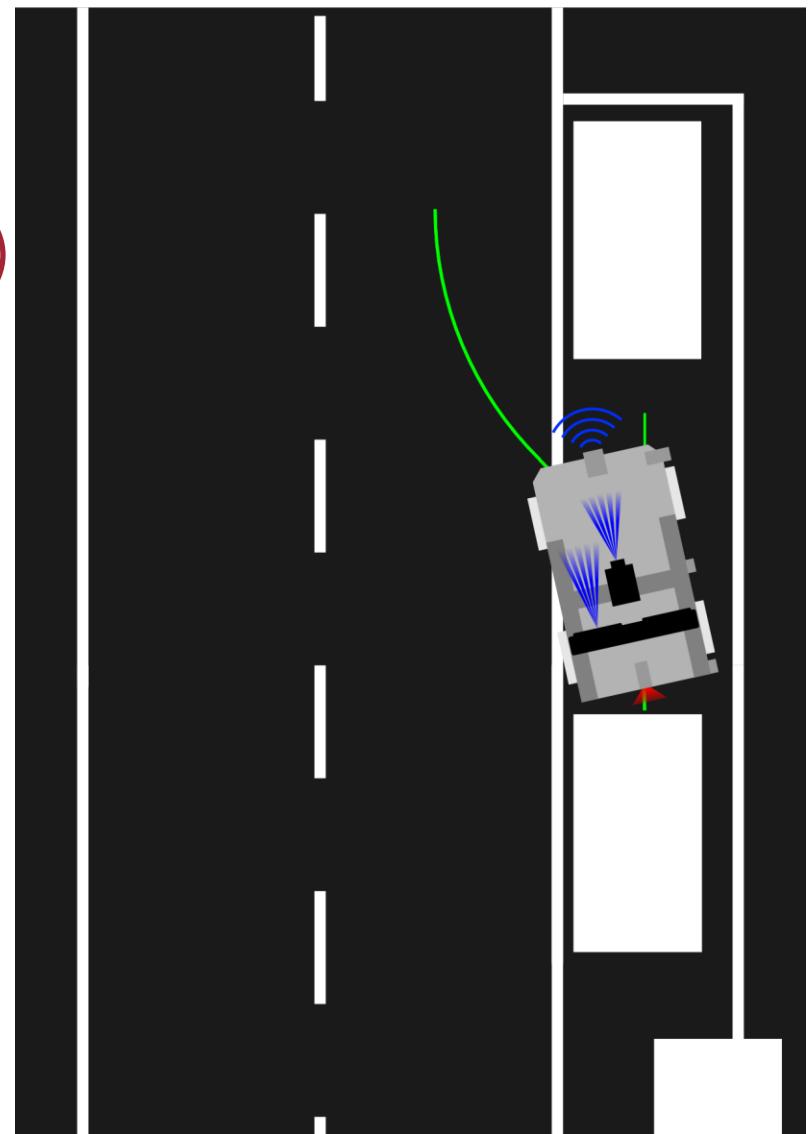
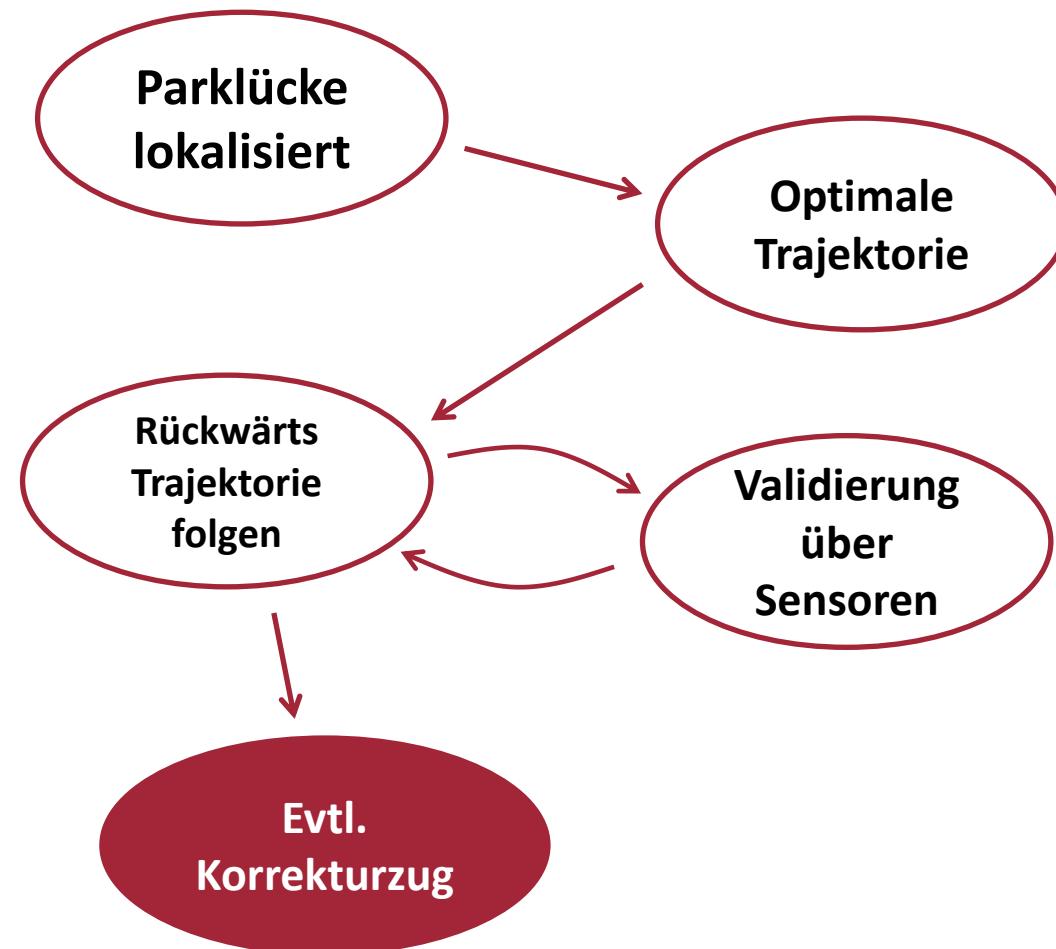
Der Einparkzug



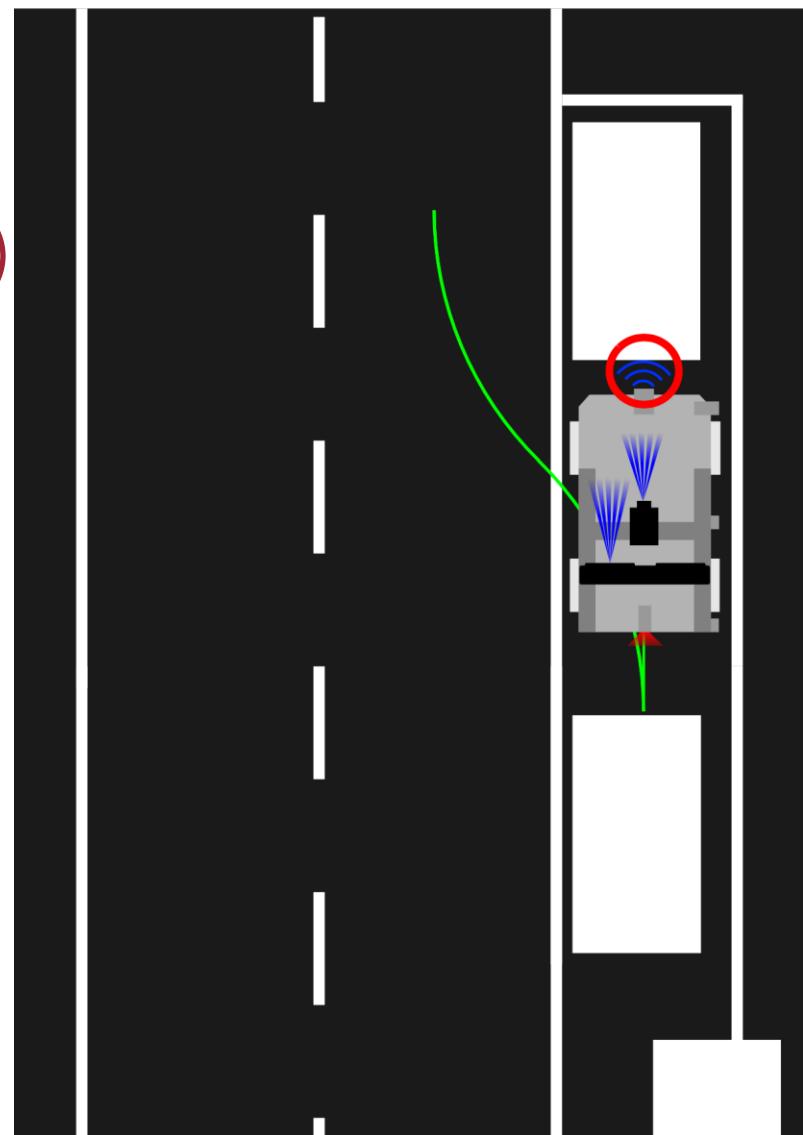
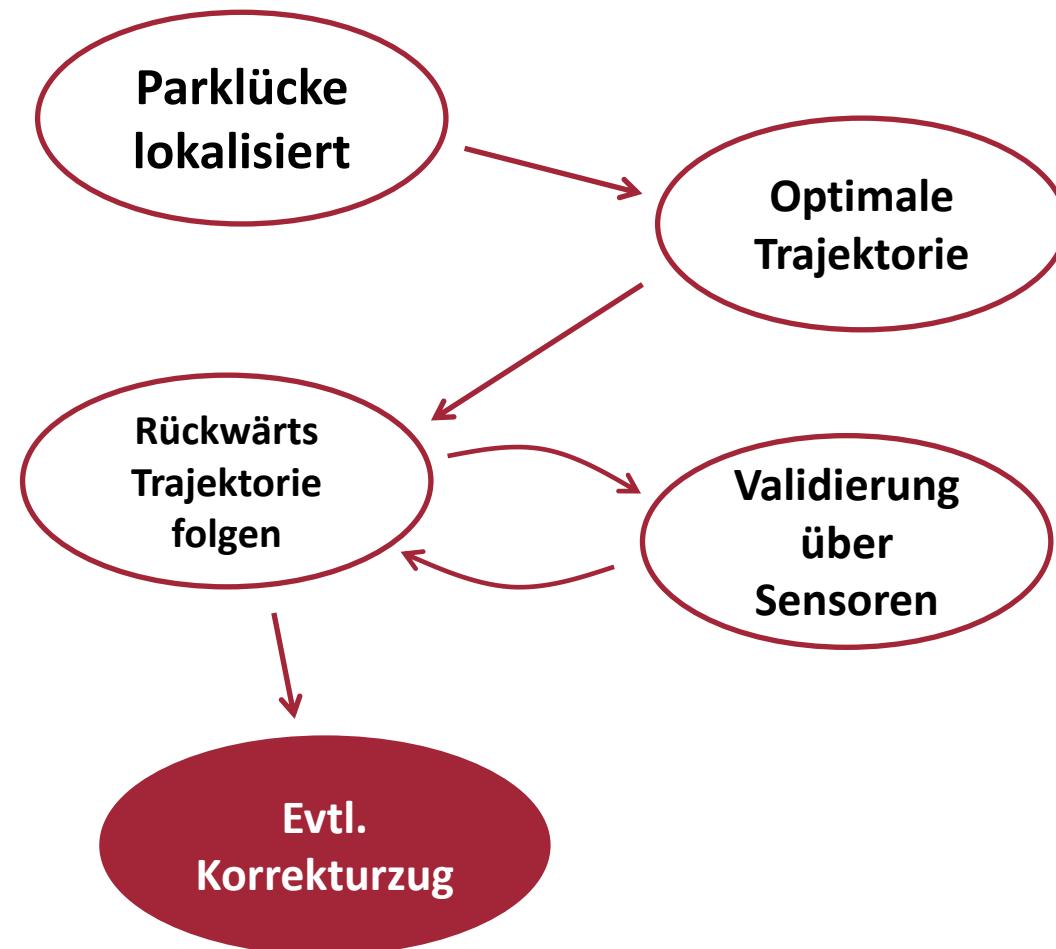
Der Einparkzug



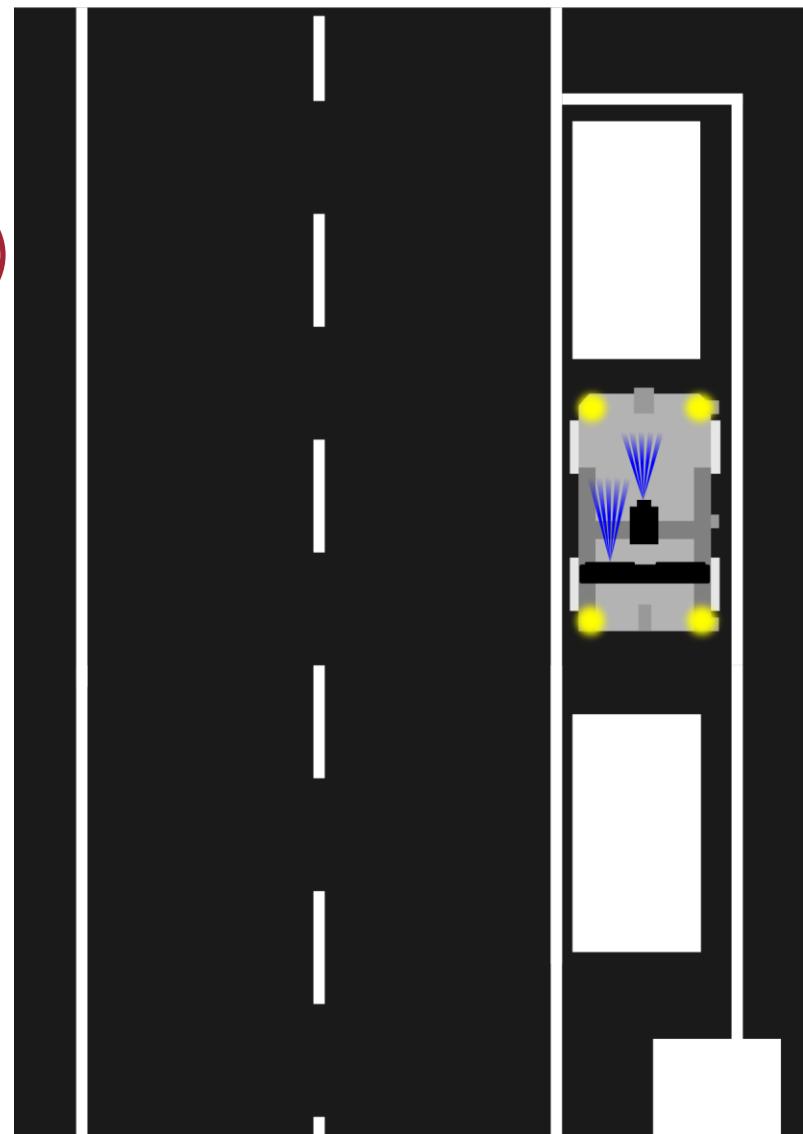
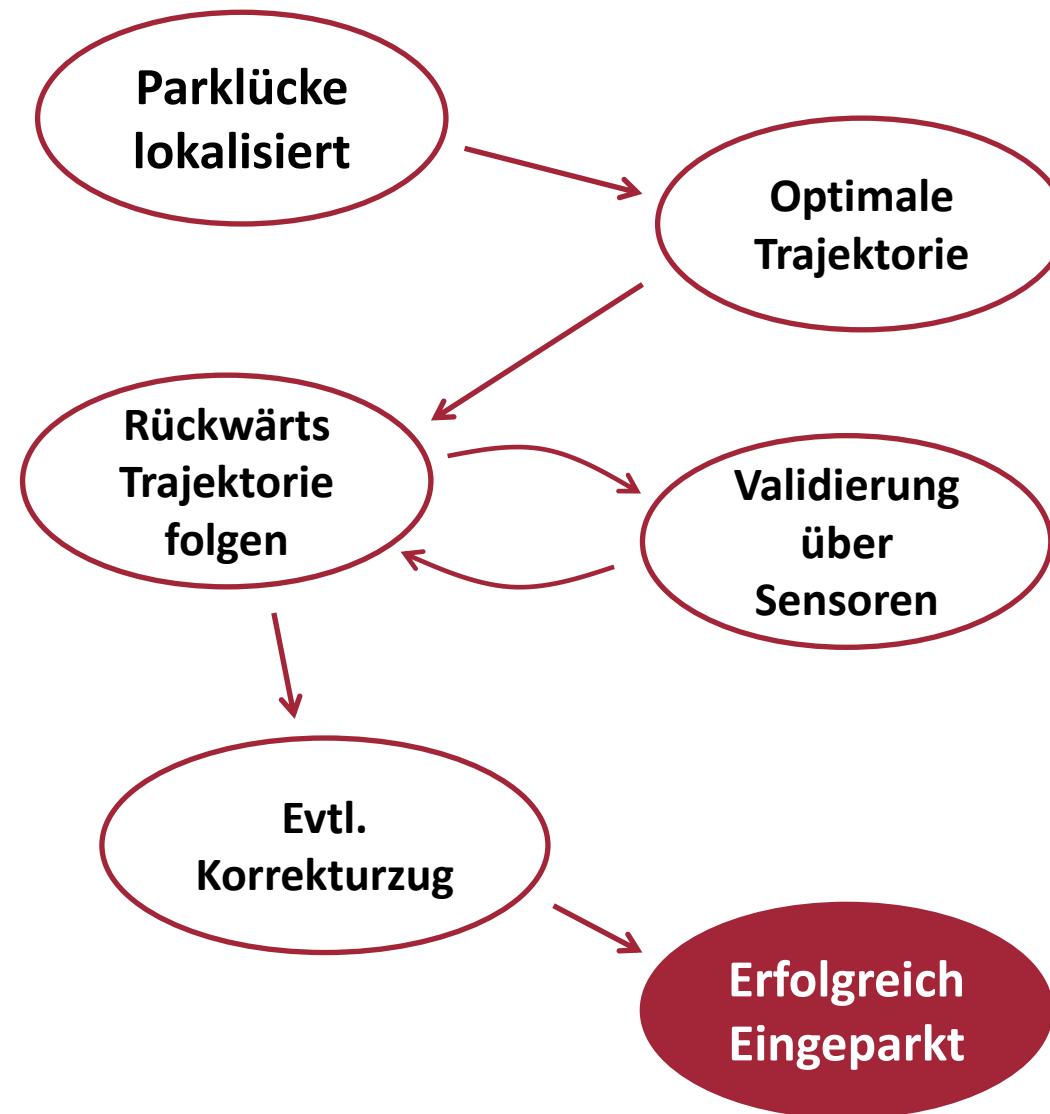
Der Einparkzug



Der Einparkzug



Der Einparkzug



Zentrale Frage:

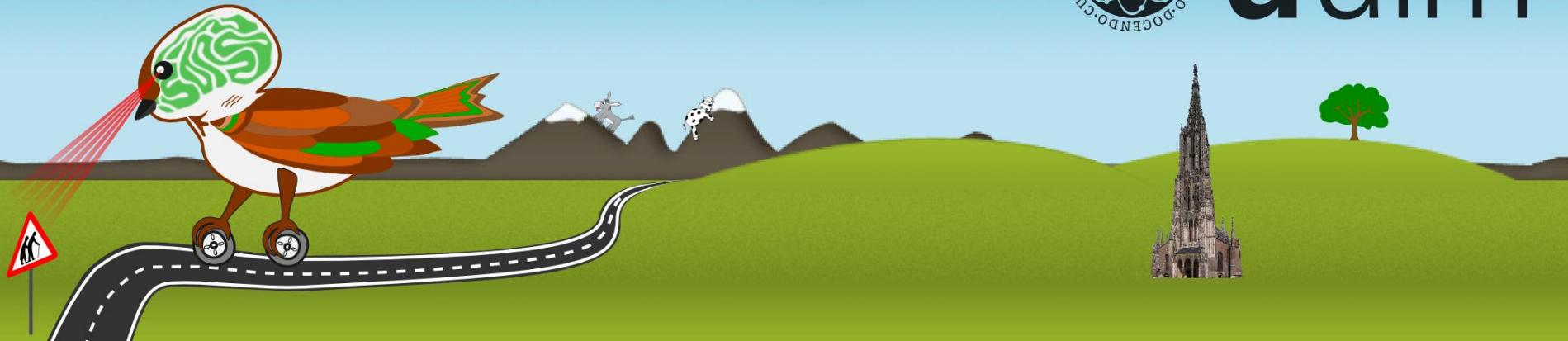
Kann euer Spatz fliegen?

**Nein, aber fahren dafür
umso besser . . .**

Team Spatzenhirn



universität
uulm



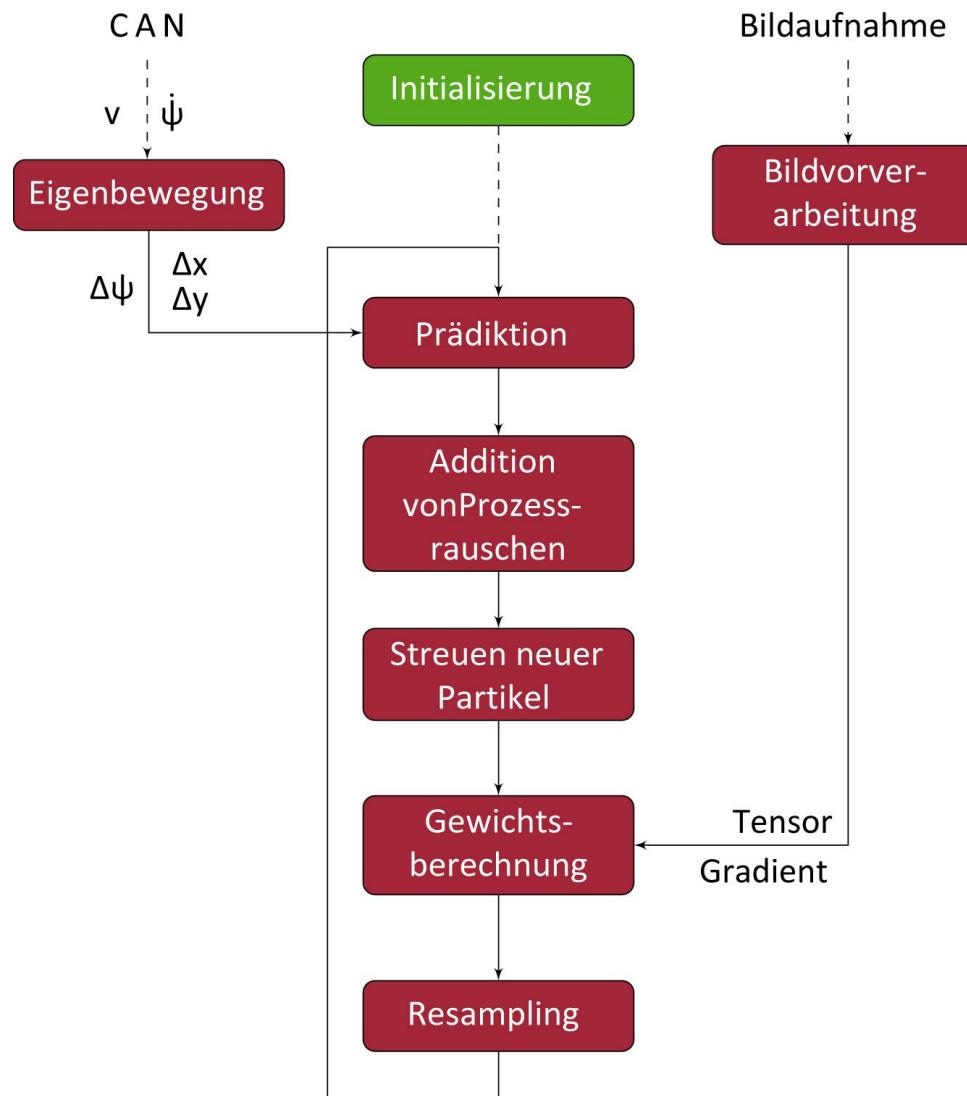
Carolo-Cup Fahrzeugpräsentation

Patrick Hatzelmann, Marcel Debout

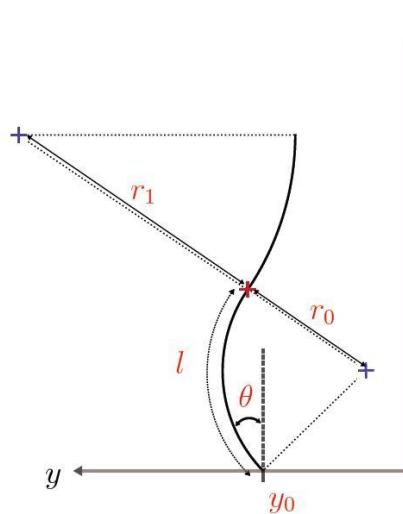
Braunschweig, 5. Februar 2013

Vielen Dank für die Aufmerksamkeit

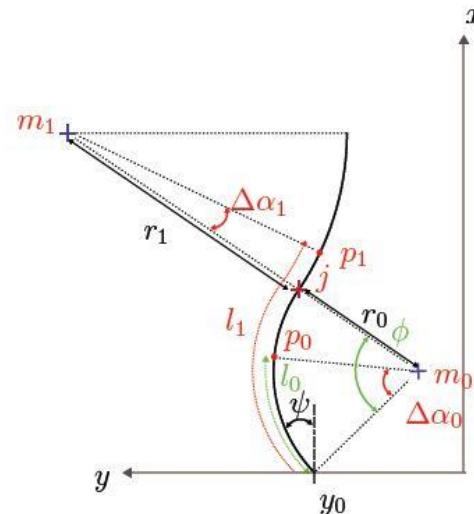
Partikelfilter Ablaufplan



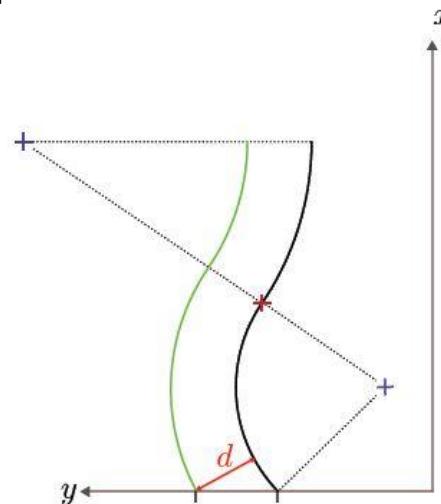
Straßenrepräsentation



Straßenrepräsentation



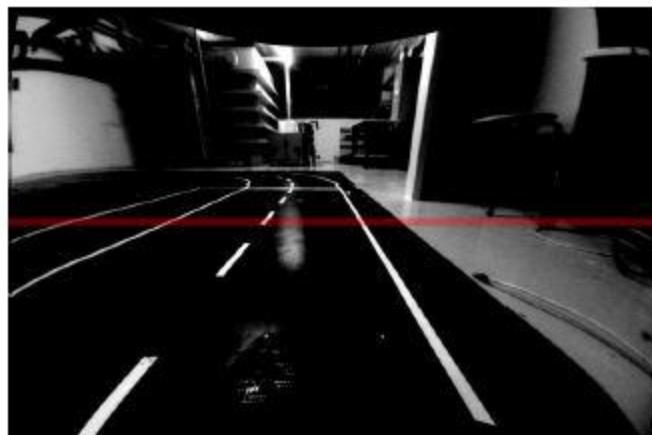
Abtastung



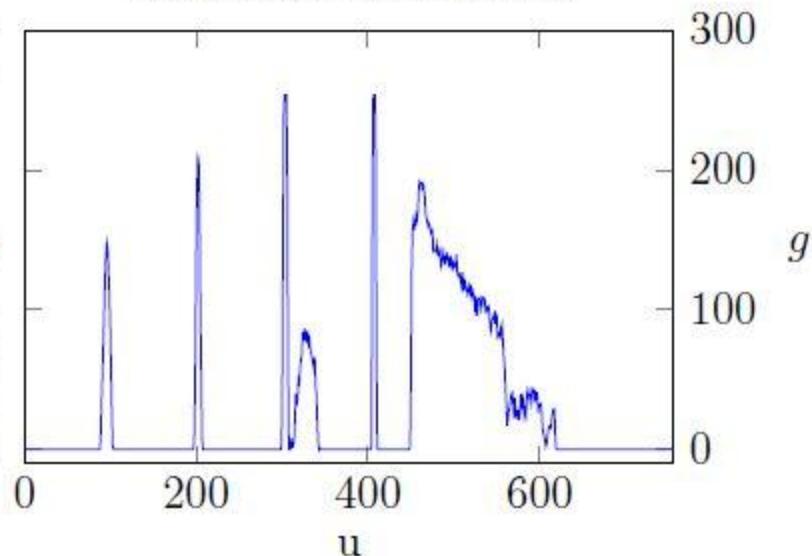
Parallele Fahrbahn

Gradientenfeature

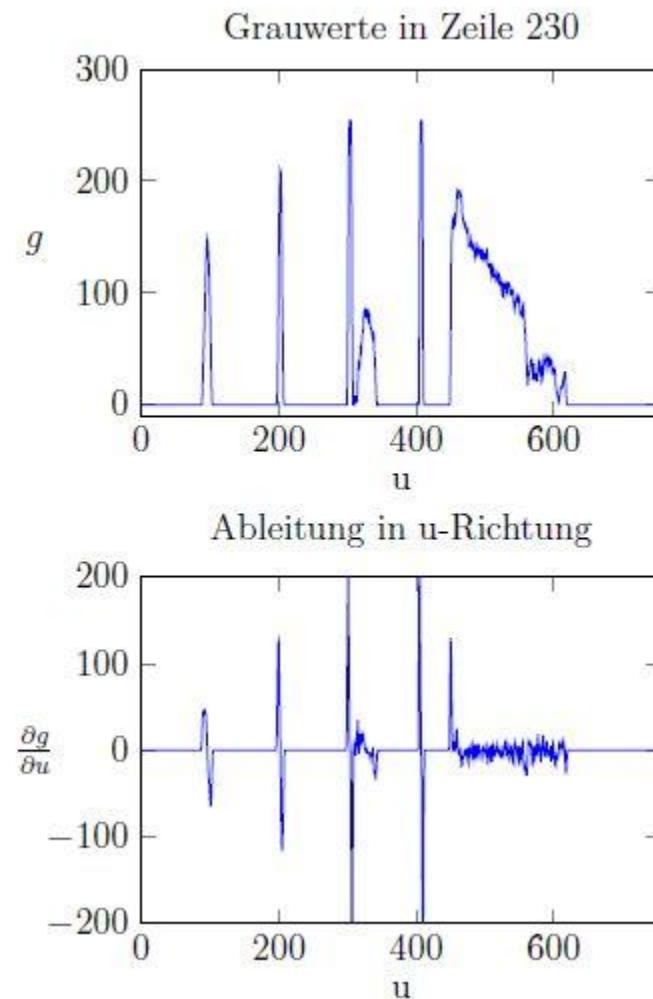
Entzerrtes Kamerabild



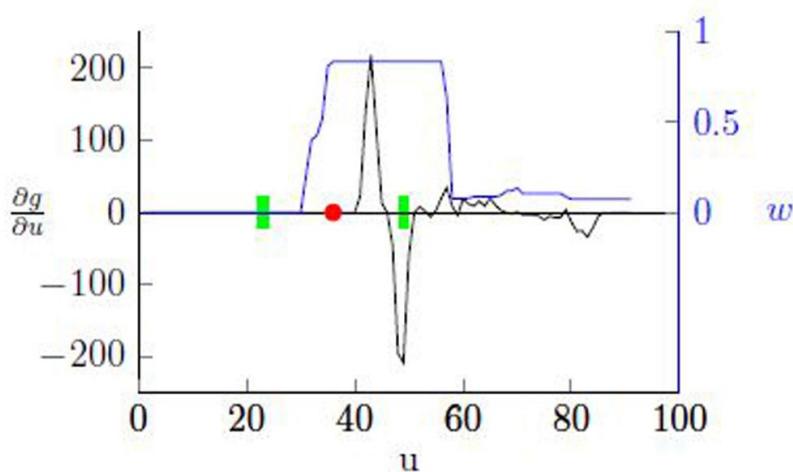
Grauwerte in Zeile 230



Gradientenfeature



Gradientenfeature

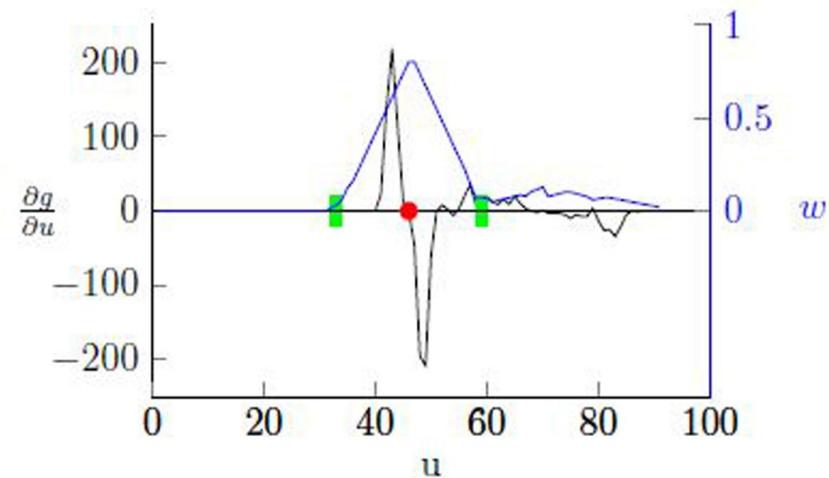


schwarz: horizontale Ableitung

blau: Verlauf der Gewichtsfunktion

rot: Beispielhafte Position eines Partikels

grün: Suchfenster um eingezeichnete Partikelposition



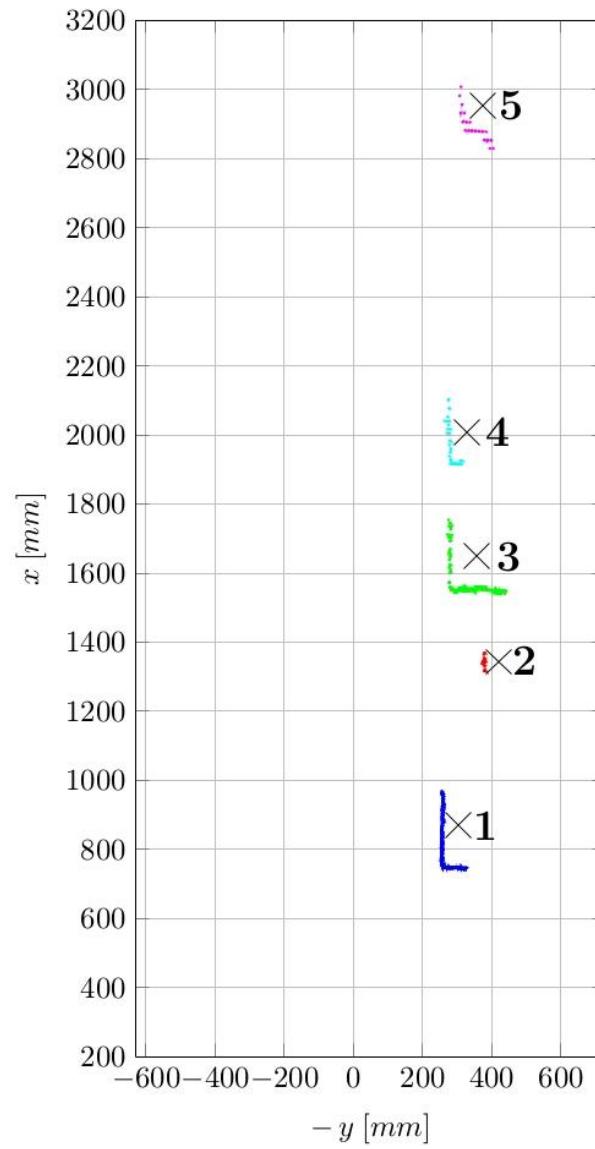
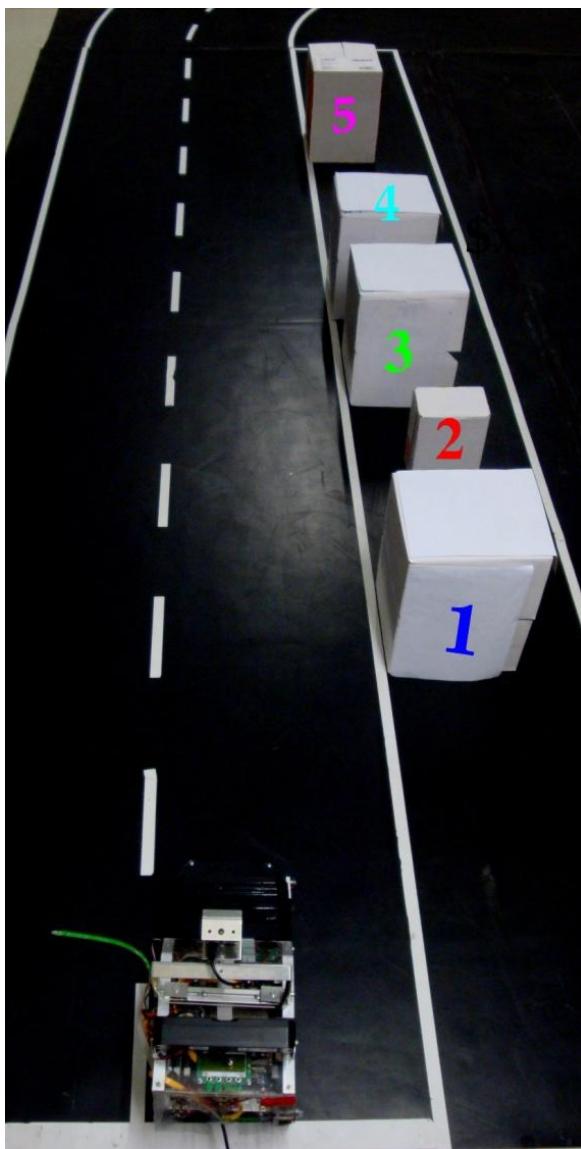
schwarz: horizontale Ableitung

blau: Verlauf der Gewichtsfunktion

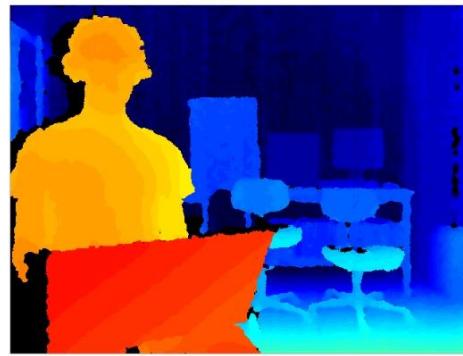
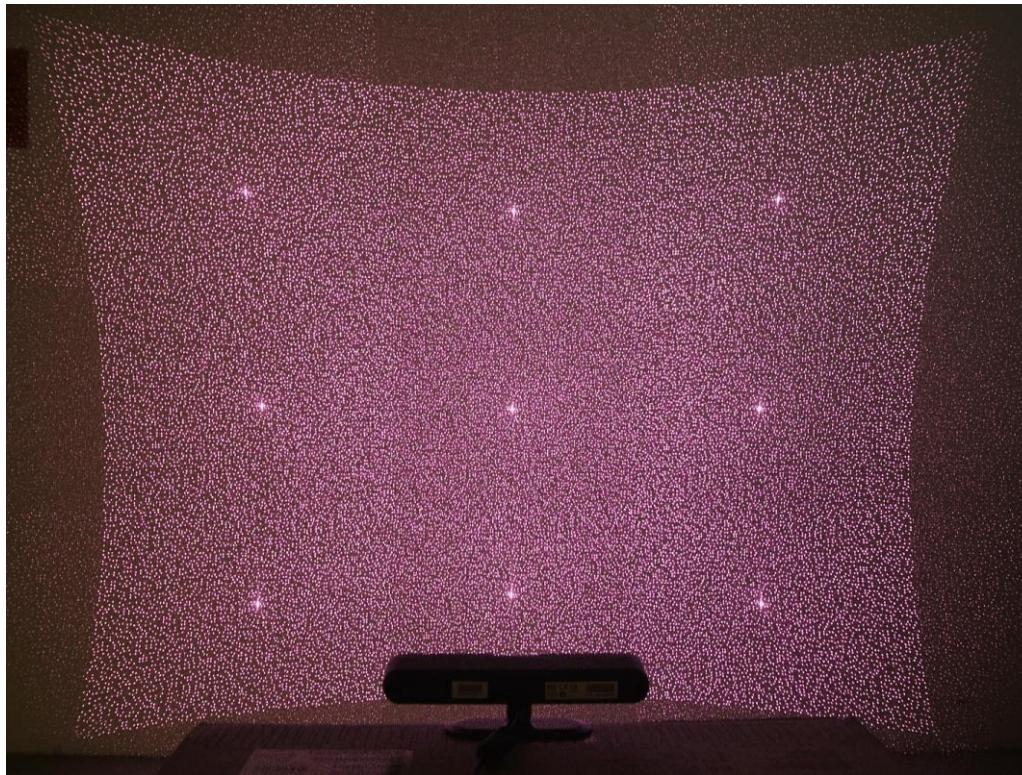
rot: Position des höchsten Gewichts

grün: Suchfenster

Hindernisdetektion - Ergebnisse

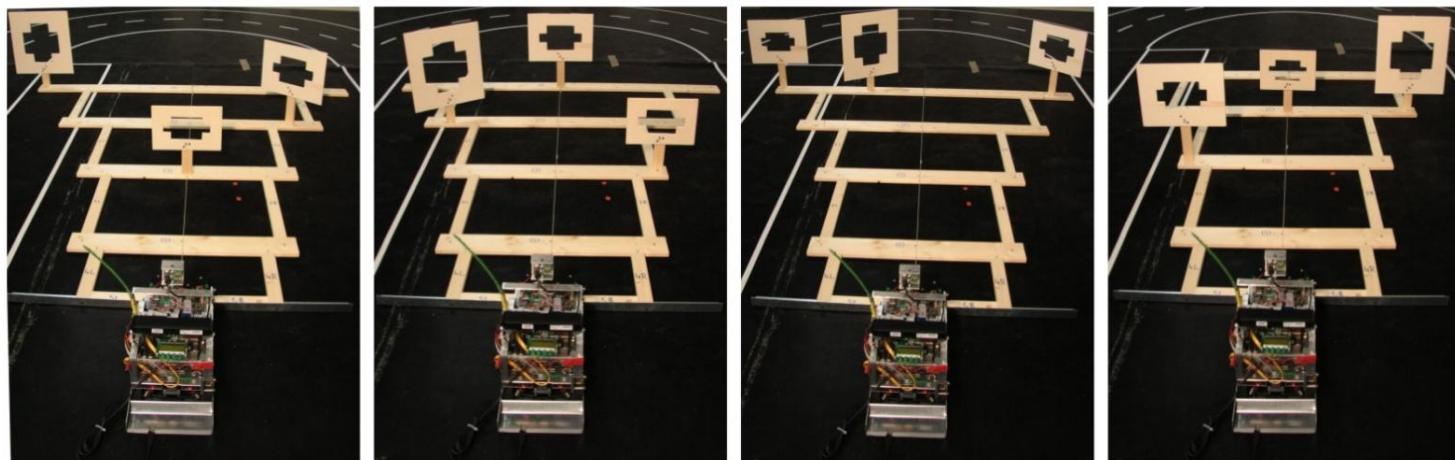
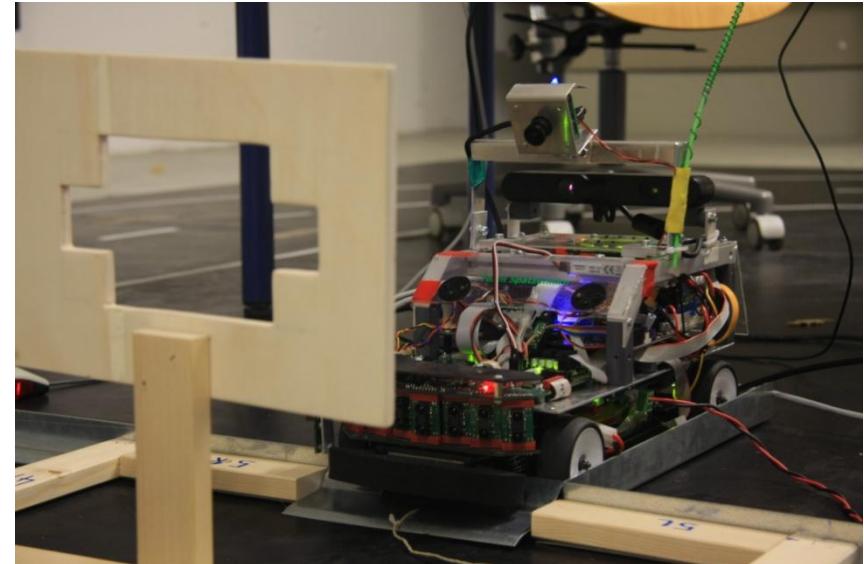
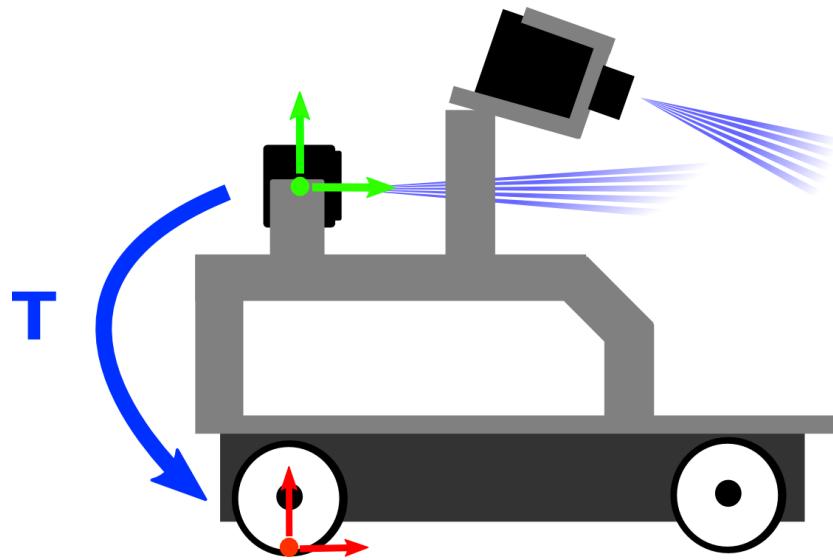


Asus Xtion PRO



Detektion - Kalibrierung

Detektion basiert auf exakter Kalibrierung der Xtion Tiefenbildkamera



Kalibrierung – Extraktion der markanten Eckpunkte

