

Fast Object Detection in Dense Point Clouds

Zielsetzung

- ▶ Robuste Detektion von Schildern, Hindernisse, Fußgänger, Steigungen für Carolo-Cup

Zielsetzung

- ▶ Robuste Detektion von Schildern, Hindernisse, Fußgänger, Steigungen für Carolo-Cup
- ▶ Auf Punkt Wolken aus Intel[©] RealSense[™] Depth Camera D435

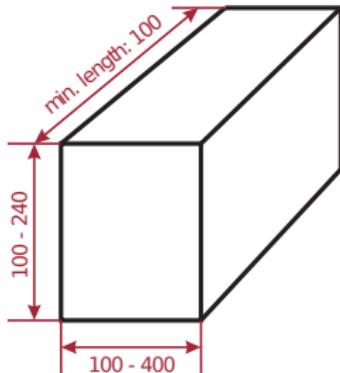
Zielsetzung

- ▶ Robuste Detektion von Schildern, Hindernisse, Fußgänger, Steigungen für Carolo-Cup
- ▶ Auf Punkt Wolken aus Intel[©] RealSense[™] Depth Camera D435
- ▶ Echtzeitfähig auf dem Fahrzeug

Zielsetzung

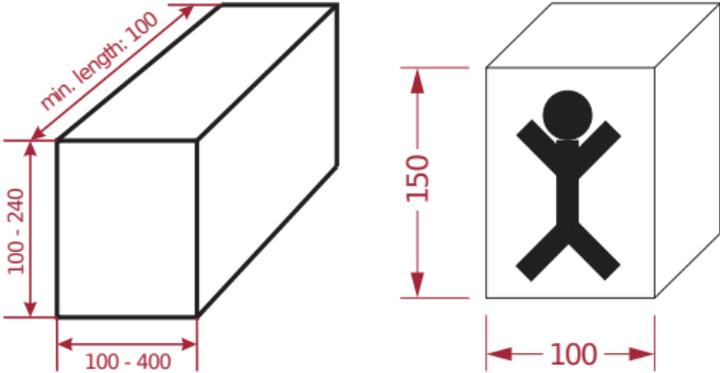
- ▶ Robuste Detektion von Schildern, Hindernisse, Fußgänger, Steigungen für Carolo-Cup
- ▶ Auf Punkt Wolken aus Intel[©] RealSense[™] Depth Camera D435
- ▶ Echtzeitfähig auf dem Fahrzeug
- ▶ Zusätzlich auf Echtweltdaten testen

Bedingungen durch das Carolo-Cup Regelwerk ¹



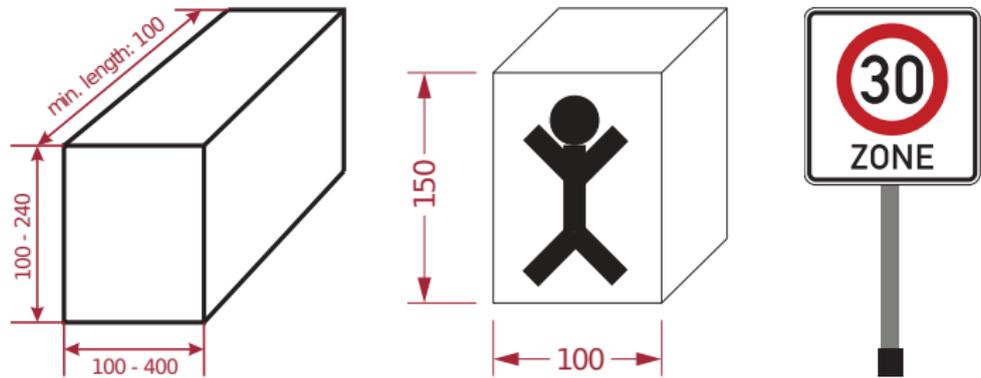
¹Grafiken basierend auf: (Braunschweig, Technische Universität:
Carolo-Cup Regelwerk. 2018)

Bedingungen durch das Carolo-Cup Regelwerk ¹



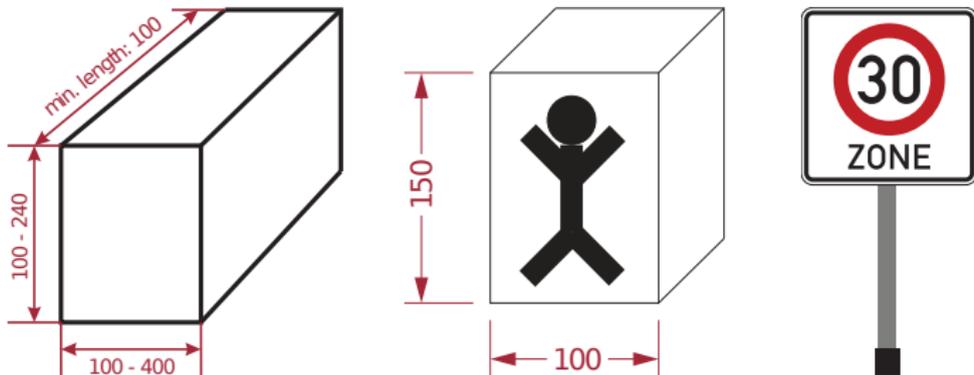
¹Grafiken basierend auf: (Braunschweig, Technische Universität: *Carolo-Cup Regelwerk*. 2018)

Bedingungen durch das Carolo-Cup Regelwerk ¹



¹Grafiken basierend auf: (Braunschweig, Technische Universität: *Carolo-Cup Regelwerk*. 2018)

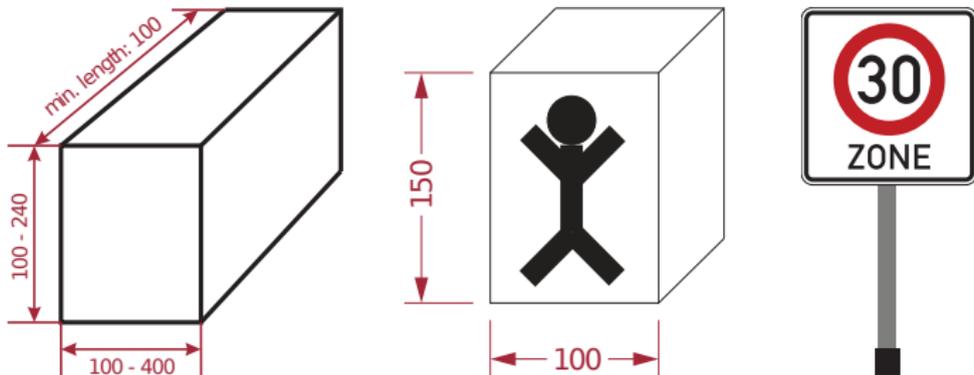
Bedingungen durch das Carolo-Cup Regelwerk ¹



- ▶ Hindernisse und Fußgänger sind nicht anhand der Punktwolke unterscheidbar

¹Grafiken basierend auf: (Braunschweig, Technische Universität: *Carolo-Cup Regelwerk*. 2018)

Bedingungen durch das Carolo-Cup Regelwerk ¹



- ▶ Hindernisse und Fußgänger sind nicht anhand der Punktwolke unterscheidbar
- ▶ Steigungen mit bis zu 10°

¹Grafiken basierend auf: (Braunschweig, Technische Universität: *Carolo-Cup Regelwerk*. 2018)

Stand der Technik

- ▶ Kombinierte Detektion und Klassifikation mit Convolutional Neural Network (CNN)

² Wang, Yan; Chao, Wei-Lun; Garg, Divyansh; Hariharan, Bharath; Campbell, Mark und Weinberger, Kilian Q.: "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". 2018.

Stand der Technik

- ▶ Kombinierte Detektion und Klassifikation mit Convolutional Neural Network (CNN)
- ▶ Langsam ohne GPU (Abschätzung: 30s für Complex-YOLO)

² Wang, Yan; Chao, Wei-Lun; Garg, Divyansh; Hariharan, Bharath; Campbell, Mark und Weinberger, Kilian Q.: "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". 2018.

Stand der Technik

- ▶ Kombinierte Detektion und Klassifikation mit Convolutional Neural Network (CNN)
- ▶ Langsam ohne GPU (Abschätzung: 30s für Complex-YOLO)
- ▶ Schneller: Separate Detektion und Klassifikation

² Wang, Yan; Chao, Wei-Lun; Garg, Divyansh; Hariharan, Bharath; Campbell, Mark und Weinberger, Kilian Q.: "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". 2018.

Stand der Technik

- ▶ Kombinierte Detektion und Klassifikation mit Convolutional Neural Network (CNN)
- ▶ Langsam ohne GPU (Abschätzung: 30s für Complex-YOLO)
- ▶ Schneller: Separate Detektion und Klassifikation
- ▶ Primär für Lidar Daten

² Wang, Yan; Chao, Wei-Lun; Garg, Divyansh; Hariharan, Bharath; Campbell, Mark und Weinberger, Kilian Q.: "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". 2018.

Stand der Technik

- ▶ Kombinierte Detektion und Klassifikation mit Convolutional Neural Network (CNN)
- ▶ Langsam ohne GPU (Abschätzung: 30s für Complex-YOLO)
- ▶ Schneller: Separate Detektion und Klassifikation
- ▶ Primär für Lidar Daten
- ▶ Für Stereo: Oftmals auf Disparitätsbildern

² Wang, Yan; Chao, Wei-Lun; Garg, Divyansh; Hariharan, Bharath; Campbell, Mark und Weinberger, Kilian Q.: "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". 2018.

Stand der Technik

- ▶ Kombinierte Detektion und Klassifikation mit Convolutional Neural Network (CNN)
- ▶ Langsam ohne GPU (Abschätzung: 30s für Complex-YOLO)
- ▶ Schneller: Separate Detektion und Klassifikation
- ▶ Primär für Lidar Daten
- ▶ Für Stereo: Oftmals auf Disparitätsbildern
- ▶ Nachbarschaft wird in Punktwolken besser repräsentiert²

² Wang, Yan; Chao, Wei-Lun; Garg, Divyansh; Hariharan, Bharath; Campbell, Mark und Weinberger, Kilian Q.: "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". 2018.

Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten

³ Börcs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten
- ▶ Vorgehen:

³ Börcs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten
- ▶ Vorgehen:
 - ▶ Segmentierung

³ BörCs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten
- ▶ Vorgehen:
 - ▶ Segmentierung
 - ▶ Clustering

³ Börzs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten
- ▶ Vorgehen:
 - ▶ Segmentierung
 - ▶ Clustering
 - ▶ Extraktion von Pseudotiefenbilder

³ BörCs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten
- ▶ Vorgehen:
 - ▶ Segmentierung
 - ▶ Clustering
 - ▶ Extraktion von Pseudotiefenbilder
 - ▶ Klassifikation

³ Börncs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

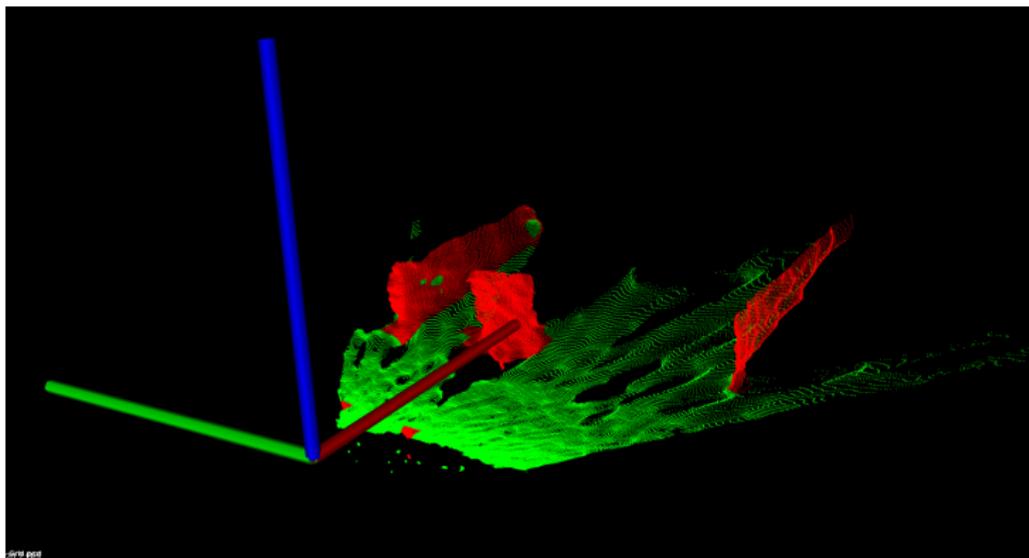
Algorithmus

- ▶ Adaptierte Version von [BNB17]³ für Stereodaten
- ▶ Vorgehen:
 - ▶ Segmentierung
 - ▶ Clustering
 - ▶ Extraktion von Pseudotiefenbilder
 - ▶ Klassifikation
- ▶ Zusätzlich: Bounding-Box- und Bodenschätzung

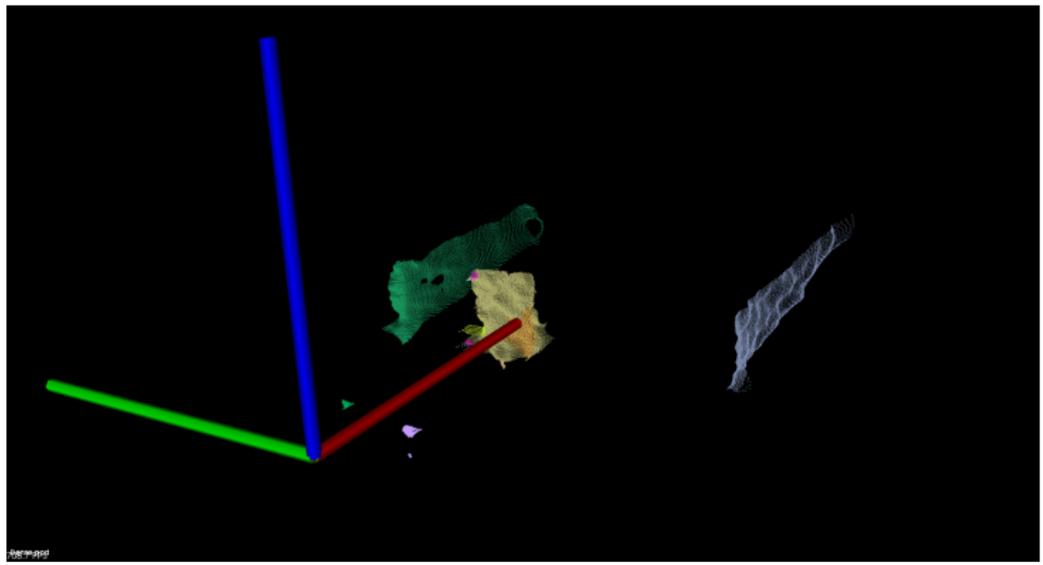
³ BörCs, A.; Nagy, B. und Benedek, C.: "Instant Object Detection in Lidar Point Clouds". 2017.

Algorithmus - Segmentierung

Klassen: Sparse, Ground, High-Foreground,
Low-Foreground

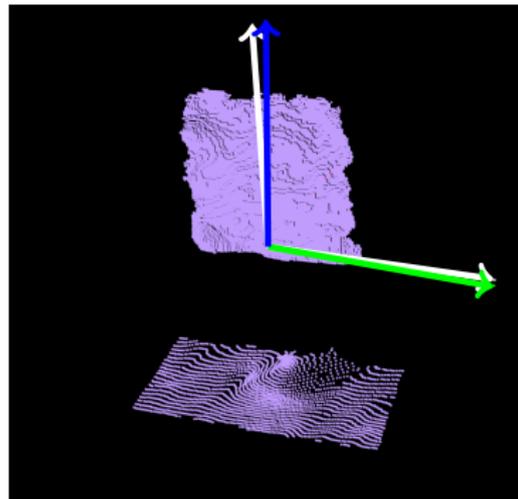


Algorithmus - Clustering



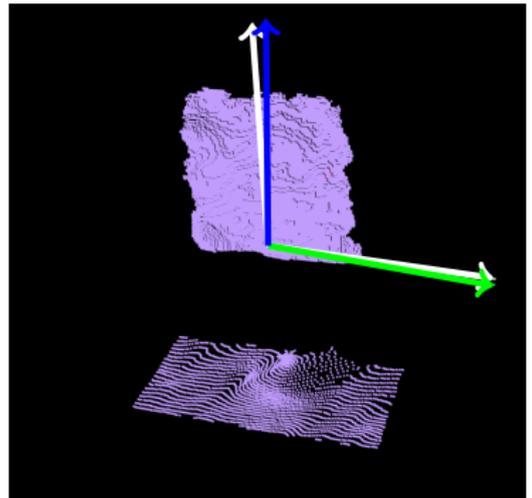
Ausrichtung bestimmen

- ▶ Hauptkomponentenanalyse des Clusters



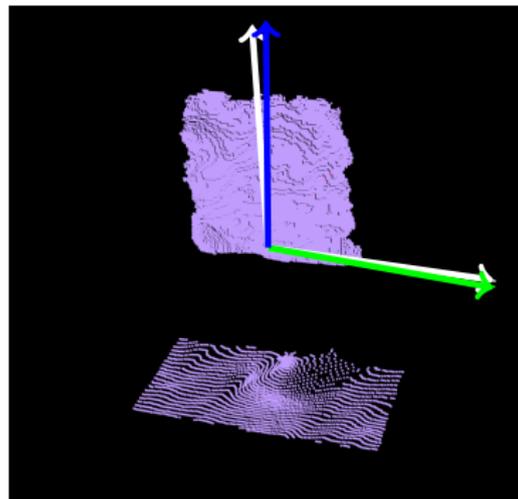
Ausrichtung bestimmen

- ▶ Hauptkomponentenanalyse des Clusters
- ▶ Wichtigste nicht vertikale Hauptachse



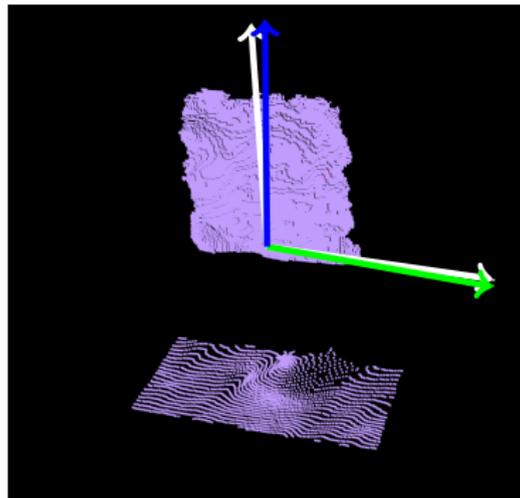
Ausrichtung bestimmen

- ▶ Hauptkomponentenanalyse des Clusters
- ▶ Wichtigste nicht vertikale Hauptachse
- ▶ Für erste HA: Winkel zur z-Achse bestimmen



Ausrichtung bestimmen

- ▶ Hauptkomponentenanalyse des Clusters
- ▶ Wichtigste nicht vertikale Hauptachse
- ▶ Für erste HA: Winkel zur z-Achse bestimmen
- ▶ Wenn größer als 45° dann erste Hauptachse



Algorithmus - Extraktion und Klassifikation

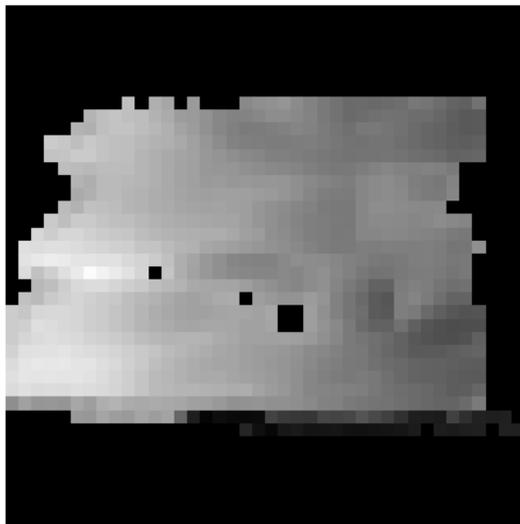


Abbildung: Hindernis

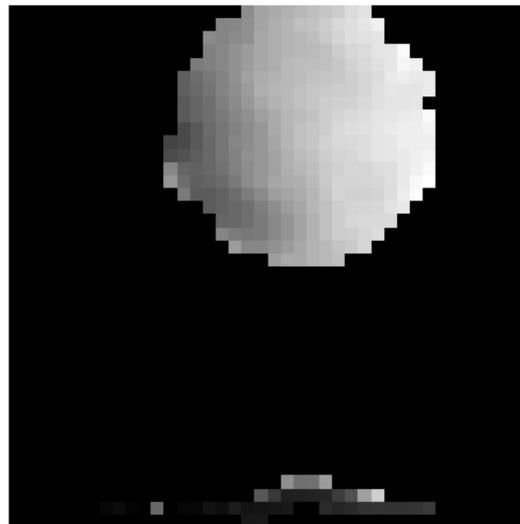


Abbildung: Schild

Algorithmus - Extraktion und Klassifikation

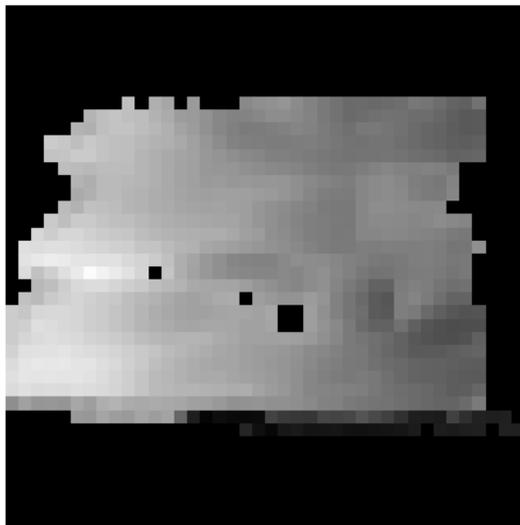


Abbildung: Hindernis

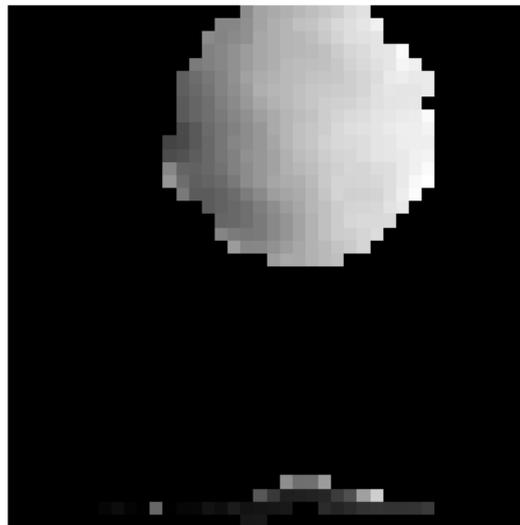


Abbildung: Schild

Klassifikation mit CNN: Vehicle, Short Facade, Street Clutter, Pedestrian

Verbesserungen - Segmentierung

- ▶ Low- und High-Foreground kombiniert

Verbesserungen - Segmentierung

- ▶ Low- und High-Foreground kombiniert
- ▶ Klassifikation über minimum/maximum anfällig

Verbesserungen - Segmentierung

- ▶ Low- und High-Foreground kombiniert
- ▶ Klassifikation über minimum/maximum anfällig
- ▶ Klassifikation über Varianz

Verbesserungen - Extraktion und Klassifikation

- ▶ Hauptkomponentenanalyse nicht notwendig

Verbesserungen - Extraktion und Klassifikation

- ▶ Hauptkomponentenanalyse nicht notwendig
- ▶ Kleinere Pseudotiefenbilder

Verbesserungen - Extraktion und Klassifikation

- ▶ Hauptkomponentenanalyse nicht notwendig
- ▶ Kleinere Pseudotiefenbilder
- ▶ Median-Filter für Distanzinvarianz

Verbesserungen - Extraktion und Klassifikation

- ▶ Hauptkomponentenanalyse nicht notwendig
- ▶ Kleinere Pseudotiefenbilder
- ▶ Median-Filter für Distanzinvarianz
- ▶ Nur drei Klassen: Obstacle, Clutter, Pedestrian

Verbesserungen - Extraktion und Klassifikation

- ▶ Hauptkomponentenanalyse nicht notwendig
- ▶ Kleinere Pseudotiefenbilder
- ▶ Median-Filter für Distanzinvarianz
- ▶ Nur drei Klassen: Obstacle, Clutter, Pedestrian
- ▶ Kleineres CNN ausreichend

Verbesserungen - Extraktion und Klassifikation

- ▶ Hauptkomponentenanalyse nicht notwendig
- ▶ Kleinere Pseudotiefenbilder
- ▶ Median-Filter für Distanzinvarianz
- ▶ Nur drei Klassen: Obstacle, Clutter, Pedestrian
- ▶ Kleineres CNN ausreichend
- ▶ Trainingsdatensatz: 2406 Bilder

Verbesserungen - Textur

- ▶ Für Kitty/Lehr: Farbinformation für jeden Punkt

Verbesserungen - Textur

- ▶ Für Kitt/Lehr: Farbinformation für jeden Punkt
- ▶ Wird statt Tiefeninformation genutzt

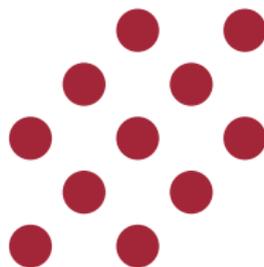
Verbesserungen - Textur

- ▶ Für Kitti/Lehr: Farbinformation für jeden Punkt
- ▶ Wird statt Tiefeninformation genutzt



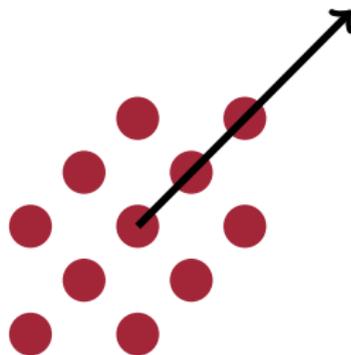
Erweiterungen - Bounding Box

- ▶ Zum Umfahren von Hindernissen/Fußgängern, bzw. Klassifikation von Schilder



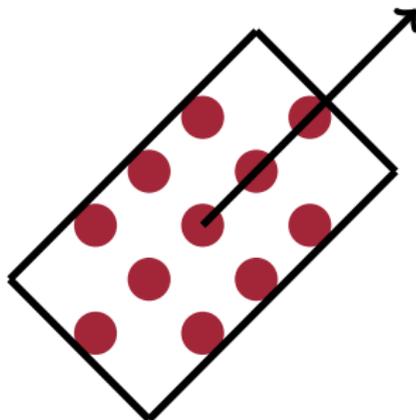
Erweiterungen - Bounding Box

- ▶ Zum Umfahren von Hindernissen/Fußgängern, bzw. Klassifikation von Schilder
- ▶ Zuerst Ausrichtung bestimmen



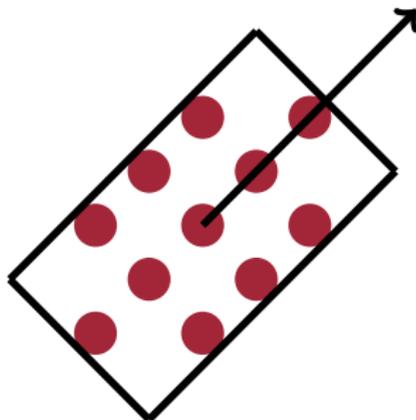
Erweiterungen - Bounding Box

- ▶ Zum Umfahren von Hindernissen/Fußgängern, bzw. Klassifikation von Schilder
- ▶ Zuerst Ausrichtung bestimmen
- ▶ Dann Ausdehnung in x und y Richtung bestimmen



Erweiterungen - Bounding Box

- ▶ Zum Umfahren von Hindernissen/Fußgängern, bzw. Klassifikation von Schilder
- ▶ Zuerst Ausrichtung bestimmen
- ▶ Dann Ausdehnung in x und y Richtung bestimmen
- ▶ Ausdehnung in z Richtung bestimmen



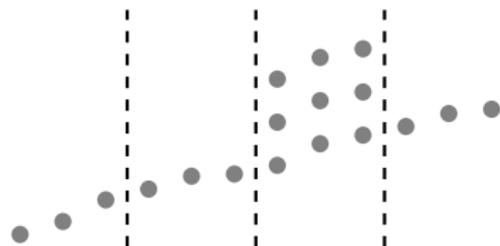
Erweiterungen - Bodenschätzung

- ▶ Alle Ground-Zellen werden Berücksichtigt



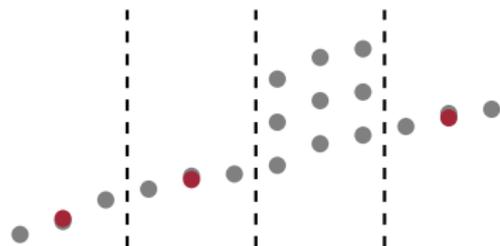
Erweiterungen - Bodenschätzung

- ▶ Alle Ground-Zellen werden Berücksichtigt
- ▶ Mittel der z-Werte pro Zelle



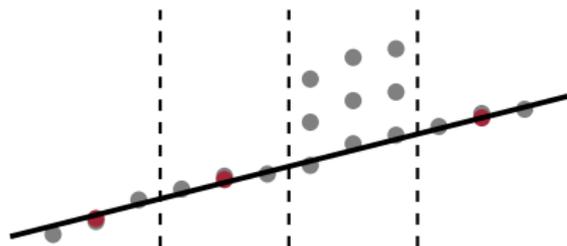
Erweiterungen - Bodenschätzung

- ▶ Alle Ground-Zellen werden Berücksichtigt
- ▶ Mittel der z-Werte pro Zelle
- ▶ Pro Zelle ein Punkt



Erweiterungen - Bodenschätzung

- ▶ Alle Ground-Zellen werden Berücksichtigt
- ▶ Mittel der z-Werte pro Zelle
- ▶ Pro Zelle ein Punkt
- ▶ Ebene mit Methode der kleinsten Quadrate



Evaluation - Vorgehen

- ▶ 20 von Hand annotierte Punktwolken aus D435

Evaluation - Vorgehen

- ▶ 20 von Hand annotierte Punktwolken aus D435
- ▶ Insgesamt 47 Objekte (22 Hindernisse/Fußgänger, 25 Schilder)

Evaluation - Vorgehen

- ▶ 20 von Hand annotierte Punktwolken aus D435
- ▶ Insgesamt 47 Objekte (22 Hindernisse/Fußgänger, 25 Schilder)
- ▶ Intersection over Union (IoU) von Bounding Boxes

Evaluation - Verbesserungen

Algorithmus	Detektionsrate	Pr	Rc	F_1
[BNB17] Lidar				
[BNB17] Stereo				
Vorgeschlagen				

Evaluation - Verbesserungen

Algorithmus	Detektionsrate	Pr	Rc	F_1
[BNB17] Lidar				
[BNB17] Stereo	79%			
Vorgeschlagen	81%			

Evaluation - Verbesserungen

Algorithmus	Detektionsrate	Pr	Rc	F_1
[BNB17] Lidar		86%		
[BNB17] Stereo	79%	86%		
Vorgeschlagen	81%	89%		

Evaluation - Verbesserungen

Algorithmus	Detektionsrate	Pr	Rc	F_1
[BNB17] Lidar		86%	83%	
[BNB17] Stereo	79%	86%	90%	
Vorgeschlagen	81%	89%	92%	

Evaluation - Verbesserungen

Algorithmus	Detektionsrate	Pr	Rc	F_1
[BNB17] Lidar		86%	83%	85%
[BNB17] Stereo	79%	86%	90%	86%
Vorgeschlagen	81%	89%	92%	90%

Vergleich aktuelle Hindernisserkennung

Vergleich nur auf Basis von Hindernissen

Algorithmus	Durchschnittliche 2D-IoU Detektionen Alle Objekte
Hindernisserkennung Vorgeschlagen	

Vergleich aktuelle Hindernisserkennung

Vergleich nur auf Basis von Hindernissen

Algorithmus	Durchschnittliche 2D-IoU Detektionen Alle Objekte
Hindernisserkennung	0.48
Vorgeschlagen	0.44

Vergleich aktuelle Hindernisserkennung

Vergleich nur auf Basis von Hindernissen

Algorithmus	Durchschnittliche 2D-IoU	
	Detektionen	Alle Objekte
Hindernisserkennung	0.48	0.087
Vorgeschlagen	0.44	0.38

Analyse der Rechenzeit

Daten von Referenzsystem vergleichbar mit Fahrzeug

	Mittelwert	Standardabweichung
Segmentierung	2.1 ms	0.49 ms
Clustering	1.2 ms	0.52 ms
Extraktion	0.62ms	0.25 ms
Klassifikation	15 ms	10 ms
Bounding Box	0.73ms	0.29 ms
Bodenschätzung	0.12ms	0.046ms
Gesamt	19 ms	11 ms

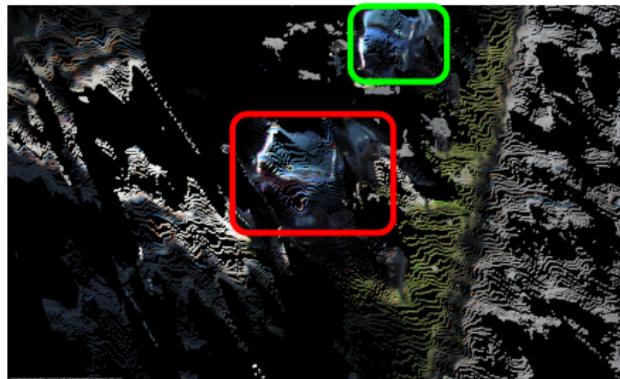
Echtweltdaten - Kitti

- Stereo-
berechnung mit
Semi-Global
Block Matching



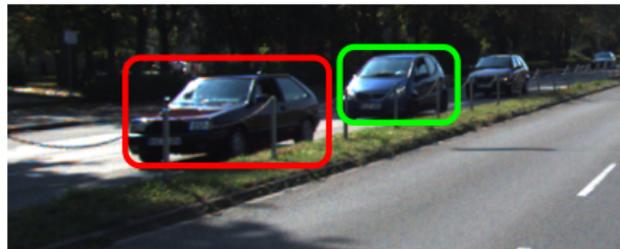
Echtweltdaten - Kitti

- ▶ Stereo-
berechnung mit
Semi-Global
Block Matching
- ▶ Kleiner
Kameraabstand
(0,54m)



Echtweltdaten - Kitti

- ▶ Stereo-
berechnung mit
Semi-Global
Block Matching
- ▶ Kleiner
Kameraabstand
(0,54m)
- ▶ Punktwolke
ungenau



Echtweltdaten - Kitty

- ▶ Stereo-
berechnung mit
Semi-Global
Block Matching
- ▶ Kleiner
Kameraabstand
(0,54m)
- ▶ Punktwolke
ungenau
- ▶ Detektion bis ca.
10m möglich



Echtweltdaten - Lehr

- ▶ Bessere Punktwolke

Echtweltdaten - Lehr

- ▶ Bessere Punktwolke
- ▶ Einfaches Szenario

Echtweltdaten - Lehr

- ▶ Bessere Punktwolke
- ▶ Einfaches Szenario
- ▶ Ermöglicht Detektion bis ca. 50m

Echtweltdaten - Lehr

- ▶ Bessere Punktwolke
- ▶ Einfaches Szenario
- ▶ Ermöglicht Detektion bis ca. 50m
- ▶ Detektionsrate: 88%

Echtweltdaten - Lehr

- ▶ Bessere Punktwolke
- ▶ Einfaches Szenario
- ▶ Ermöglicht Detektion bis ca. 50m
- ▶ Detektionsrate: 88%
- ▶ Durchschnittliche IoU: 0.53 (2D), 0.43 (3D)

Fazit

- ▶ Algorithmus adaptiert und erweitert für Stereo Daten

Fazit

- ▶ Algorithmus adaptiert und erweitert für Stereo Daten
- ▶ Verbesserungen für Stereo Daten

Fazit

- ▶ Algorithmus adaptiert und erweitert für Stereo Daten
- ▶ Verbesserungen für Stereo Daten
- ▶ Verbesserungen gegenüber aktueller Hinderniserkennung

Fazit

- ▶ Algorithmus adaptiert und erweitert für Stereo Daten
- ▶ Verbesserungen für Stereo Daten
- ▶ Verbesserungen gegenüber aktueller Hinderniserkennung
- ▶ Ergebnisse auf Echtweltdaten abhängig von Punktwolke

Mögliche Verbesserungen

- ▶ Besseres Stereo Matching

Mögliche Verbesserungen

- ▶ Besseres Stereo Matching
- ▶ Segmentierung durch MLP

Mögliche Verbesserungen

- ▶ Besseres Stereo Matching
- ▶ Segmentierung durch MLP
- ▶ Klassifikation immer über Farbbild

Abkürzungen

CNN Convolutional Neural Network

D435 Intel© RealSense™ Depth Camera D435

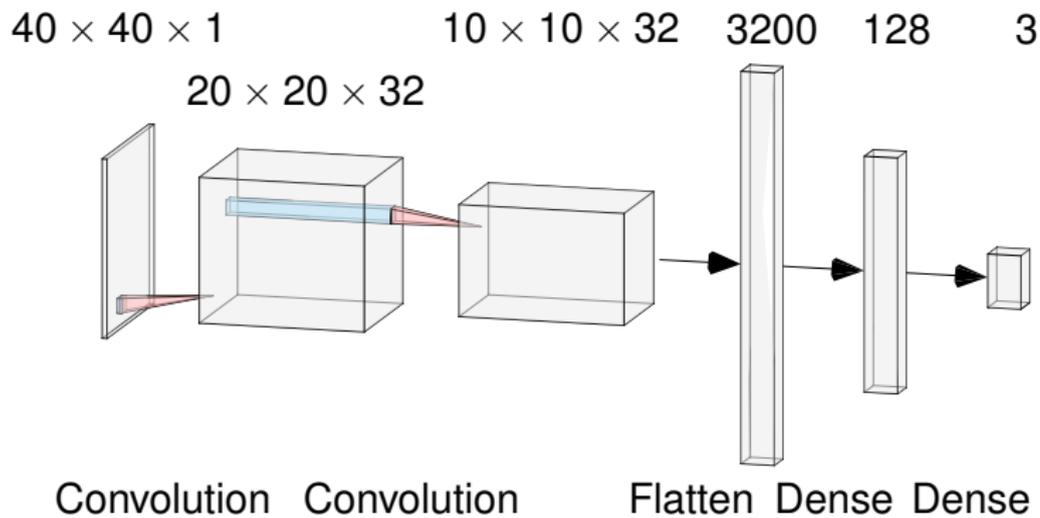
GPU Graphics processing unit

IoU Intersection over Union

Abschätzung Laufzeit

	Titan X	Fahrzeug
YOLO	33ms	46s
Complex-YOLO	20ms	ca. 31s

CNN



Segmentierung - Punktweise

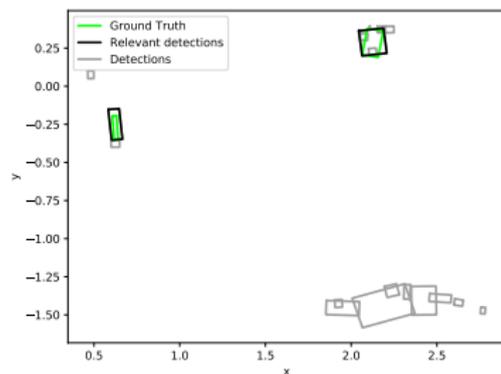
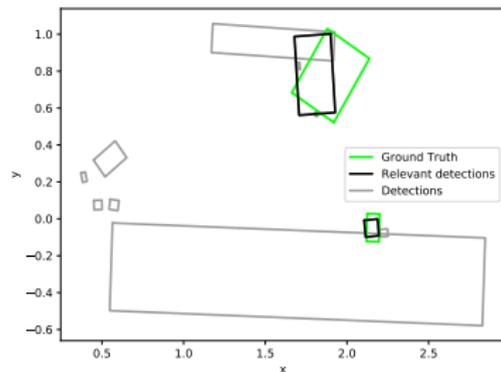
Predicted \ Actual	Ground	Foreground	Sparse
Ground	958,704 (92.1%)	3,436 (2.0%)	12,224 (53.3%)
Foreground	70,293 (6.8%)	164,209 (97.9%)	10,416 (45.4%)
Sparse	11,688 (1.1%)	71 (0.04%)	273 (1.2%)
Gesamt	1,040,685	167,716	22,913

Segmentierung - Zellenweise

Predicted \ Actual	Ground	Foreground	Sparse
Ground	8,302 (96.9%)	45 (14.8%)	227 (0.45%)
Foreground	184 (2.1%)	248 (81%)	45 (0.1%)
Sparse	79 (0.92%)	12 (4.0%)	49,943 (99.5%)
Gesamt	8,565	305	50,215

Clustering und Bounding-Box Schätzung

Durchschnitt 2D	0.444
Durchschnitt 3D	0.321



Klassifikation

Predicted \ Actual	Clutter	Obstacle	Sign
Clutter	555 (89.5%)	3 (15.8%)	3 (16.7%)
Obstacle	49 (7.9%)	15 (78.9%)	1 (5.6%)
Sign	16 (2.5%)	1 (5.2%)	14 (77.8%)
Gesamt	620	19	18

Evaluation - Verbesserungen

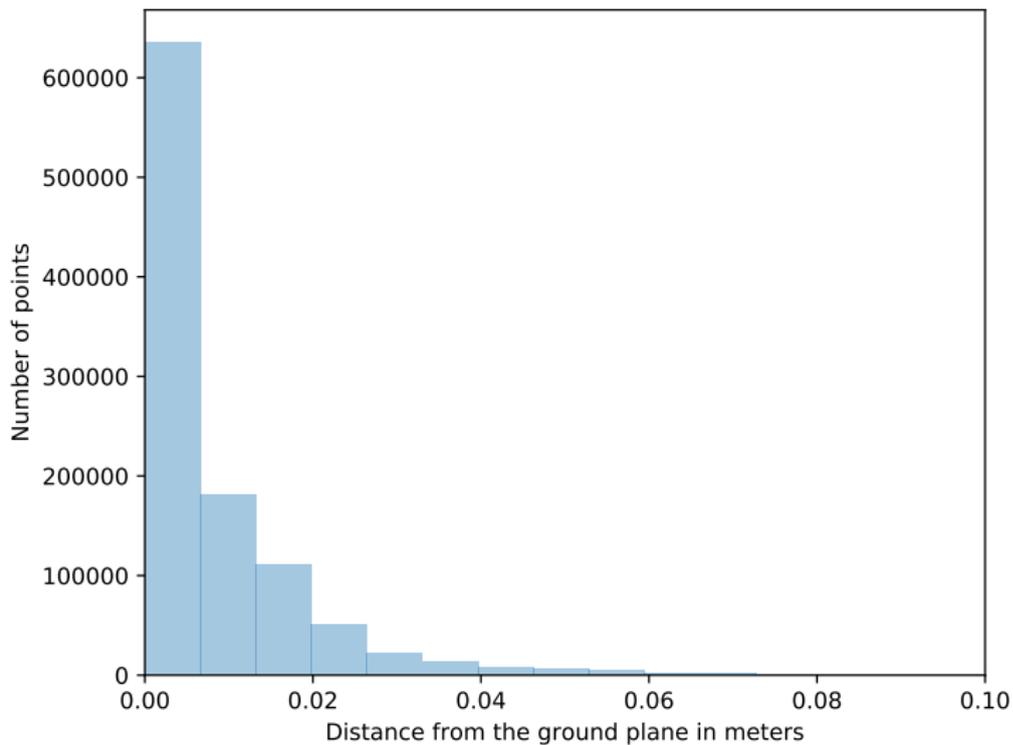
Vorgeschlagen:

Kategorie	Anzahl	Pr	Rc	F_1
Obstacle/Pedestrian	20	100%	84%	91%
Sign	18	78%	100%	88%
Durchschnitt/Summe	38	89%	92%	90%

[BNB17]:

Kategorie	Anzahl	Pr	Rc	F_1
Obstacle/Pedestrian	20	100%	80%	89%
Sign	17	71%	100%	83%
Durchschnitt/Summe	37	86%	90%	86%

Bodenschätzung



Evaluation Lehr

Durchschnitt 2D 0.53

Durchschnitt 3D 0.43

