

Automatisierung der Kartographierung von Straßenschildern

Abschlusspräsentation
Bachelorarbeit
Philipp Unger

- Hintergrund
- Grundidee
- Ziel der Arbeit
- Android Applikation: ‚GPStreetCam‘
- Java Programm: ‚GPStreetTracker‘
 - Framework zur Objekterkennung
 - Plugin zur Straßenschilderkennung
- Schilderkennung anhand eines Beispiels
- Demonstration des Programms
- Fazit

- Projekt wurde 2004 gegründet
- Ziel: Erschaffung einer freien Weltkarte
- Es werden geographische Daten über Flüsse, Wälder, Häuser und auch Straßen gesammelt.
- Daten sind lizenzkostenfrei und dürfen von jedem benutzt und weiterverarbeitet werden.
- Um die Rechte an diesen Daten zu erhalten, müssen die geographischen Daten von den Mappern selbst erhoben werden.
- Erfassen der Daten ist ein großer Aufwand und stellt im Straßenverkehr ein erhebliches Risiko dar.

- Arbeitserleichterung für den OpenStreetMapper
- Automatische Objekterkennung im Straßenverkehr
- Speicherung der GPS-Daten zu den erkannten Objekten

In dieser Arbeit

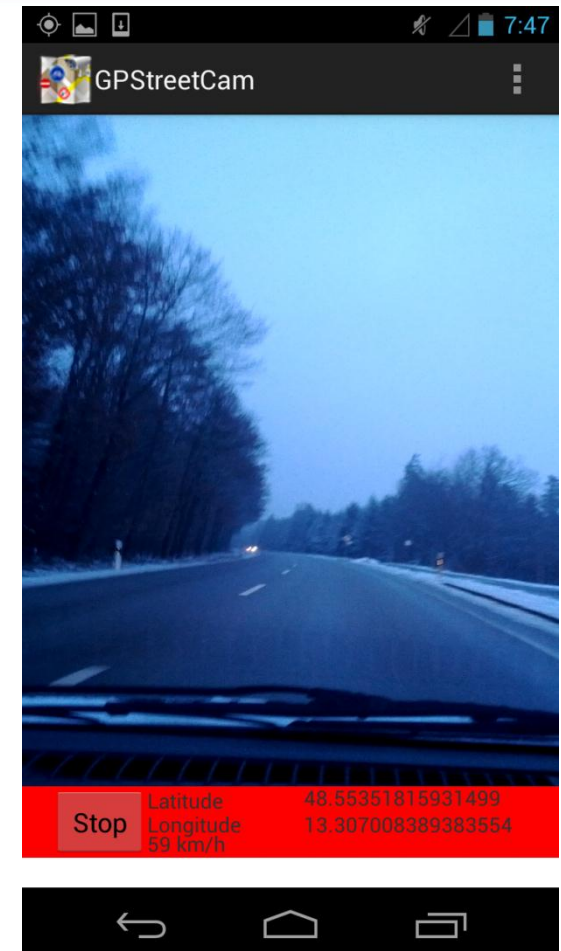
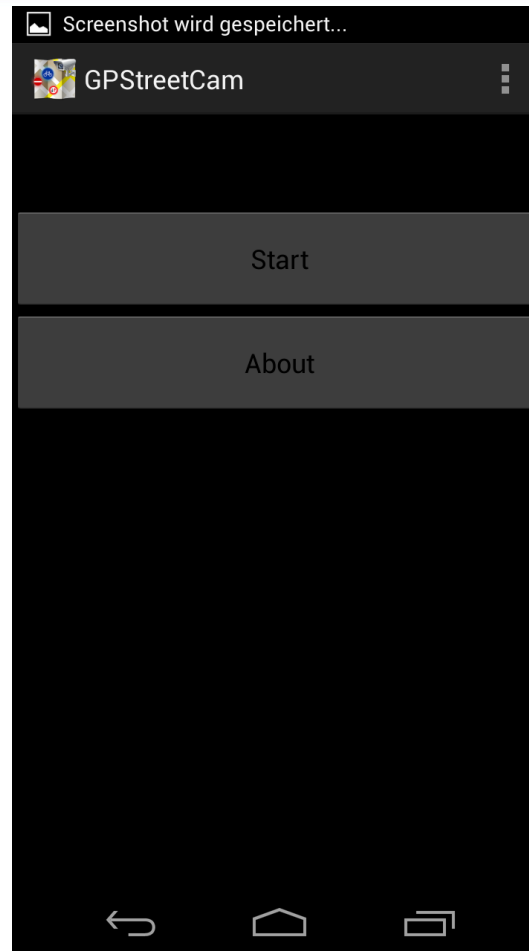
- Aufgabe:
 - Automatische Erkennung von Objekten im Straßenverkehr
 - Kartographierung der erkannten Objekte
- Lösungsansatz:
 - Entwicklung einer Android-Applikation zur Aufzeichnung der gefahrenen Strecke
 - Aufzeichnung von Video- und GPS-Daten
 - Verarbeitung der gesammelten Daten in einem entwickelten Framework
 - Straßenschilderkennung mithilfe eines entwickelten Plugins

Android App: ‚GPStreetCam‘

- Aufzeichnung der gefahrenen Strecke über die interne Smartphone-Kamera
- Aufzeichnung mit bestmöglicher Videoqualität
- Gleichzeitiges Aufzeichnen von GPS-Daten
- Speicherung der GPS-Daten als GPX-Track
 - Gängiges Format zur Speicherung von Geodaten
 - Einfache Weiterverarbeitung
- Zweckmäßigkeit und einfache Bedienung

Android App: „GPStreetCam“

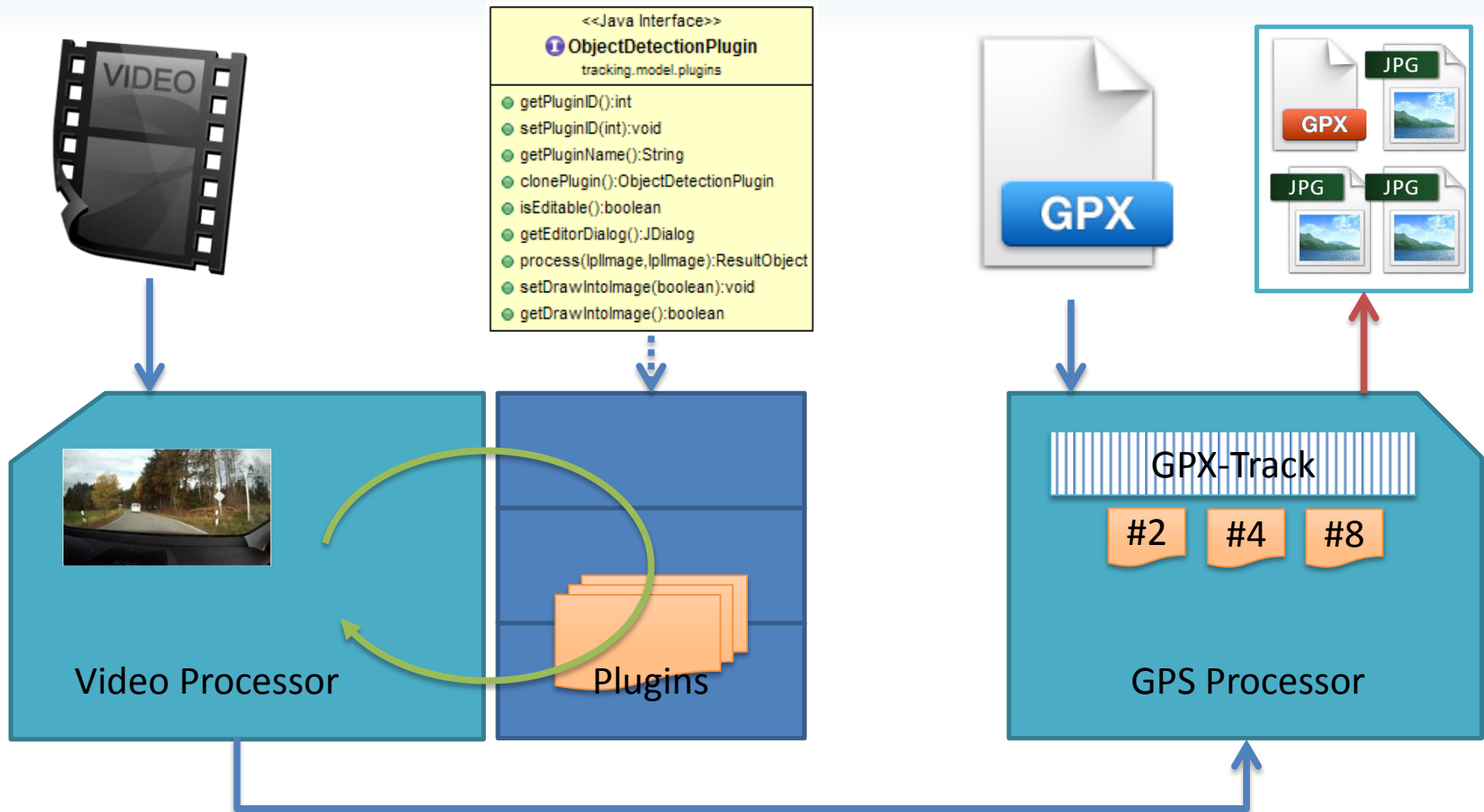
- Einfaches Hauptmenü
- GPS-Verbindung wird beim Aufruf des Recorders gesucht
- Nach erfolgreicher Verbindung werden dem Benutzer GPS-Daten angezeigt
- Bei stehender Verbindung kann die Aufnahme beliebig oft gestartet und gestoppt werden
- Aufnahme wird rot angezeigt



Java Framework: ,GPStreetTracker‘

- Verarbeitung der erfassten Daten
- Einlesen des aufgezeichneten Videos
- Verarbeitung der einzelnen Frames mit Objekterkennungsalgorithmen
 - Erweiterbarkeit durch Plugins
 - Einheitliches Plugin-Interface
- Speicherung und Kartographierung der erkannten Objekte

Java Framework: ,GPStreetTracker‘



Plugin zur Schilderkennung: ,HoughSignRecognition‘

- Einfaches Verfahren zur Erkennung von Straßenschildern
- Finden der *Regions of Interest* über einen Farbfilter und Blob-Erkennung
- Formerkennung mit Hilfe von Hough-Transformation
- Schildverfolgung über einen einfachen Tracking-Algorithmus
- Verwendung von OpenCV-Algorithmen zur Bildverarbeitung

Schilderkennung



- Extrahiere Frame aus Video
- Umwandlung in HSV-Farbraum



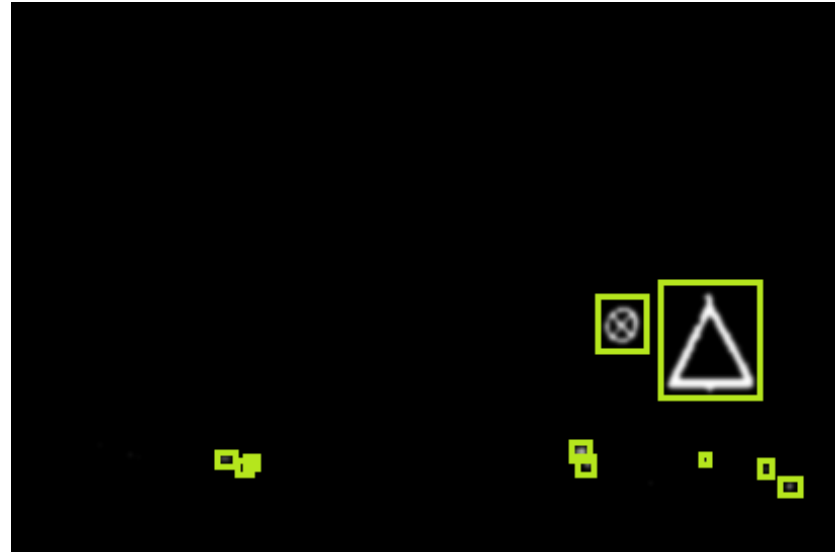
HSV-Raum ermöglicht die Manipulation der Farben über die 3 Kanäle:

Farbe(Hue) – Sättigung(Saturation) – Dunkelstufe(Value)

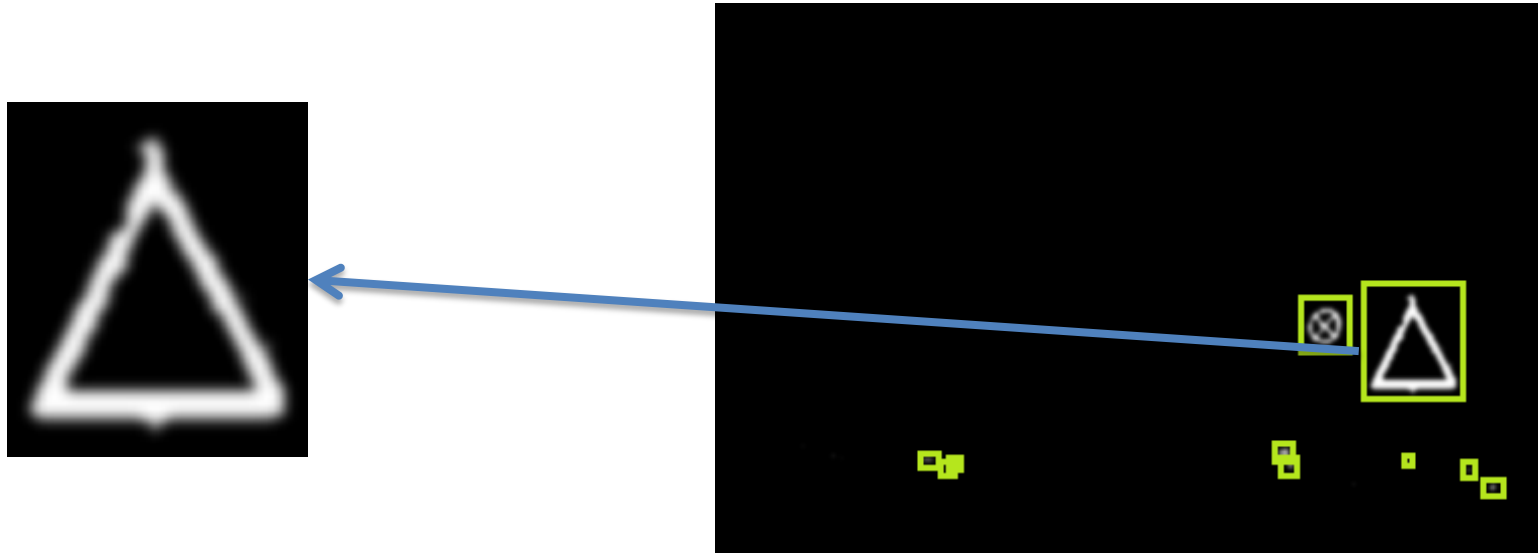
- Dunkelstufe im HSV-Farbraum wird auf den Maximalwert gesetzt
- Farben werden reiner
- Farbflächen heben sich stärker voneinander ab
- Die schlechten Lichtverhältnisse der Aufnahme wurden ausgeglichen



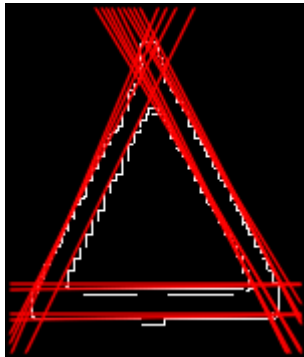
- Anwendung des Farbfilters
- Weichzeichnung des Bildes
- Finden von Blobs



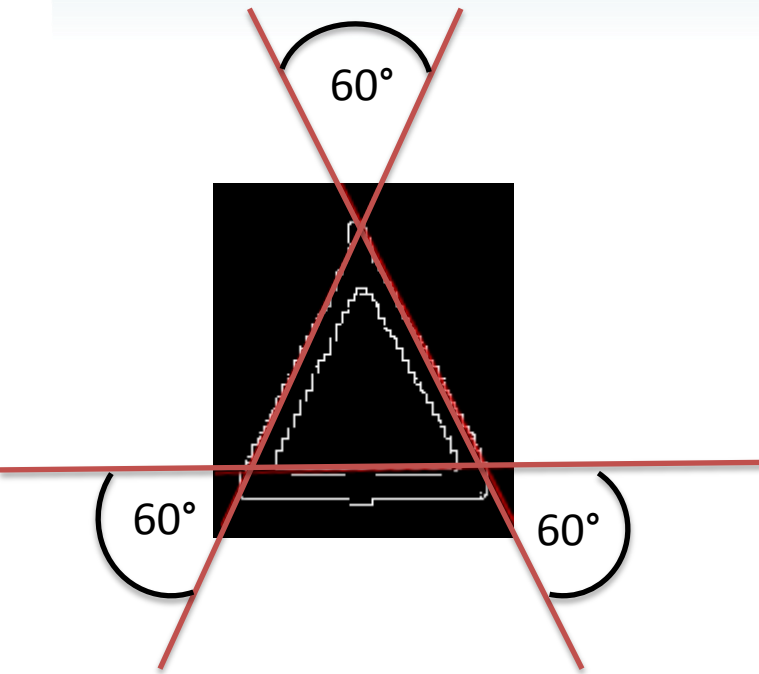
- Weichzeichner verwischt Unregelmäßigkeiten und lässt kleine Störungen verschwinden
- Blob: zusammenhängende Fläche von Pixeln mit der selben Farbinformation
- Bild wird anhand der gefundenen Blobs segmentiert



- Segmentierung des Bildes für die weitere Erkennung
- Zu kleine/große Blobs werden verworfen



- Anwendung einer Kantendetektion (Canny-Edge)
- Anwendung von Hough-Transformation zur Linienfindung
- Um eine gute Linienerkennung zu erreichen muss ein niedriger Threshold gesetzt werden
- Führt zur Erkennung von Geradenbüscheln
- Linien müssen zur weiteren Verarbeitung gefiltert werden

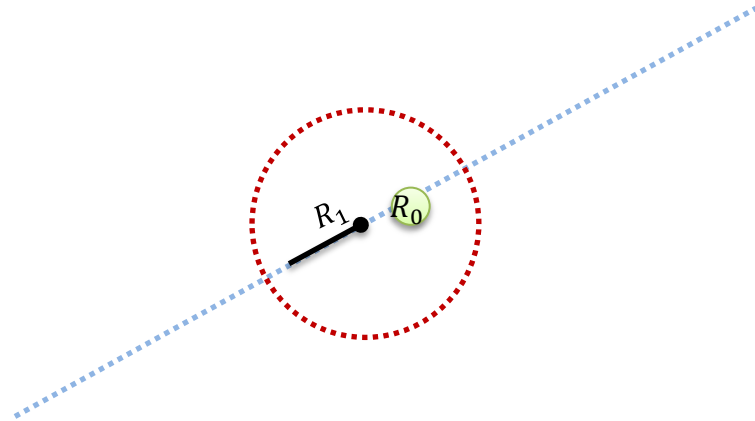
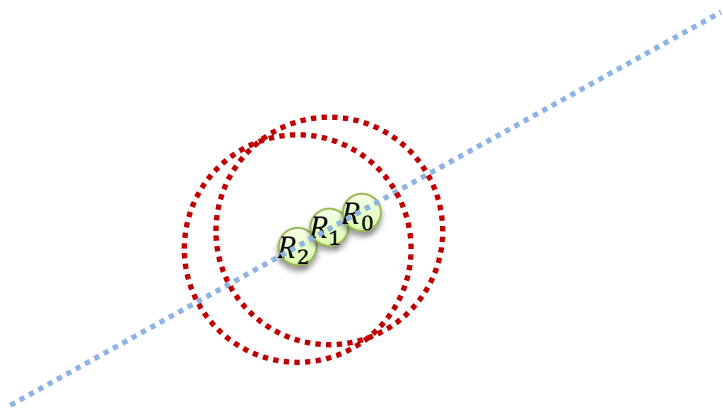


- Filterung der Geraden
- Betrachtung der Schnittwinkel zur Formerkennung

- Schneiden sich die 3 Linien jeweils in einem Winkel von 60° , wurde ein Dreieck erkannt
- In einem Viereck schneiden sich 4 Linien jeweils in einem Winkel von 90°
- In einem Achteck schneidet sich jeweils eine Linie mit der Übernächsten in einem Winkel von 90°
- Achtecke werden oft als Vierecke erkannt

- Das Objekt besitzt die gefilterte Farbe
- Und die erkannte Form
- Mit hoher Wahrscheinlichkeit ein Straßenschild
- Schilder treten in vielen Frames hintereinander auf, was zu vielen Erkennungen führt
- Schilder müssen getrackt werden
- Tracking mittels einfacher Klassifizierung über Form und einfachem Bewegungsschema

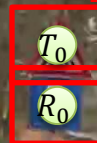
- Zwei Tracking-Objekte:
 - Tracking-Punkte
 - Tracking-Linien
- Tracking-Algorithmus:
 - Such zu Linien gehörende Punkte
 - Suche Punkte in einer Linie
- Vorgegebener Radius
- Vorgegebener Maximalabstand zur Linie
- Vorgegebenes maximales Framealter



Schildverfolgung

Frame 0

SIGN_TRIANGLE



SIGN_RECTANGLE

- Tracking Lines = 0 kein Line-Tracking
- Tracking Points < 3 kein Point-Tracking

Schildverfolgung

Frame 1



- Tracking Lines = 0 kein Line-Tracking
- Tracking Points = 4 Suche nach Punkten in einer Linie

Schildverfolgung

Frame 2



- Tracking Lines = 0 kein Line-Tracking
- Tracking Points = 6 Suche nach Punkten in einer Linie

Schildverfolgung

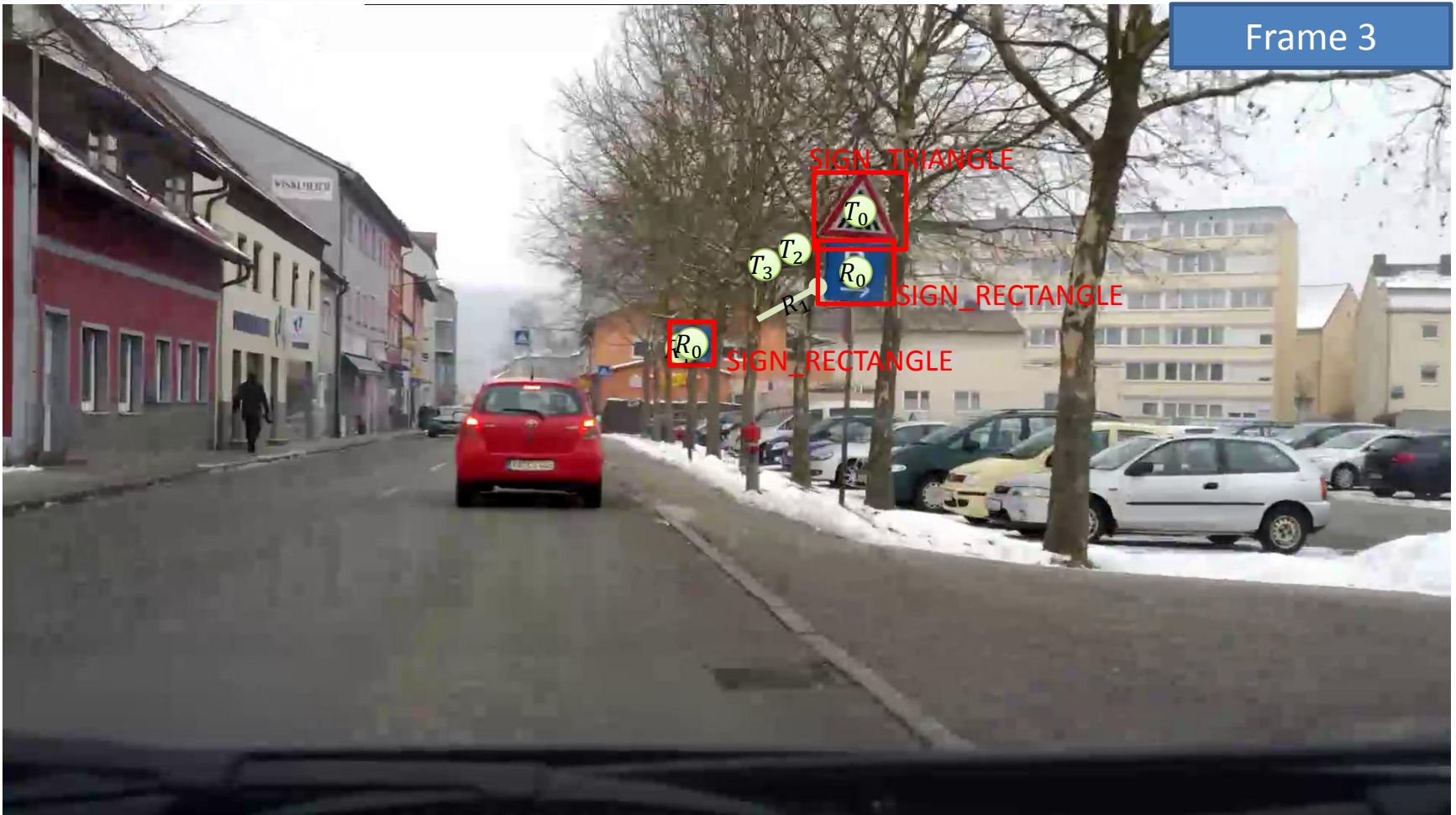
Frame 2



- Tracking Lines = 0 kein Line-Tracking
- Tracking Points = 6 Suche nach Punkten in einer Linie

Schildverfolgung

Frame 3





- Tracking Lines = 1 Such nach neuen Punkten



- Tracking Lines = 1 Such nach neuen Punkten
- Tracking Points = 5 Suche nach Punkten in einer Linie

Schildverfolgung

Frame 3



- Tracking Lines = 1 Such nach neuen Punkten
- Tracking Points = 5 Suche nach Punkten in einer Linie

Schildverfolgung

Frame 4



- Alle Tracking Objekte altern um 1 Frame

Schildverfolgung

Frame 5



Schildverfolgung

Frame 6



- Frame Delay ≥ 4 : Punkt R_4 wird gelöscht
- Schild gilt als erfolgreich erkannt

Schildverfolgung



- Es wurden 4 Schilder erkannt

- Frame:



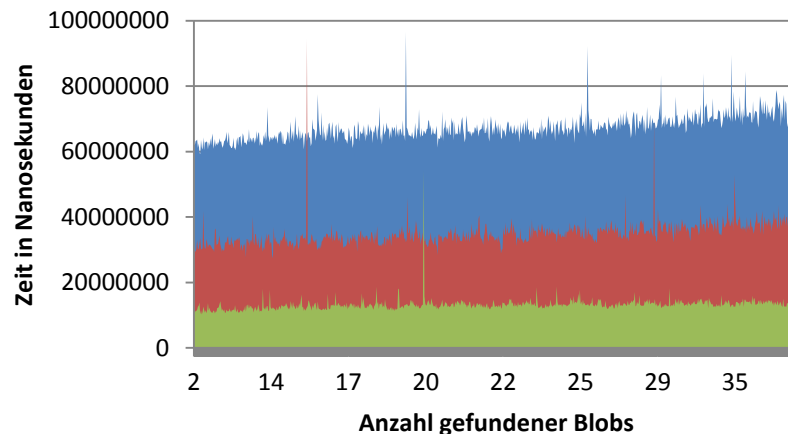
- Trotz Tracking kann ein Schild mehrfach erkannt werden
- Mehrfacherkennungen selten und überschaubar
- Schilder werden dem Benutzer zur Auswahl und Speicherung angezeigt

Demonstration

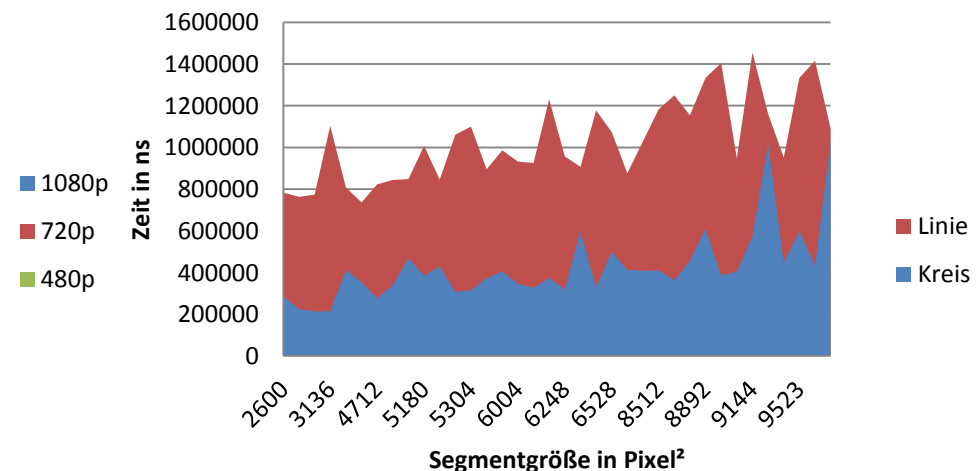
- Es wurde ein funktionierendes System zur Objekterkennung im Straßenverkehr geschaffen
- Plugin zur schnellen Schilderkennung
 - Skalierbare Erkennungsmethode: Hough-Transformation
 - Erkennung von Schildern mit Formerkennung
 - Erkennung beliebiger Schilder

Frameverarbeitung

HoughSignRecognition



Kreis- und Polygonerkennung



**VIELEN DANK FÜR IHRE
AUFMERKSAMKEIT**