

# Erheben einer Verkehrsstatistik durch Klassifizierung von Verkehrsobjekten unter Verwendung eines neuronalen Netzes

**Studienarbeit - T3201**

des Studiengangs Informatik

an der Dualen Hochschule Baden-Württemberg Stuttgart

von

**Florian Drinkler, Luca Stanger**

11. Juni 2021

Bearbeitungszeitraum  
Matrikelnummer, Kurs  
Ausbildungsfirma  
Betreuer

06.10.2020 - 11.06.2021  
6653948, 7474265, TINF-18B  
Balluff GmbH, camos GmbH, Stuttgart  
Sebastian Trost, Telefónica Germany

# Sperrvermerk

Der Inhalt dieser Arbeit darf weder als Ganzes noch in Auszügen Personen außerhalb des Prüfungsprozesses und des Evaluationsverfahrens zugänglich gemacht werden, sofern keine anderslautende Genehmigung der Ausbildungsstätte vorliegt.

Stuttgart, 11. Juni 2021

---

Florian Drinkler, Luca Stanger

# Erklärung

Wir versichern hiermit, dass wir unsere Studienarbeit - T3201 mit dem Thema: *Erheben einer Verkehrsstatistik durch Klassifizierung von Verkehrsobjekten unter Verwendung eines neuronalen Netzes* selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben. Wir versichern zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Stuttgart, 11. Juni 2021

---

Florian Drinkler, Luca Stanger

## Abstract

# Inhaltsverzeichnis

<b>Abkürzungsverzeichnis</b>	<b>VI</b>
<b>Abbildungsverzeichnis</b>	<b>VII</b>
<b>Tabellenverzeichnis</b>	<b>VIII</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation und Problemstellung . . . . .	1
1.2 Zielsetzung . . . . .	1
1.3 Methodik und Aufbau der Arbeit . . . . .	1
<b>2 Grundlagen und Stand der Forschung</b>	<b>2</b>
2.1 Implementierungsumgebung Jupyter . . . . .	2
2.2 Bildverarbeitungsalgorithmen mit OpenCV . . . . .	2
2.3 Maschinelle Lernverfahren . . . . .	2
2.3.1 Supervised Learning . . . . .	2
2.3.2 Unsupervised Learning . . . . .	2
2.3.3 Reinforcement Learning . . . . .	2
2.4 Datenstromorientierte Programmierung mit TensorFlow . . . . .	2
2.5 Deep Learning mit Keras . . . . .	2
2.6 Verwandte Arbeiten . . . . .	2
<b>3 Analyse der Datenströme</b>	<b>3</b>
3.1 Anforderungen an die Analyse . . . . .	3
3.2 Datenaufbereitung . . . . .	3
3.2.1 Datenerhebung und Integration . . . . .	3
3.2.2 Datenberechnung . . . . .	3
3.2.3 Datenaggregation . . . . .	3
3.2.4 Datenbereinigung . . . . .	3
<b>4 Entwicklung des Modells</b>	<b>4</b>
4.1 Vorverarbeitung der Daten . . . . .	4
4.2 Entwurf eines Netzwerks zur Klassifikation von Objekten . . . . .	4
4.4 Erheben einer Verkehrsstatistik . . . . .	4
4.5 Statistische Bewertung des Modells . . . . .	4
4.5.1 Determination aussagekräftiger Metriken . . . . .	4
<b>5 Prototypische Implementierung</b>	<b>5</b>
5.1 Aufbau des Prototypen . . . . .	5
5.2 Modellierung . . . . .	5

5.3	Deployment . . . . .	5
5.4	Anpassbarkeit . . . . .	5
<b>6</b>	<b>Evaluation des Prototypen</b>	<b>6</b>
6.1	Metriken zur Bewertung der Klassifikation . . . . .	6
6.2	Optimierung des neuronalen Netzes . . . . .	6
6.3	Evaluierung der Ergebnisse . . . . .	6
<b>7</b>	<b>Abschluss</b>	<b>7</b>
7.1	Fazit . . . . .	7
7.2	Ausblick . . . . .	7

# Abkürzungsverzeichnis

# Abbildungsverzeichnis



# Tabellenverzeichnis

# 1 Einleitung

## 1.1 Motivation und Problemstellung

## 1.2 Zielsetzung

## 1.3 Methodik und Aufbau der Arbeit

## 2 Grundlagen und Stand der Forschung

2.1 Implementierungsumgebung Jupyter

2.2 Bildverarbeitungsalgorithmen mit OpenCV

2.3 Maschinelle Lernverfahren

2.3.1 Supervised Learning

2.3.2 Unsupervised Learning

2.3.3 Reinforcement Learning

2.4 Datenstromorientierte Programmierung mit TensorFlow

2.5 Deep Learning mit Keras

2.6 Verwandte Arbeiten

# 3 Analyse der Datenströme

## 3.1 Anforderungen an die Analyse

## 3.2 Datenaufbereitung

### 3.2.1 Datenerhebung und Integration

### 3.2.2 Datenberechnung

### 3.2.3 Datenaggregation

### 3.2.4 Datenbereinigung

## 4 Entwicklung des Modells

### 4.1 Vorverarbeitung der Daten

### 4.2 Entwurf eines Netzwerks zur Klassifikation von Objekten

### 4.3

### 4.4 Erheben einer Verkehrsstatistik

### 4.5 Statistische Bewertung des Modells

#### 4.5.1 Determination aussagekräftiger Metriken

# 5 Prototypische Implementierung

## 5.1 Aufbau des Prototypen

## 5.2 Modellierung

## 5.3 Deployment

## 5.4 Anpassbarkeit

# 6 Evaluation des Prototypen

6.1 Metriken zur Bewertung der Klassifikation

6.2 Optimierung des neuronalen Netzes

6.3 Evaluierung der Ergebnisse

# 7 Abschluss

## 7.1 Fazit

## 7.2 Ausblick