

# Parknet

Urbanen Parkraum optimal nutzen mit KI basierten, autonom agierenden Fahrzeugen

Lukas Haverbeck, Frederik Schittny, Lukas Grave

Jugend Forscht 2020 Arbeitswelt

Unter Betreuung von Hendrik Büdding  
Annette-von-Droste-Hülshoff-Gymnasium Münster

## Kurzfassung

Durch ineffizientes Parken bleiben erhebliche Teile urbanen Parkraums ungenutzt, sodass sich Fahrzeiten und -strecken bei der Suche nach einem Parkplatz erhöhen. Vernetzte, autonom interagierende Fahrzeuge können zur Lösung dieses Problems beitragen, indem sie eine möglichst effiziente Anordnung der Fahrzeuge auf einem Parkstreifen autonom umsetzen und dabei intelligent auf das Verhalten anderer Fahrzeuge sowie ihr Umfeld reagieren. So würden sie ausparkenden Autos Platz schaffen, indem sie aufrücken. Auf diese Weise wird der Güter- und Personenverkehr bzw. der allgemeine Verkehrsfluss optimiert.

Die Ausgestaltung der Arbeit beschäftigt sich mit der Konzeptionierung und technischen Umsetzung dieses Systems. Dazu gehören die Planung von Kommunikationsprozessen zwischen den Fahrzeugen sowie der Entwurf als auch die Implementation der Steuerung und Interaktion der Akteure. Dies schließt auch die Entwicklung und Optimierung eines neuronalen Netzes für das autonome Fahren ein.

# Inhaltsverzeichnis

<b>Kurzfassung</b>	<b>1</b>
<b>1 Einleitung</b>	<b>3</b>
<b>2 Entwicklung</b>	<b>4</b>
2.1 Hardware . . . . .	4
Anforderungen . . . . .	4
Antrieb . . . . .	4
Motoransteuerung . . . . .	4
Sensoren . . . . .	5
2.2 Kommunikation . . . . .	5
Anforderungen . . . . .	5
Wahl der Kommunikationstechnologie . . . . .	6
Kommunikationsablauf . . . . .	6
Herausforderungen . . . . .	6
2.3 Interaktion . . . . .	7
Anforderungen . . . . .	7
Synchronisation der Formation auf dem Parkstreifen . . . . .	7
Kooperatives Verhalten . . . . .	8
Koordination des Verhaltens der Akteure . . . . .	9
2.4 Autonomes Parken und Fahren . . . . .	9
Anforderungen . . . . .	9
Autonome Parkplatzsuche . . . . .	9
Autonomes Ein- Ausparken . . . . .	10
Autonomes Fahren in komplexen Verkehrssituationen . . . . .	10
<b>3 Ergebnis</b>	<b>11</b>
<b>4 Ergebnisdiskussion</b>	<b>11</b>
4.1 Schlussfolgerungen . . . . .	11
4.2 Ausblick . . . . .	12
<b>5 Unterstützungsleistungen und Danksagung</b>	<b>13</b>
<b>A Informationen zur Implementation</b>	<b>14</b>
<b>B Verwendete Bauteile</b>	<b>14</b>
<b>C Abbildungsverzeichnis</b>	<b>14</b>
<b>D Quellen- und Literaturverzeichnis</b>	<b>15</b>

# 1 Einleitung

Unzureichend verfügbarer Parkraum stellt in fast allen deutschen Großstädten ein signifikantes Verkehrsproblem dar. Eine groß angelegte Studie des privaten Mobilitätsunternehmens INRIX aus dem Jahr 2017 hat gezeigt, dass der Mangel an Parkraum in Deutschlands Städten jährliche volkswirtschaftliche Kosten in Höhe von 40,4 Milliarden Euro verursacht [CP17], die sich vor allem auf den erhöhten Kraftstoffverbrauch, die verschwendete Zeit und den zusätzlichen Ausstoß von Abgasen bei der Parkplatzsuche zurückführen lassen [INR17]. Eine Untersuchung der Parkraumnutzung in London, deren Ergebnisse sich aber in ihren Grundaussagen auch auf deutsche Großstädte übertragen lassen, hat außerdem gezeigt, dass ein Großteil aller Parkvorgänge in einem geschäftlichen oder kommerziellen Zusammenhang steht [BL12, S. 35] und dass die Anreise zur Arbeit der häufigste Grund für das Suchen eines Parkplatzes ist [BL12, S. 7], weshalb mangelnder Parkraum zu Stoßzeiten zur Überlastung des Personen- und Güterverkehrs beiträgt.

Da die Schaffung zusätzlicher Parkkapazitäten aber mit hohen Kosten verbunden und gerade in dicht besiedelten Gebieten häufig kaum möglich ist, besteht ein begründetes Interesse an der effizienteren Ausnutzung bestehender Parkplätze. Dabei bieten vor allem Parkstreifen Optimierungspotenzial, da mit zunehmender Besiedlungsdichte der Anteil der auf der Straße geparkten Fahrzeuge zunimmt [BL12, S. 28]. Die konventionelle Nutzung der Parkplätze auf einem Parkstreifen birgt vornehmlich zwei Probleme:

1. Jedes parkende Fahrzeug muss nach vorne und hinten mindestens so viel Abstand halten, wie es selbst benötigt, um ausparken zu können. Dieser für das Ausparken benötigte Platz wird also nicht auf den gesamten Parkstreifen verteilt, obwohl in den meisten Fällen nur ein Fahrzeug auf einmal Platz zum Ausparken benötigt.
2. Da teilweise große Längenunterschiede zwischen parkenden Autos bestehen, ergeben sich Situationen, in denen verhältnismäßig kurze Autos Parklücken nutzen, in denen vormals ein deutlich längeres Fahrzeug geparkt hat, sodass Platz verschwendet wird, da das kurze Fahrzeug nun mehr Raum beansprucht als es benötigt.

Ständig wechselnde Parksituationen erzeugen also ein dynamisches Problem, das mit kooperativem Schwarmverhalten gelöst werden kann. Simultan zur zunehmenden Urbanisierung und der damit einhergehenden Verschärfung der Parksituation in vielen Städten lässt sich in der Automobilindustrie der Trend hin zum autonomen Fahren und zu einer größeren Vernetztheit der Fahrzeuge feststellen. Dabei gibt es bereits Systeme, die zwar autonom einparken können, ohne dabei jedoch die mögliche Vernetzung der Fahrzeuge zu nutzen. Eine durch Daimler und Benz geförderte Publikation von Markus Maurer et al., die sich mit autonomer Mobilität auseinandersetzt, nennt als einen möglichen Anwendungsfall autonomen Fahrens das autonome Valet-Parken:

“Der Fahrroboter stellt das Fahrzeug nach Verlassen der Passagiere und dem Ausladen von Transportgut in einer [...] Parkposition ab [...]. Der Fahrroboter fährt das Fahrzeug wieder von der Parkposition an eine Wunschadresse und besitzt die Möglichkeit und Berechtigung umzuparken. Der Fahrer spart die Zeit für die Parkplatzsuche, das Abstellen sowie die Fußwege eines entfernteren Parkplatzes. Außerdem wird der Zugang zum Fahrzeug erleichtert (räumlich wie zeitlich). Zusätzlich wird der Parkraum besser genutzt und die Parkplatzsuche effizienter gestaltet.” [Mau+15, S. 14f.]

Gleichzeitig gehen Maurer et al. aber davon aus, dass die effizientere Nutzung des Parkraums durch die “Bündelung des Parkraumangebots in Form von Sammelgaragen erfolgt” [Mau+15, S. 229]. Das Projekt Parknet untersucht die Möglichkeit, urbanen Parkraum mit Hilfe autonomer Mobilität und vernetzter Fahrzeuge auf Parkstreifen effizienter zu nutzen und verfolgt das Ziel, zu zeigen, dass die von Maurer et al. postulierte Notwendigkeit eines kosten- und zeitaufwendigen infrastrukturellen Wandels in Form der Schaffung von Sammelgaragen, dafür nicht notwendig ist. Es soll gezeigt werden, dass ein funktionierendes System autonomer, effizienter parkender Fahrzeuge schon heute technisch realisierbar ist.

Dafür wurde ein entsprechendes System konzeptionell entwickelt und modellhaft umgesetzt. Besonders hervorzuheben ist dabei die vernetzte Kommunikation der Akteure, die es den Fahrzeugen erlaubt, ihre Umgebung und Beziehung zu anderen Akteuren nachzuvollziehen und dabei ihre Aktionen intelligent aufeinander abzustimmen. Auf diese Weise wird ein hoher Effizienzgrad erreicht, da die Fahrzeuge auf den gesamten Parkstreifen verteilt lediglich so viel Abstand zueinander halten müssen, dass zu jedem Zeitpunkt das längste Fahrzeug ausparken kann, indem es diese Absicht kommuniziert und die anderen

autonomen Akteure darauf reagieren, indem sie zueinander aufrücken und dementsprechend Platz schaffen. Dadurch, dass das System in eng definierten Szenarien – nämlich der Parkplatzsuche, dem Einparken und dem Parken selbst – angewandt wird, kann allein mit Hilfe von Sensortechnik, klassischen Algorithmen sowie einem neuronalen Netz zur Steuerung ein Autonomiegrad der Stufe vier<sup>1</sup> erreicht werden.

Diese Arbeit beschäftigt sich mit der Entwicklung eines Prototyps des beschriebenen Systems, was vor allem die Konzeptionierung und den Bau von Modellfahrzeugen sowie die Implementation deren benötigter Software einschließt.

## 2 Entwicklung

Um die Funktionsfähigkeit des Systems zu demonstrieren, wurde ein Prototyp eines autonomen Fahrzeugs entwickelt, der als Referenzimplementation des konzeptionierten Systems dient. Im Folgenden wird auf den Bau des Prototyps, die Gestaltung der Kommunikation zwischen den Akteuren und deren Interaktion eingegangen.

### 2.1 Hardware

#### Anforderungen

Damit das Fahrzeug die durch das Projekt gestellten Fahraufgaben bewältigen kann und dabei erweiterbar ist, sollten die Bauteile folgende Anforderungen erfüllen:

- Die Hardware und die dazugehörigen Dokumentation sind einfach und kostengünstig zugänglich, um die Reproduktion und Weiterentwicklung des Projektes zu erleichtern.
- Die modulare Zusammensetzung der Komponenten ermöglicht die Erweiterung des Prototyps.
- Die Hardware muss eine hohe Präzision der Lenkung ermöglichen.

#### Antrieb

Das verwendete Grundmodell eines Elektrofahrzeugs verfügt im Lieferzustand über einen herkömmlichen Gleichstrommotor sowie über eine Antriebsübersetzung mit einem für Modellautos des Maßstabs eins zu zehn üblichen Aufbau aus einem Stirnrad sowie einem Antriebsritzel des Standards 48P<sup>2</sup>.

Diese Grundausstattung wurde ersetzt, da der Gleichstrommotor entsprechend seines Aufbaus bei geringen Stromflüssen nur eine ungenaue Geschwindigkeitskontrolle zuließ, die es erschwerte, Fahraufgaben wie das Ein- und Ausparken zu bewältigen, da diese hohe Präzision beim Fahren mit geringen Geschwindigkeiten voraussetzen.

Beim Umbau des Modells kam ein Schrittmotor vom Typ Nema17 [Ste] zum Einsatz. Die Verwendung eines Schrittmotors erlaubt dem Fahrzeug dabei auch sehr geringe Geschwindigkeiten und kleine Strecken mit großer Präzision zu fahren, da die kleinste fahrbare Einheit mit der verwendeten Kombination aus Motor und Übersetzung etwa 0,86 mm beträgt.

Um die vom Schrittmotor ausgeübte Kraft richtig übersetzen zu können, wurden die Antriebsräder mit Einheiten ersetzt, welche dem Standard 32P entsprechen. Diese kommen normalerweise in größeren Modellen mit dem Maßstab eins zu zwölf zum Einsatz und haben größere und damit stabilere Zähne, die auch für die Übertragung großer Kräfte geeignet sind.

#### Motoransteuerung

Um die verschiedenen Fahraufgaben, welche das System voraussetzt, bewältigen zu können, verwendet der entwickelte Prototyp eines Fahrzeugs einen Motor zum Lenken und einen weiteren Motor, der das Fahrzeug vorwärts und rückwärts bewegt.

---

<sup>1</sup>Ein autonomes Fahrzeug der Stufe vier gilt als vollautomatisiert und kann innerhalb definierter Systemgrenzen ohne eine externe, überwachende Instanz agieren [Gas12]

<sup>2</sup>48P und 32P beschreiben verschiedene Standards zum Aufbau von Getrieberädern, die sich auf den Zahnabstand beziehen. Dabei hat 32P größere Zähne, die weiter auseinanderstehen.

Die zentrale Steuereinheit ist dabei ein Einplatinencomputer der Bauart Raspberry Pi [Foub]. Dieser Computer wurde aufgrund seiner Fähigkeit gewählt, sowohl die Kommunikation über WLAN, die Datenverarbeitung als auch die Steuerung der Sensoren und Motoren bewältigen zu können.

Die Lenkung basiert dabei auf einem Servomotor mit angebauter Achse. Da dieser mittels Pulsweitenmodulation<sup>3</sup> (PWM) angesteuert wird, ist ein vom Raspberry Pi externer Servocontroller [Ada] verbaut. Dieser hat eine höhere Präzision bei der Ausgabe von PWM-Signalen als die im Raspberry Pi integrierten GPIO-Schnittstellen<sup>4</sup> und verfügt zudem über die Möglichkeit, eine externe Stromquelle anzuschließen, um den Raspberry Pi vor zu hohen Stromflüssen zu schützen. Da der Motor nicht direkt mit den Rädern verbunden ist, wurde anhand von Messdaten eine Funktion entwickelt, die den Einschlagwinkel der Räder in einen PWM-Wert umrechnet.

Der Antrieb des Fahrzeugs erfolgt mit Hilfe eines Schrittmotors. Für die Ansteuerung dieses Motortyps wird ebenfalls eine zusätzliche Steuereinheit [Pol] benötigt. Die Steuerung erfolgt über einen Schrittpin, der den Motor um  $\frac{1}{200}$  einer Umdrehung dreht sowie über einen Richtungspin. Auch hier wurde aufgrund der komplizierten Übersetzung mittels Zahnrädern und Zahnriemen eine neue Funktion ermittelt, welche die Anzahl an Schritten berechnet, die für eine bestimmte Distanz notwendig sind.

## Sensoren

Um dem Fahrzeug das Sammeln und Verarbeiten von Umgebungsdaten zu ermöglichen, kommen zum einen Abstandssensoren und zum anderen eine Kamera zum Einsatz.

Mit Hilfe der Ultraschallsensoren [Spa] wird die Distanz zwischen dem Fahrzeug und einem festen Körper ermittelt, indem sie ein Ultraschallsignal aussenden und aus der Zeitdifferenz zwischen dem Aussenden und dem Empfang des reflektierten Signals die vom Signal zurückgelegte Strecke berechnet wird. Am Fahrzeug sind insgesamt vier solcher Sensoren verbaut, um die Distanz zu Objekten nach vorne, hinten und nach rechts zu ermitteln. Der vierte Sensor ist in einem 45° Winkel nach hinten rechts angebracht, da es andernfalls nicht möglich wäre, den Abstand zu schräg stehenden Flächen zu messen. Dies liegt daran, dass das Signal an schrägen Flächen vom Sensor weg reflektiert wird und so eine Messung des reflektierten Signals nicht mehr erfolgen kann.

Die Kamera [Foua] ist so am Fahrzeug befestigt, dass sie den Bereich der Straße nach vorne erfasst. Das Fischaugenobjektiv bietet dabei einen großen Bildwinkel. Sie ist über einen CSI-2 Anschluss direkt mit dem Raspberry Pi verbunden und wird über diesen angesteuert.

## 2.2 Kommunikation

Ein wichtiger Bestandteil des Projekts liegt in der Gestaltung der Kommunikation zwischen den Fahrzeugen. Ein kooperatives Schwarmverhalten der Fahrzeuge wird erst durch den Datenaustausch zwischen den Akteuren möglich, da dieser es ihnen erlaubt, Informationen auszutauschen und darauf basierend intelligent aufeinander zu reagieren.

### Anforderungen

Für eine reibungslose Kommunikation müssen folgende Anforderungen erfüllt sein:

- Die Nachrichtenübertragung im Netzwerk läuft zuverlässig mit einer möglichst geringen zeitlichen Verzögerung ab.
- Es muss ein Datentransfer zwischen zwei und mehr Teilnehmern möglich sein.
- Damit nicht für jede Datenübertragung eine neue Verbindung aufgebaut werden muss, besteht die Verbindung dauerhaft.
- Es wird zwischen verschiedenen Arten von Nachrichten unterschieden.

---

<sup>3</sup>Pulsweitenmodulation ist ein System, mit dem die Spannung eines Ausgangs durch schnelles Ein- und Ausschalten künstlich zwischen dem Maximalwert und null Volt verstellbar ist.

<sup>4</sup>GPIO steht für "general purpose in/out" und ist eine Schnittstelle, welche mittels Software analoge und digitale Signale ausgeben und lesen kann.

- Alle Kommunikationsteilnehmer haben stets den gleichen Informationsstand

## Wahl der Kommunikationstechnologie

Für die Kommunikation der Akteure waren neben WLAN auch andere Technologien in der engeren Wahl. Im weiteren wird jede drahtlose Übertragungsart gesondert betrachtet und verglichen.

Zunächst stand die Funktechnik Bluetooth® im Raum, da diese auch im verwendeten Mini-Computer mit der Bluetooth-Version 5.0 bereits integriert ist, die laut c't Magazin [Živ17] eine maximal mögliche Übertragungsgeschwindigkeit von 2 MB/s zulässt. Um die Möglichkeit einer Erweiterung, die das Kamerasbild in Echtzeit überträgt, offen zu halten, ist es wichtig, dass auch größere Datenmengen schnell und stabil gesendet werden können, weshalb diese Funktechnik also nicht für das Projekt Parknet geeignet ist.

Anschließend wurde auch eine Kommunikation über das Internet in Betracht gezogen, wobei vorrangig der Standard MQTT<sup>5</sup> zu nennen ist. Grundvoraussetzung dafür ist eine stabile Internetverbindung. Zurzeit ist diese in Deutschland noch nicht verzögerungsfrei und flächendeckend gewährleistet. Da sich das Projekt möglichst nah an der Realität orientiert, kommt diese Methode ebenfalls nicht in Frage.

Schließlich wurde eine Datenübertragung in einem lokalen Netzwerk über eine WLAN-Verbindung in Erwägung gezogen. Laut c't Projekte 2019 [Sch19] ist mit dem im verwendeten Mini-Computer verbauten WLAN-Modul eine ausreichend schnelle Übertragungsrate von 8 MB/s realistisch. Zudem entspricht die Kommunikation über WLAN Industriestandards, wie die Studie von Igor Ostrowski, Piotr Szulewski und Andrzej Maslowski [OSM16] belegt. Des Weiteren erlaubt es eine WLAN-Verbindung, das Fahrzeug über eine lokale Webseite zu steuern. Deshalb fiel die Entscheidung auf den weit verbreiteten Standard WLAN.

## Kommunikationsablauf

Wie in den Anforderungen schon erwähnt, sollten alle Akteure der nahen Umgebung stets miteinander verbunden sein. Dies wird mit der folgenden Fallunterscheidung gewährleistet: Falls ein Kommunikationsteilnehmer kein bestehendes Netzwerk erkennt, wird er selbst tätig und erstellt einen eigenen WLAN-Access-Point<sup>6</sup>, dessen SSID<sup>7</sup> aus einer Kombination von "parknet" und dem aktuellen UNIX-Timestamp<sup>8</sup> besteht.

Bei einem oder mehreren bestehenden Netzwerken loggt sich der Teilnehmer in das Netzwerk ein, welches bereits am längsten besteht und daher die ID mit dem geringsten Zeitstempel hat. Das hat zur Folge, dass alle Akteure stets miteinander verbunden sind und jederzeit Daten austauschen können. Dafür betreibt jedes Fahrzeug einen Webserver<sup>9</sup>, der ankommende Anfragen annimmt und verarbeitet. Um Daten effizient an alle Geräte in einem Netzwerk zu senden, werden zunächst die jeweiligen IP-Adressen der Geräte mit Hilfe eines Netzwerkscans<sup>10</sup> ermittelt. Anschließend wird mittels einer Post-Request<sup>11</sup> die zu sendende Nachricht übertragen. Eine Nachricht besteht dabei aus einem Sender, einem Thema, einem Zeitstempel und einem Inhalt. Das Thema ist dabei von einer entscheidenden Bedeutung, da hierüber eine Zuordnung vorgenommen wird, wie die Nachricht verarbeitet werden soll. So würde z. B. eine Nachricht mit dem Thema "AppControl" an die Steuerung adressiert werden, die dann das Fahrzeug bewegt.

## Herausforderungen

Die größten Herausforderungen für die Kommunikation bestanden im Aufbau einer WLAN-Verbindung mit einem anderen Netzwerk. Der Einplatinencomputer sollte sich selbstständig mit einem neuen Netzwerk verbinden. Die Vorgängerversion des Mini-Computers, die gut erschlossen und dokumentiert ist, unterschied sich grundlegend von der neuen Version 4. Da das Projekt Parknet sich aber für die Version 4 des Raspberry Pi entschieden hat, war unvorhergesehenerweise auch der Prozess des WLAN-Aufbaus

<sup>5</sup>MQTT ist ein internetbasiertes Kommunikationsprotokoll.

<sup>6</sup>Schnittstelle für die drahtlose Datenübertragung, es wird ein neues eigenes WLAN erstellt

<sup>7</sup>SSID bedeutet Service Set Identifier und steht für den Namen eines WLAN Netzes.

<sup>8</sup>Zeit in Sekunden nach 1970

<sup>9</sup>Ein Webserver verarbeitet und beantwortet ankommende Anfragen.

<sup>10</sup>Mit Hilfe eines Netzwerkscans lassen sich alle IP-Adressen der Geräte in einem Netzwerk ermitteln.

<sup>11</sup>Eine Post-Request wird verwendet, um Daten an einen Webserver zu übertragen.

verschieden. Nach zahlreichen Versuchen und Recherchen, wurde eine Lösung gefunden, die auf der Schnittstelle mit dem Namen wireless von Josh Villbrandt [A15] basiert.

Ein weiteres Problem, welches es zu lösen galt, bestand darin, dass der Webserver für die Kommunikation nach einem Netzwerkwechsel nicht mehr verfügbar war. Um dieses Problem zu lösen, wurde versucht, den Webserver nach einem Wechsel neu zu starten. Dies gestaltete sich aufgrund veränderter IP-Adresse schwierig und so musste zunächst eine valide neue IP-Adresse ermittelt werden. Nach einer gewissen Zeitspanne beim Netzwerkwechsel ist auch die neue IP-Adresse verfügbar. Somit wurden beide Problem gelöst.

## 2.3 Interaktion

Die angestrebte Effizienzsteigerung bei der Nutzung von Parkraum erfordert dahingehend eine Interaktion der Fahrzeuge, dass die autonomen Akteure dynamisch auf das Verhalten anderer Fahrzeuge reagieren, indem sie einander Platz schaffen oder platzsparend zueinander aufrücken. Außerdem müssen die einzelnen Handlungsabsichten der Fahrzeuge untereinander koordiniert werden, um etwa sicherzustellen, dass stets maximal ein Fahrzeug ausparkt. Diese Art der Interaktion zwischen den Fahrzeugen wird durch ihre Vernetztheit ermöglicht, die einen ständigen Informationsaustausch über die zuvor beschriebenen Kommunikationsschnittstelle erlaubt.

### Anforderungen

Um die Funktionsfähigkeit gänzlich unüberwachter autonomer Interaktion sicher gewährleisten zu können, müssen die folgenden Anforderungen erfüllt sein:

- Jeder Akteur hält während des Parkens genau so viel Abstand nach vorne, dass auf die gesamte Reihe autonomer Fahrzeuge verteilt genügend aber nicht mehr Platz als nötig für den Fall zur Verfügung steht, dass das längste Fahrzeug der Kette beabsichtigt, auszuparken.
- Damit jeder Akteur einem ausparkenden Fahrzeug Platz schaffen kann, muss jederzeit eindeutig bestimmbar sein, in welche Richtung er dafür aufrücken muss.
- Um Kollisionen zu vermeiden, muss jederzeit eindeutig bestimmbar sein, welcher Akteur welche Aktion ausführen darf.
- Die Abstimmung der Interaktion berechtigt alle beteiligten Akteure gleich und erfolgt ohne eine koordinative, externe Instanz.

### Synchronisation der Formation auf dem Parkstreifen

Die Formation einer Menge von Akteuren bezeichnet ihre Reihenfolge inklusive der Information über die Länge des längsten Fahrzeugs, das sie beinhaltet. Für die Ermittlung des optimalerweise nach vorne zu haltenden Abstands und damit beim Platzschaffen für ein ausparkendes Fahrzeug in die richtige Richtung aufgerückt werden kann, muss jeder Akteur seine Formation kennen. Die zentrale Problemstellung besteht dabei darin, dass jeder Akteur nur über einen Teil der benötigten Informationen - nämlich seine eigene Länge und seine relative Positionierung zu seinem Vordermann - verfügt. Folglich erfordert die Bestimmung der Formation einen Austausch unter den Akteuren innerhalb des lokalen Netzwerkes, sodass die Teilinformationen zu einer vollständigen Formation zusammengesetzt werden können. Dieser Informationsaustausch erfolgt nach dem im Folgenden beschriebenen Algorithmus, der fortlaufend wiederholt wird, um die Formation aktuell zu halten, sollten einzelne Akteure den Parkstreifen verlassen oder weitere Akteure einparken.

Über die Frontkamera des Fahrzeugs kann ein Akteur die Signatur seines Vordermannes eindeutig identifizieren. Sollte ein Akteur keine solche Signatur im Kamerabild finden, bedeutet dies, dass er das erste Mitglied einer Formation ist. In diesem Fall sendet der Akteur eine von ihm signierte Formation, die zunächst nur ihn selbst enthält, an alle anderen Fahrzeuge innerhalb seines lokalen Netzwerkes. Obwohl alle Akteure diese Nachricht erhalten, reagiert nur derjenige darauf, der die Signatur der Formation als die seines Vordermannes identifiziert, indem er zunächst bestätigt, dass er eine Formation von seinem Vordermannes erhalten hat, sich selbst zur Formation hinzufügt und die aktualisierte Formation dann erneut innerhalb des Netzwerkes teilt. Dieser Vorgang wird so lange wiederholt, bis der Akteur, der



zuletzt eine Formation signiert und versandt hat, keine Empfangsbestätigung erhält. Das bedeutet, dass sich hinter diesem Akteur kein weiterer befindet und die zuletzt versandte Formation daher vollständig ist. Zu diesem Zeitpunkt verfügt lediglich das letzte Mitglied der Formation über eine vollständige Repräsentation dieser und sendet sie deshalb mit der Information über ihre Vollständigkeit an alle Akteure innerhalb des Netzwerkes. Damit diese die Formation nun auch als ihre vollständige Formation akzeptieren können, muss jeder Akteur für sich validieren, ob er selbst in der Formation enthalten ist, da andernfalls der Sonderfall, dass sich mehrere abgeschlossenen Formationen dasselbe Netzwerk teilen, zu dem Problem führen würde, dass die Akteure nicht zwischen den Formationen unterscheiden könnten und daher eine fremde Formation akzeptieren würden. Um darüber hinaus sicherzustellen, dass jeder Akteur immer nur die neueste Version einer Formation verarbeitet, wird jede Formation zusammen mit einem Zeitstempel versandt, sodass überprüft werden kann, ob der Akteur bereits eine neuere Version der Formation kennt.

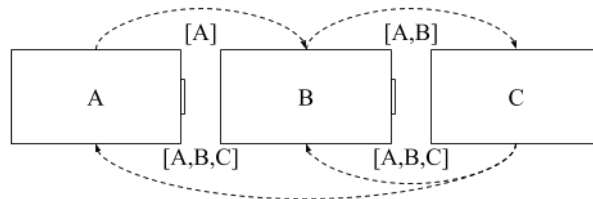


Abbildung 1: Illustration des Informationsaustauschs für die Bestimmung der Formation

Die Implementation dieses Algorithmus verwendet die zuvor beschriebene Kommunikationsschnittstelle, die es erlaubt, typisierte Nachrichten zu versenden, sodass zwischen vollständigen und unvollständigen Formationen unterschieden werden kann. Die Identifikation der Signatur des Vordermannes erfolgt über einen QR-Code am Heck des Fahrzeuges, der seine eindeutige interne ID repräsentiert. Eine Formation wird dann als vollständig angesehen, wenn ihr Empfang nicht innerhalb eines zeitlichen Schwellwertes bestätigt wird, was dadurch möglich ist, dass die Kommunikation in dem lokalen Netzwerk mit konstant geringer zeitlicher Verzögerung erfolgt.

### Kooperatives Verhalten

Die Vernetztheit der Akteure ermöglicht es ihnen, intelligent auf das Verhalten anderer Fahrzeuge innerhalb ihres Netzwerks zu reagieren. Diese Kooperation wird in eine Effizienzsteigerung bei der Nutzung des Parkraums umgesetzt, indem die Fahrzeuge basierend auf ihrer Formation auf einem Parkstreifen den minimal nach vorne zu haltenden Abstand berechnen und autonom einnehmen. Gegeben eine Formation mit  $n$  Akteuren, die Länge  $l_{max}$  des längsten Fahrzeugs dieser Formation, eine Konstante  $b$ , die das Verhältnis zwischen der Länge eines Fahrzeugs und seines zum Ausparken benötigten Platzes beschreibt, und einen Sicherheitsabstand  $s$ , ist dieser minimale Abstand  $d$  definiert durch:

$$d = \frac{1}{n} \cdot b \cdot l_{max} + s$$

Während des Parkens wird bei einer Veränderung der Formation der einzuhaltende Abstand neu berechnet und dann autonom von jedem Akteur umgesetzt, indem die Fahrzeuge basierend auf ihren Sensordaten entsprechend vor- oder zurückrücken.

Da der für das Ausparken notwendige Platz auf diese Weise auf die gesamte Kette von Fahrzeugen verteilt wird, sind die Akteure aus diesem Zustand heraus nicht dazu in der Lage, auszuparken, weshalb ein sicheres und unkompliziertes Ausparken erneut kooperatives Verhalten erfordert. Ein ausparkendes Fahrzeug kommuniziert seine Absicht, den Parkstreifen zu verlassen innerhalb seiner Formation, indem es eine entsprechende Nachricht über die Kommunikationsschnittstelle sendet. Die anderen Akteure reagieren so darauf, dass sie basierend auf der bekannten Formation so lange in diejenige Richtung aufrücken, in der sie dem ausparkenden Fahrzeug Platz schaffen, bis sie lediglich den Sicherheitsabstand  $s$  halten. Auf diese Weise wird vor und hinter dem ausparkenden Akteur so viel Platz frei, wie das längste Fahrzeug benötigen würde, um auszuparken. So ermöglicht es die Kooperation zwischen den Akteuren, einen geringen Abstand zu halten, während gleichzeitig zu jedem Zeitpunkt sichergestellt ist, dass jedes

Fahrzeug den Parkstreifen wieder verlassen kann. Nachdem ein Akteur den Parkstreifen verlassen hat, ist er nicht mehr Teil der Formation und die übrigen Fahrzeuge beginnen erneut, zueinander aufzurücken, um auch den frei gewordenen Platz optimal zu nutzen.

### **Koordination des Verhaltens der Akteure**

Um Kollisionen zu vermeiden, muss sichergestellt werden, dass das Verhalten der autonomen Akteure aufeinander abgestimmt ist. So dürfen beispielsweise zu keinem Zeitpunkt zwei Fahrzeuge gleichzeitig versuchen, ausparken, da der zur Verfügung stehende Platz dafür nicht ausreicht. Um gewährleisten zu können, dass die einzelnen Aktionen der Akteure miteinander kompatibel sind, kommt ein Verfahren zur Bestimmung der Handlungsberechtigung eines Akteurs zum Einsatz, das im Folgenden beschrieben wird.

Jede Aktion, die ein Akteur beabsichtigt auszuführen, wird innerhalb der Formation mit dem aktuellen Zeitstempel und der Signatur des agierenden Fahrzeugs geteilt und dahingehend von den anderen Akteuren verarbeitet, dass diese validieren, ob die empfangene Aktion ausgeführt werden darf oder nicht. Im Sinne der Gleichberechtigung der Akteure ist das genau dann der Fall, wenn kein anderer Akteur einer höher priorisierten Aktion, also eine Aktion mit einem geringeren Zeitstempel, ausführt oder plant auszuführen. Falls es eine solche höher priorisierte Aktion innerhalb der Formation gibt, so wird diese nach demselben Verfahren im Netzwerk der Akteure geteilt. Ein Akteur verifiziert seine Berechtigung zu einer Aktion, indem er überprüft, ob nach einem bestimmten zeitlichen Schwellwert seine Aktion nicht durch die eines anderen Akteurs überschrieben worden ist. Sollte dies nicht der Fall sein, führt er die Aktion aus und sendet im Anschluss daran eine Bestätigung über die Terminierung des Vorgangs. Gleichzeitig speichert jeder Akteur die momentan am höchsten priorisierte Aktion und reagiert darauf, sodass es zu einem kooperativen Schwarmverhalten kommt, das ausparkende Fahrzeuge darin unterstützt, den Parkstreifen zu verlassen. Dabei ist zu beachten, dass es eine Menge von Aktionen gibt, die keine koordinative Erlaubnis erfordern, wohl aber voraussetzen, dass eine solche nicht zeitgleich ausgeführt wird. So sind alle Akteure so lange simultan dazu berechtigt, zueinander aufzurücken, um Platz zu sparen, bis ein Fahrzeug seine Absicht, ausparken kommuniziert. Auch darf jeder Akteur nur solche Aktionen verarbeiten und darauf reagieren, die innerhalb seiner eigenen abgeschlossenen Formation gesendet worden sind, weshalb bei jeder der genannten Operationen die mit der Aktion gesendete Signatur dahingehend zu validieren ist, ob sie Teil der Formation des Akteurs ist.

## **2.4 Autonomes Parken und Fahren**

Um den Fahrer bei der Parkplatzsuche und beim Einparken selbst zu entlasten, sucht das autonome Fahrzeug selbstständig einen freien Parkplatz auf einem Parkstreifen am rechten Straßenrand und parkt dort ein.

### **Anforderungen**

Dafür müssen die folgenden Anforderungen erfüllt sein:

- Das Fahrzeug findet selbstständig eine passende Parklücke auf einem Parkstreifen am rechten Straßenrand.
- Das Fahrzeug parkt autonom in die Parklücke ein.

### **Autonome Parkplatzsuche**

Bei der Suche nach einem Parkplatz berechnet der Akteur fortlaufend, ob am rechten Straßenrand eine passende Parklücke vorhanden ist. Dies ist dann der Fall, wenn die vom rechten Ultraschallsensor gemessene Tiefe der Lücke in einem angemessenen Verhältnis zur Breite des Fahrzeugs steht und die Länge der Parklücke zum Einparken ausreicht.

Das Fahrzeug fährt mit konstanter Geschwindigkeit entlang der Straße und misst dabei fortlaufend den Abstand zum rechten Straßenrand. Sobald dieser Abstand über eine Strecke, die der benötigten Länge der Parklücke entspricht, ausreichend war, wurde eine passende Parklücke identifiziert. Anschließend fährt das Fahrzeug bis zum Beginn der Parklücke vor, um von dort aus einparken zu können.

## Autonomes Ein- Ausparken

Nachdem das Fahrzeug einen passenden Parkplatz gefunden hat, befindet es sich stets in der gleichen Position zur Parklücke. Um autonom einzuparken, fährt es zunächst im maximal möglichen Winkel rückwärts in die Parklücke hinein, bis der Abstand zur rechten Seite einen Schwellwert erreicht, nach dessen Überschreitung es nicht mehr möglich wäre, gegenzulenken. Daraufhin lenkt der Akteur vollständig in die Gegenrichtung ein, um das Fahrzeug in eine parallele Stellung zur Parklücke zu bringen. Dadurch, dass der Abstand nach vorne kontinuierlich durch das System minimiert wird, rückt das Fahrzeug zu seinem Vordermann auf und hält dabei mindestens den Sicherheitsabstand ein.

## Autonomes Fahren in komplexen Verkehrssituationen

Die innerstädtische Steuerung eines Fahrzeuges lässt sich kaum durch eine Menge expliziter Regeln beschreiben. Damit ein Fahrzeug dennoch gänzlich autonom einen vakanten Parkplatz auf einem Parkstreifen finden kann, muss es dazu in der Lage sein, auch in komplexen Verkehrsszenarien selbstständig zuverlässig lenken zu können. Forscher des Grafikkartenherstellers NVIDIA haben gezeigt, dass dieses Problem mit Hilfe eines konvolutionellen neuronalen Netzes gelöst werden kann [Boj16, S. 1], indem es darauf trainiert wird, basierend auf dem momentanen Lenkungswinkel des Fahrzeugs und einem Bild der Frontkamera, einen neuen Steuerungswinkel zu berechnen, der das Fahrzeug auf der Fahrbahn hält. Ein solches neuronales Netz wurde mit Hilfe der Deep-Learning-Bibliotheken Tensorflow und Keras implementiert, wofür die in der NVIDIA-Studie von Mariusz Bojarski et al. verwendete Netzarchitektur [Boj16, S. 4] übernommen wurde.

Das neuronale Netz verarbeitet als Eingabedaten den momentanen Lenkungswinkel in Grad sowie eine quantifizierte und normalisierte, dreidimensionale Pixelrepräsentation eines RGB-Bildes der Frontkamera. In den ersten fünf Schichten des neuronalen Netzes erfolgen die Konvolutionen, die der Erkennung visueller Muster in den Bilddaten dienen. Darauf folgt in insgesamt vier fully-connected Schichten die Berechnung des Ausgabewerts für den neuen Lenkungswinkel. Mit Ausnahme der letzten beiden Schichten, die jeweils eine lineare Aktivierungsfunktion haben, werden alle Schichten mit der ELU-Funktion aktiviert. Das Netz verfügt über etwa 713.000 anpassbare Parameter.

Um die Eingabegröße zu reduzieren, und da für die Steuerung eines Fahrzeuges vor allem das Zentrum des Kamerabildes relevant ist, wird nur ein zentraler Ausschnitt des Bildes von 100x200 Pixeln betrachtet. Während des Trainings wird im Sinne der Datenaugmentierung zur Förderung der Generalisierung durch das Netz in 50% der Fälle das Kamerabild an der vertikalen Achse gespiegelt, das Bild zufällig marginal verschoben und ein zufälliges Rauschen zu den Bildpunkten hinzugefügt. Darüber hinaus wird zur Förderung der Generalisierung eine Regularisierung der Parameter nach der L2-Norm angewandt. Das neuronale Netz wird überwacht auf in realen Szenarien erfasste Daten trainiert, indem die absolute Differenz zwischen den vorhergesagten und tatsächlich vorgenommenen Lenkungswinkeln minimiert wird.

Der entwickelte Prototyp besitzt die Fähigkeit, während des manuellen Fahrens Daten über das Fahrverhalten zu sammeln, auf die das neuronale Netz dann trainiert werden kann. Dafür wird das momentane Kamerabild gespeichert und einem vorherigen sowie dem tatsächlich vorgenommenen Lenkungswinkel zugeordnet. Da der Prototyp allerdings auf die Handhabung von Parkszenarien ausgelegt ist, erreicht das Fahrzeug lediglich Geschwindigkeiten, bei denen keine effiziente Datensammlung in einem für das Training ausreichenden Maße möglich war. Aus diesem Grund wurde das neuronale Netz auf einen öffentlichen Datensatz [A16] mit etwa 45.000 entsprechenden Datenpunkten aus realen Verkehrsszenarien trainiert. Dabei konnte auf die Validierungsdaten eine mittlere Abweichung des Lenkungswinkels von  $1,10^\circ$  erreicht werden, was im Verhältnis zum durchschnittlichen, tatsächlich vorgenommenen Lenkungswinkel einer Genauigkeit von über 90% entspricht.

In Kombination mit der eingesetzten Sensortechnik, die der Einhaltung von Sicherheitsabständen und der Plausibilitätsprüfung der vom neuronalen Netz ermittelten Lenkungswinkel dienen würden, könnte dieses neuronale Netz zur autonomen Steuerung des Fahrzeugs in urbanen Gebieten bei geringen Geschwindigkeiten eingesetzt werden. Dabei ist allerdings zu beachten, dass bestimmte Verkehrsszenarien, wie etwa Baustellen, in dem verwendeten Datensatz unterrepräsentiert sind und das neuronale Netz in solchen Situationen mit hoher Wahrscheinlichkeit keine zielführenden Werte ermitteln könnte.

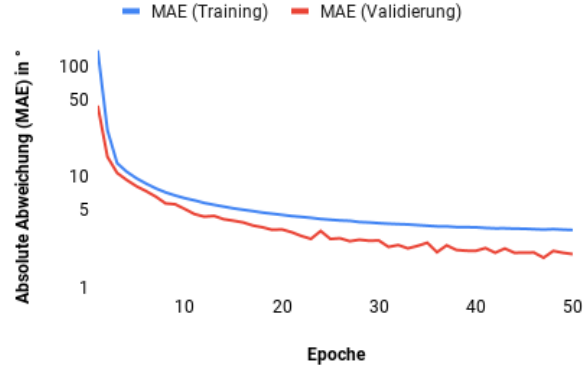


Abbildung 2: Mittlere Abweichung des Lenkungswinkels im Trainingsverlauf (logarithmisch skaliert)

### 3 Ergebnis

Der Platz, der von herkömmlichen Fahrzeugen auf einem Parkstreifen mit  $n$  Fahrzeugen mit der mittleren Länge  $l$  und dem mittleren Abstand  $a_m$  zueinander benötigt wird, wird durch  $d_1(n) = n \cdot l + n \cdot a_m$  beschrieben.

Der Platz, der von den  $n$  autonomen Akteuren mit der mittleren Länge  $l$ , dem maximal für ein einzelnes Fahrzeug zum Ausparken benötigten Abstand  $a_{max}$  und dem Sicherheitsabstand  $s$  zueinander benötigt wird, wird durch  $d_2(n) = n \cdot l + n \cdot s + a_{max}$  beschrieben.

Damit die autonomen Akteure im Vergleich zu den herkömmlichen Fahrzeugen einen Parkplatz einsparen, muss gelten:

$$\begin{aligned}
 d_1(n) &= d_2(n+1) \\
 \Leftrightarrow n \cdot l + n \cdot a_m &= (n+1) \cdot l + (n+1) \cdot s + a_{max} \\
 \Leftrightarrow n \cdot a_m &= l + (n+1) \cdot s + a_{max} \\
 \Leftrightarrow n(a_m - s) &= l + s + a_{max} \\
 \Leftrightarrow n &= \frac{l+s+a_{max}}{a_m-s}
 \end{aligned}$$

Geht man von einer mittleren Länge der Autos von  $l=4,4\text{m}$ , einem durch die präzisen Sensoren möglichen Sicherheitsabstand von  $s=0,05\text{m}$ , einem mittleren Abstand von  $a_m=0,85\text{m}$  und einem maximal benötigten Abstand von  $a_{max}=1\text{m}$  aus, so lässt sich  $n$  bestimmen als:

$$n = \frac{4,4\text{m}+0,05\text{m}+1\text{m}}{0,85\text{m}-0,05\text{m}} = \frac{5,45\text{m}}{0,8\text{m}} \approx 6,81$$

Demnach würde auf einem Parkstreifen mit sieben herkömmlich geparkten Fahrzeugen ein weiterer Parkplatz nutzbar, sodass acht autonome Fahrzeuge Platz fänden. Das entspricht einer Effizienzsteigerung von etwa 15%.

## 4 Ergebnisdiskussion

### 4.1 Schlussfolgerungen

Anhand der Schilderung der Konzeptionierung und der Entwicklung des Systems inklusive eines Prototyps konnte gezeigt werden, dass autonome Mobilität mit vernetzten Akteuren zu einer effizienteren Nutzung von Parkraum beitragen kann, ohne einen Umbau der Infrastruktur zu erfordern. Durch das intelligente, kooperative Verhalten der Akteure können Fahrer, Umwelt und Verkehr entlastet werden.

Um die volle Effizienzsteigerung zu erreichen, ist allerdings ein signifikanter Anteil autonomer Akteure an den parkenden Fahrzeugen nötig. Außerdem hat sich gezeigt, dass bei der momentanen Ausgestaltung des Systems ein gewisses Maß an Kontrolle durch eine Person notwendig ist, um den Vorgang zu überwachen und so in bestimmten Sonderfällen, wie z. B. das Parken auf kostenpflichtigen Parkflächen, eingreifen zu können.

## 4.2 Ausblick

Eine Nachrüstung herkömmlicher Autos ist deshalb möglich, da der entwickelte Prototyp so entworfen wurde, dass sämtliche verwendeten Größen für die Anwendung in realen Fahrzeugen verschiedener Bauarten skalierbar sind. Teile der benötigten Hardware sind durch die zunehmende Verbreitung von Assistenzsystemen bereits in vielen Fahrzeugen verbaut oder können einfach und kostengünstig nachgerüstet werden. Dies erfordert allerdings entweder die Verwendung des Systems durch Automobilhersteller oder die Bereitstellung standardisierter Schnittstellen zur Ansteuerung des Fahrzeuges durch externe Software, sodass das System ebenfalls in bestehende Fahrzeugtypen integriert werden könnte.

Bei der Verwendung realer Fahrzeuge kann es durch die größeren Distanzen zwischen den Fahrzeugen zu Situationen kommen, in denen die beschränkte Reichweite eines WLAN-Netzwerks<sup>12</sup> die Kommunikation zwischen den Fahrzeugen verhindert. Dabei würde das aktuelle System, wie in Abbildung 3 gezeigt, neue Netzwerke erstellen, was in einen Zustand führen würde, der einen korrekten Kommunikationsablauf unterbindet. Ein hinzukommendes Fahrzeug, das beide Netzwerke erreichen kann, ist nur dazu in der Lage, sich mit einem Netzwerk zu verbinden. Deshalb wäre es optimal, wenn Fahrzeuge, die WLAN-Verbindung verstärken würden, indem sie ein eigenes Netzwerk mit dem gleichen Namen eröffnen.

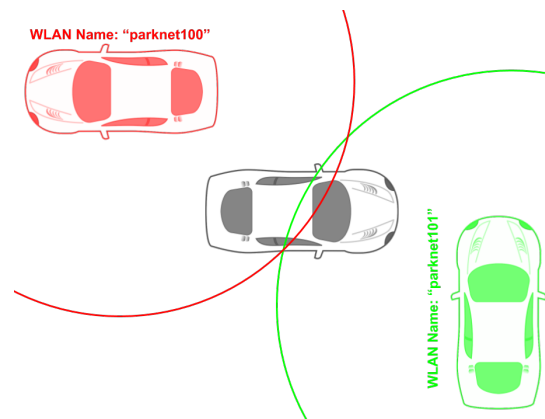


Abbildung 3: Beispielfhafte Problemsituation überlappender Netzwerke

Einen wichtigen Entwicklungsschritt stellt die Erweiterung des Systems um zusätzliche Anwendungsszenarien, wie das Parken in Parkhäusern, dar, da die Anwendung bislang nur auf Parkstreifen möglich ist. Auf diese Weise kann eine noch größere Effizienzsteigerung bei der Nutzung von Parkraum erreicht werden. Die Implementation einer Ruffunktion, die es ermöglicht, dass das Fahrzeug kurze Strecken autonom zurücklegt, um Personen in der Nähe des Parkplatzes abzuholen, würde die Benutzerfreundlichkeit erhöhen und die Integration in Carsharing-Konzepte ermöglichen, die das Abstellen dieser autonomen Fahrzeuge zwischen Stoßzeiten effizienter gestalten würde und wie von Markus Maurer et al. geschildert erfolgen könnte:

“Erreicht ein Fahrer sein Fahrtziel [...], stoppt dieser das Fahrzeug, steigt aus und gibt dem Fahrroboter den Befehl, das Fahrzeug abzustellen. Somit kann der Fahrroboter das Fahrzeug nun zu einem privaten Parkplatz, einem öffentlichen Abstellplatz oder einem Parkplatz des Dienstleisters wie z. B. einer Carsharing-Flotte bringen. [...] Der Fahrroboter fährt das Fahrzeug an diese Zieladresse und hält an, sodass der Fahrer einsteigen und die Fahraufgabe übernehmen kann.” [Mau+15, S. 15]

Für die Betreiber großer Fahrzeugflotten ist diese Lösung außerdem deshalb besonders attraktiv, da so eher sichergestellt ist, dass ein erheblicher Anteil autonomer Fahrzeuge zusammen abgestellt wird.

Bevor das System aber praktisch angewandt werden kann, ist eine Erweiterung der Sicherheitsvorkehrungen angedacht. Zwar verfügt der Prototyp bereits über eine Funktion, die die Einhaltung eines Sicherheitsabstandes zu jedem Zeitpunkt sicherstellen soll, allerdings werden bewegliche Objekte geringer Größe

<sup>12</sup>Im Normalfall beträgt die Reichweite eines WLAN-Netzwerks etwa 20m.

dabei nicht zuverlässig erkannt. Die Gewährleistung der Sicherheit von z. B. Personen könnte durch die Integration bestehender Algorithmen zur Objekterkennung erreicht werden. Um darüber hinaus die Sicherheit vor der Einflussnahme Dritter zu erhöhen, müsste die Verschlüsselung der Kommunikation und die Authentifikation der Akteure erweitert werden. In der aktuellen Umsetzung werden die Nachrichten nicht verschlüsselt, weshalb es Außenstehende leicht möglich wäre, falsche Daten zu versenden und somit das Verhalten der Akteure zu manipulieren.

## **5 Unterstützungsleistungen und Danksagung**

Abschließend möchten wir uns bei all denjenigen bedanken, die uns bei der Umsetzung dieses Projekts unterstützt haben. Das schließt vor allem unseren Betreuer Hendrik Büdding ein, der uns neben fachlichen Anregungen auch methodisch sowie bei der Planung und Finanzierung dieses Projekts geholfen hat. Unser besonderer Dank gilt natürlich auch der Gerlind & Ernst Denert-Stiftung und ebenso der Sparkasse Münsterland Ost sowie dem Förderverein des Annette-von-Droste-Hülshoff-Gymnasiums für ihre finanzielle Unterstützung, ohne die wir dieses Projekt nicht hätten umsetzen können. Wir bedanken uns außerdem bei Herrn Professor Denert für seine fachlichen Anregungen.

## A Informationen zur Implementation

Die im Rahmen des Projekts (<http://projekte.annette-gymnasium.de/informatik/parknet/>) entworfene Software wurde in Python 3 implementiert und ist unter <https://github.com/lukashaverbeck/parknet> inklusive des Softwareentwurfs öffentlich zugänglich.

Bei der Entwicklung wurden die im Folgenden aufgelisteten Bibliotheken verwendet.

	Bibliothek	Version	Link
[1]	Adafruit-PCA9685	1.0.1	<a href="https://github.com/adafruit/Adafruit_Python_PCA9685">https://github.com/adafruit/Adafruit_Python_PCA9685</a>
[2]	flask	1.1.1	<a href="https://github.com/pallets/flask">https://github.com/pallets/flask</a>
[3]	matplotlib	3.1.1.	<a href="https://github.com/matplotlib/matplotlib">https://github.com/matplotlib/matplotlib</a>
[4]	netifaces	0.10.9	<a href="https://github.com/al45tair/netifaces">https://github.com/al45tair/netifaces</a>
[5]	numpy	1.16.4	<a href="https://github.com/numpy/numpy">https://github.com/numpy/numpy</a>
[6]	pandas	0.25.1	<a href="https://github.com/pandas-dev/pandas">https://github.com/pandas-dev/pandas</a>
[7]	picamera	1.13	<a href="https://pypi.org/project/picamera/">https://pypi.org/project/picamera/</a>
[8]	Pillow	6.2.0	<a href="https://github.com/python-pillow/Pillow">https://github.com/python-pillow/Pillow</a>
[9]	PyAccessPoint	0.2.5	<a href="https://github.com/Goblenus/pyaccesspoint">https://github.com/Goblenus/pyaccesspoint</a>
[10]	pyzbar	0.1.8	<a href="https://github.com/NaturalHistoryMuseum/pyzbar">https://github.com/NaturalHistoryMuseum/pyzbar</a>
[11]	requests	2.22.0	<a href="https://github.com/psf/requests">https://github.com/psf/requests</a>
[12]	RPi.GPIO	0.7.0	<a href="https://pypi.org/project/RPi.GPIO/">https://pypi.org/project/RPi.GPIO/</a>
[13]	tensorflow	2.0.0	<a href="https://github.com/tensorflow/tensorflow">https://github.com/tensorflow/tensorflow</a>
[14]	wifi	0.6.1	<a href="https://github.com/rockymeza/wifi">https://github.com/rockymeza/wifi</a>
[15]	wireless	0.3.3	<a href="https://github.com/joshvillbrandt/wireless">https://github.com/joshvillbrandt/wireless</a>

[16] Für das Training des neuronalen Netzes für die Steuerung des Fahrzeugs wurde ein öffentlicher Datensatz benutzt, der unter <https://kaggle.com/sameerqayyum/nvidia-self-driving-car-training-set> einsehbar ist.

## B Verwendete Bauteile

[Ada]	Adafruit. <i>PCA9685</i> . URL: <a href="https://cdn-shop.adafruit.com/datasheets/PCA9685.pdf">https://cdn-shop.adafruit.com/datasheets/PCA9685.pdf</a> .
[Foua]	Raspberry Pi Foundation. <i>Picamera</i> . URL: <a href="https://www.raspberrypi.org/documentation/hardware/camera/">https://www.raspberrypi.org/documentation/hardware/camera/</a> .
[Foub]	Raspberry Pi Foundation. <i>Raspberry Pi 4B</i> . URL: <a href="https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-4-Product-Brief.pdf">https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-4-Product-Brief.pdf</a> .
[Pol]	Pololu. <i>DRV8825</i> . URL: <a href="https://www.pololu.com/file/0J590/drv8825.pdf">https://www.pololu.com/file/0J590/drv8825.pdf</a> .
[Spa]	Sparkfun. <i>HC-SR04</i> . URL: <a href="https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf">https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf</a> .
[Ste]	Stepperonline. <i>Nema17</i> . URL: <a href="https://www.omc-stepperonline.com/download/17HS16-2004S1.pdf">https://www.omc-stepperonline.com/download/17HS16-2004S1.pdf</a> .

## C Abbildungsverzeichnis

1	Illustration des Informationsaustauschs für die Bestimmung der Formation . . . . .	8
2	Mittlere Abweichung des Lenkungswinkels im Trainingsverlauf (logarithmisch skaliert) . .	11
3	Beispielhafte Problemsituation überlappender Netzwerke . . . . .	12

## D Quellen- und Literaturverzeichnis

- [BL12] J. Bates and D. Leibling. *Spaced Out: Perspectives on parking policy*. Hrsg. v. *The Royal Automobile Club Foundation for Motoring*. 2012. URL: <https://trid.trb.org/view/1225136>. zuletzt geprüft am: 03.01.2020.
- [Boj16] M. Bojarski et al. *End to end learning for self-driving cars*. Hrsg. v. *IEEE*. 2016. URL: <https://arxiv.org/pdf/1604.07316.pdf>. zuletzt geprüft am: 07.01.2020.
- [CP17] G. Cookson and B. Pishue. *The Impact of Parking Pain in the US, UK and Germany*. Hrsg. v. *INRIX Research*. 2017. URL: <http://www2.inrix.com/research-parking-2017>. zuletzt geprüft am: 22.12.2019.
- [Gas12] T.M. Gasser et al. *Rechtsfolgen zunehmender fahrzeugautomatisierung*. Hrsg. v. *Bundesanstalt für Straßenwesen*. 2012. URL: [https://www.bast.de/BAST\\_2017/DE/Publikationen/Foko/Downloads/2017-2010/2012-11.html?nn=1836812](https://www.bast.de/BAST_2017/DE/Publikationen/Foko/Downloads/2017-2010/2012-11.html?nn=1836812). zuletzt geprüft am: 03.01.2020.
- [INR17] INRIX. *Deutsche Verschwenden 41 Stunden Im Jahr Bei Der Parkplatzsuche*. 2017. URL: <http://inrix.com/press-releases/parking-pain-de/>. zuletzt geprüft am: 03.01.2020.
- [Mau+15] M. Maurer et al. *Autonomes Fahren: technische, rechtliche und gesellschaftliche Aspekte*. 2015. URL: <https://www.oopen.org/download?type=document&docid=1002192>. zuletzt geprüft am: 03.01.2020.
- [OSM16] I. Ostrowski, P. Szulewski, and Maslowski. *Analysis of WiFi Communication (Data Interchange) for Mobile Robot in Industrial Environment*. *Applied Mechanics and Materials*. 2016. URL: [https://www.researchgate.net/publication/289495919\\_Analysis\\_of\\_WiFi\\_Communication\\_Data\\_Interchange\\_for\\_Mobile\\_Robot\\_in\\_Industrial\\_Environment](https://www.researchgate.net/publication/289495919_Analysis_of_WiFi_Communication_Data_Interchange_for_Mobile_Robot_in_Industrial_Environment). zuletzt geprüft am: 07.01.2020.
- [Sch19] M. Schumacher. *c't Projekt 2019 Heft*. 2019. URL: <https://books.google.de/books?id=kkehDwAAQBAJ&lpg=PA13&dq=CYW43455%2085%20mbit%2Fs&hl=de&pg=PA13#v=onepage&q=CYW43455%2085%20mbit/s&f=false>. zuletzt geprüft am: 07.01.2020.
- [Živ17] D. Živadinović. *c't Magazin Ausgabe 1/2017 Digital Gebrandmarkt*. 2017. URL: <https://www.heise.de/select/ct/2017/1/1483620747999859>. zuletzt geprüft am: 07.01.2020.