



Duale Hochschule Baden-Württemberg Mannheim

## Projektbericht Integrationsseminar

### SmartGarage

## Studiengang Wirtschaftsinformatik

### Data Science

Verfasser:

Andreas Edte

Kilian Ebi

Luca Fennuciu

Miguel Sarasa-y-Zimmermann

Matrikelnummern:

6715309

9803923

2512023

4151972

Kurs:

WWI19DSB

Bearbeitungszeitraum:

dd.mm.yyyy – DD.MM.YYYY

# Ehrenwörtliche Erklärung

Wir versichern hiermit, den vorliegenden Projektbericht selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben.

Mannheim, 19.02.2022

Ort, Datum

Team SmartGarage

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>IV</b>
<b>Tabellenverzeichnis</b>	<b>V</b>
<b>Abkürzungsverzeichnis</b>	<b>V</b>
<b>Kurzfassung (Abstract)</b>	<b>VI</b>
<b>1 Motivation</b>	<b>1</b>
<b>2 Grundlagen</b>	<b>2</b>
2.1 Konzeption . . . . .	2
2.2 Hardware . . . . .	3
2.3 Einrichtung . . . . .	3
2.4 Ultraschallsensor . . . . .	3
2.5 RFID . . . . .	4
2.6 Weboberfläche . . . . .	6
<b>3 Kennzeichenerkennung</b>	<b>8</b>
3.1 Datenbeschaffung . . . . .	8
3.2 Modelltraining und Implementierung . . . . .	8
3.3 Geometrische Lösung . . . . .	8
3.4 OCR . . . . .	8
3.5 Validierung . . . . .	9
<b>4 Technische Umsetzung</b>	<b>10</b>
4.1 Implementierung Alexa . . . . .	10
4.2 Schaltlogik . . . . .	10
4.3 Weboberfläche . . . . .	10
<b>5 Risiken und Limitierungen</b>	<b>11</b>
5.1 Wirtschaftliche Aspekte . . . . .	11
5.1.1 Kosten . . . . .	11
<b>6 Zusammenfassung</b>	<b>12</b>
6.1 Fazit . . . . .	12
6.2 Ausblick . . . . .	12

**Anhang**

**Literaturverzeichnis**

**14**

# Abbildungsverzeichnis

# Abkürzungsverzeichnis

SSH      Secure Shell

# Kurzfassung (Abstract)

Hier können Sie die Kurzfassung (engl. Abstract) der Arbeit schreiben. Beachten Sie dabei die Hinweise zum Verfassen der Kurzfassung.

# 1 Motivation

Im wachstumsstarken Markt des **IoT!** (**IoT!**) war der Bereich der Smart Home Technologien in 2021 mit knapp 14 Mrd. Euro das größte Marktsegment.<sup>1</sup> Bis 2026 wird von einer jährlichen Wachstumsrate von 11,6% ausgegangen.<sup>2</sup> Neben vielen bereits gut am Markt bedienten Use-Cases wie die Lichtsteuerung, Bewässerungssteuerung für Zimmerpflanzen, schaltung von Steckdosen oder der Temperaturregelung gibt es allerdings einige noch nicht angebotene, die daher ein großes Potential haben. Unter Anderem zählen wir hierzu die Personen- und Objekterkennung, die mittlerweile auch mit kleinen und preiswerten IoT-Devices möglich ist. Beispielsweise könnten Licht und Musik in einem Raum durch die individuellen Präferenzen der sich darin aufhaltenden Personen gesteuert werden. Ebenfalls möglich wäre eine automatische Türöffnung durch Gesichtserkennung oder eine automatische Öffnung der Garage durch Erkennung des Autos.

Letzteres wurde als Use-Case für dieses Projekt ausgewählt, also eine in Netzwerk eingebundene automatisierte Garagentorsteuerung, die das Kennzeichen heranfahrender Fahrzeuge erkennt und, sofern sie zur Einfahrt berechtigt sind, das Garagentor öffnet.

---

<sup>1</sup>*Internet der Dinge - Weltweit 2022.*

<sup>2</sup>*Smart Home - Weltweit 2022.*



# 2 Grundlagen

## 2.1 Konzeption

Als Ausgangslage dient in diesem Projekt eine handelsübliches elektrisch öffnenbares Garagentor, das mit einer Fernbedienung gesteuert wird. Dabei wird davon ausgegangen, dass es sich um eine 1-Kanal Steuerung handelt, also dass ein einmaliges Drücken eines Knopfes der Fernbedienung das Garagentor öffnet und ein weiteres Betätigen desselben Knopfes das Garagentor wieder schließt. Weiterhin wird davon ausgegangen, dass das Garagentor beim Auftreffen auf ein Hindernis automatisch reversiert. Dies ist sogar, wie durch ein Gerichtsurteil des OLG Frankfurt von 2015 festgestellt, gesetzlich vorgeschrieben. Die Ansteuerung und Regelung des Garagentor-Motors selbst sind also nicht Bestandteil dieser Projektarbeit.

Neben der automatisierten Öffnung durch die Kennzeichenerkennung wurden folgende Features als Projektumfang festgelegt:

- Amazon Alexa oder Google Home Integration, um das Garagentor auch als "Fußgängeründ ferngesteuert öffnen zu können (um Gartengeräte zu entnehmen oder den Postboten ein Paket abstellen zu lassen)
- Logging der Ein- und Ausfahrenden Fahrzeuge, um das Produkt eventuell später auch für kommerzielle Parkhäuser und Tiefgaragen nutzen zu können

Folgende Features wurden als optional festgehalten und deren Implementierung vom Projektverlauf abhängig gemacht:

- Lichtschranke zur Erkennung ob die Garage bereits belegt ist. In diesem Fall soll das Tor nicht geöffnet werden und es einem anderen Fahrzeug, das ebenfalls einfahrtsberechtigt ist, ermöglicht werden VOR der Garage zu parkieren ohne ständig durch das sichtbare Kennzeichen die automatische Öffnung auszulösen.
- Öffnung per Transponder, um das Tor vor Ort und ohne Smartphone öffnen zu können, falls der Akku leer ist oder man anderen Personengruppen (evtl. temporären) Zutritt erteilen möchte

## 2.2 Hardware

Aufgrund der vielseitigen Verwendbarkeit, des geringen Preises, der guten Konnektivität und Kompatibilität wurde sich für ein Raspberry Pi 3+ bzw. 4 entschieden. Diese Modelle sind hervorragend für IoT-Anwendungen geeignet und verfügen mit 4 bzw. 8 GB trotzdem über genug Arbeitsspeicher um einfache Bildverarbeitung durchführen zu können.

Als einfachste Schnittstelle zum Torantrieb wurde der Handsender identifiziert. Hier genügt es zwei Drähte am Ein- und Ausgang des Aktivierungsknopfes anzulöten und diese mit einem Relais zu verbinden. TODO Direkt am Raspi möglich ? Dann kann durch die **GPIO! (GPIO!)-**Pins das Relais geöffnet oder geschlossen werden und so ein Betätigen der Fernbedienung simuliert werden.

## 2.3 Einrichtung

## 2.4 Ultraschallsensor

Ultraschallsensoren finden in der heutigen Zeit in vielen Anwendungsbereichen Verwendung. Beispiele hierfür ist beispielsweise die Werkstoffprüftechnik, medizinische Diagnostik oder auch Näherungsschalter. Ein weiteres bekanntes Beispiel aus dem Alltag ist die Verwendung in Fahrzeugen, bei denen Systeme, die beispielsweise beim Einparken helfen sollen, auf Ultraschallsensoren basieren. Durch Anwendungsfelder wie die Durchhangregelung, die Höhenmessung, die Lagerregelung, den Kollisionsschutz oder auch die Füllstandserfassung sowie die Objekterkennung und Objektzählung ist Ultraschall als Messtechnik in fast jeder Branche einsetzbar.<sup>1</sup> Ultraschall selbst liegt bei Frequenzen über dem Frequenzbereich, den ein Mensch hören kann. Also über 20 kHz bis zu 1 GHz.<sup>2</sup> Erzeugt werden kann Ultraschall über zwei verschiedene Methoden, pneumatisch oder elektrisch beziehungsweise piezoelektrisch oder magnetostruktiv.<sup>3</sup> Da sich Ultraschallwellen über die Luft bewegen spielt die Temperatur bei der Messung eine Rolle und kann die Genauigkeit beeinflussen. Die Schallgeschwindigkeit bei einer Temperatur von 0° Celsius liegt so beispielsweise bei 331,6 m/s während die Schallgeschwindigkeit bei 20° Celsius bereits 343,8 m/s beträgt.

---

<sup>1</sup>ultraschall2.

<sup>2</sup>ultraschall2.

<sup>3</sup>sensoren.

Man kann hierbei also feststellen, dass sich der Schall bei einer wärmeren Temperatur schneller bewegt. Auch einen Einfluss auf die Messgenauigkeit kann neben der Temperatur auch der Luftdruck haben. Bei ansteigendem Luftdruck nimmt die Schallgeschwindigkeit zu. Auch die Zusammensetzung der Luft spielt eine Rolle, also unter anderem der CO<sub>2</sub>-Gehalt der Luft sowie die relative Luftfeuchte. Diese Einflussfaktoren sind bei Entwurf von Ultraschallsensoren zu beachten, um später ein möglich akkurates Messergebnis zu erzielen.<sup>4</sup> Die meistverwendete Art von Ultraschallsensoren sind Ultraschall-Abstandssensoren, die aus einem Sender und Empfänger bestehen, die beide in demselben Gehäuse eingebaut sind. Bei dieser Art von Ultraschallsensoren wird der Sender, periodisch angesteuert, sodass dieser einen Ultraschallimpuls von 100 Mikrosekunden bis zu 1 Millisekunde in einem Frequenzbereich von etwa 40 bis 400 kHz aussendet. Nachdem ein Signal ausgesendet wurde versucht der Empfänger das Echo des Ultraschalls zu erfassen. Dies ist möglich, wenn sich innerhalb der Schallkeule, die vom Sender ausgesendet wird, ein Objekt befindet. Ist dies der Fall, wird der Ultraschallimpuls vom Objekt reflektiert, sodass dieser wieder zurück zum Sender gesendet wird, und dadurch vom Empfänger aufgenommen werden kann.

Für die Abstandsbemessung im Projekt SmartGarage wird der HC SR04 Ultraschallsensor implementiert. Der HC SR04 Ultraschallsensor besteht aus zwei Komponenten die als Trigger bzw. Sender und Echo bzw. Empfänger fungieren. Die Funktionsweise ist die Selbe, wie bereits allgemein zu Ultraschall-Abstandssensoren erläutert wurde. Angesteuert wird der Ultraschallsensor über die GPIO Ports des Raspberry Pis. Die Messung selbst wird mit Hilfe eines Python-Skripts zur Abstandsmessung gesteuert.

Eingesetzt wird die Abstandsbemessung mittels Ultraschallsensors im Projekt für die Messung, ob ein Objekt, also in diesem Anwendungsfall ein Automobil, in der Garage geparkt ist, oder nicht. Daraus lässt sich die Information ableiten, ob die Garage frei oder belegt ist. Diese Information wird über die Weboberfläche ausgegeben.

## 2.5 RFID

RFID, kurz für Radio-Frequenz-Identifikation, ist eine Technologie, die Verwendet wird um Gegenständen oder auch Personen sowie Tieren eine Kennzeichnung zu Vergeben. Zu einem solchen RFID-System gehören zwei grundlegende Komponenten, ein Transponder sowie ein Lesegerät, die mittels Radiowellen untereinander kommunizieren. RFID-Systeme gehören

---

<sup>4</sup>sensoren.

wie der Barcode zu sogenannten Auto-ID-Systemen, die in der Lage sind ein Objekt zu identifizieren. Der Barcode ist das bekannteste Auto-ID-System und lässt sich im Alltag beispielsweise in Supermärkten oder im Einzelhandel aufgedruckt auf Produkte auffinden. Die Radio-Frequenz-Identifikation hat gegenüber dem Barcode als Auto-ID-System jedoch einen Vorteil. Bei der Radio-Frequenz-Identifikation muss nicht beachtet werden, dass die Komponenten richtig ausgerichtet sind. Dies ist bei dem Scan eines Barcodes notwendig. Mittels Radio-Frequenz-Identifikation können Objekte ohne spezielle Ausrichtung erkannt werden. Ebenfalls ist die Erkennung mehrerer Objekte gleichzeitig möglich, auch wenn diese beispielsweise in einer Verpackung sind.<sup>5</sup>

Zur Implementierung eines RFID-Systems ist ein Transponder und ein Lesegerät notwendig. Das Lesegerät ist meist stationär angebracht während der Transponder eine Karte, ein Chip oder auch ein Mikrochip sein kann.<sup>6</sup> Ein bekanntes Beispiel aus der Geschäftswelt sind RFID-Chipkarten, mit denen man beispielsweise Zugang auf ein Betriebsgelände, in ein Büro oder auch dem Firmenparkplatz erlangt. Eine gängige Implementierung ist somit, dass ein Mitarbeiter eine Chipkarte erhält, die den Transponder verkörpert. Will dieser Mitarbeiter beispielsweise auf den Firmenparkplatz mit seinem Auto fahren, muss dieser vor einer Schranke halten, und seine Chipkarte an das Lesegerät, welches stationär vor dieser Schranke angebracht ist, halten. Das Lesegerät liest die Chipkarte mit Hilfe von Radiowellen aus und kann somit identifizieren, dass es sich um einen Mitarbeiter handelt und gewährt Zugriff auf den Firmenparkplatz in Form von Öffnen der Schranke.

Wichtig zu beachten bei Radio-Frequenz-Identifikationssystemen ist Frequenz. Die Wahl dieser hat einen hohen Einfluss auf die Funktion und Sicherheit dieser Systeme. Grund hierfür ist, dass nicht nur RFID-Systeme die Radiofrequenz verwenden, sondern auch wie der Name verbirgt Radiosender sowie andere Funkanlagen. Die gängigen Frequenzen für RFID-Systeme sind daher 120 - 135 kHz für die Low Frequenz, 13,56 MHz für die High Frequenz sowie 868 MHz, 915 MHz, 2,45 GHz und 5,5 GHz für die Ultra-High Frequenzen. Der gängige Standard ist hierbei jedoch 13,56 MHz.<sup>7</sup>

Die Implementierung eines RFID-Systems in der SmartGarage bietet eine weitere Möglichkeit die Garage zu öffnen. Für das Projekt wird daher ein RFID-RC522 Modul an den Raspberry Pi angeschlossen. Dieses dient als Lesegerät, welches später den Transponder auslesen wird. Angesteuert vom Raspberry Pi wird das Modul mittels der GPIO-Ports ähn-

---

<sup>5</sup>rfid2.

<sup>6</sup>rfid.

<sup>7</sup>rfid.

lich wie beispielsweise bereits dem Ultraschallsensors. Das Auslesen wird dann mit Hilfe eines Python-Skripts gesteuert und mit einer Whitelist abgeglichen.

Als Transponder kann im Projekt jede beschreibbare RFID-Karte oder RFID-Chipkarte verwendet werden, solange diese die vorausgesetzte Frequenz von 13,56MHz unterstützt. Im Projekt wird hierfür eine RFID-Karte, die nicht mit dem richtigen Code beschriftet ist, als Beispiel für eine fehlerhafte Identifikation verwendet. Wenn diese an das Lesegerät gehalten wird, wird kein Zugriff gewährt, da der Code auf der Karte nicht mit den Codes, die in der Whitelist festgelegt sind, übereinstimmt. Als Beispiel für eine erfolgreiche Identifizierung wird im Projekt ein RFID-Chip mit dem richtigen Code beschriftet. Wird dieser an das Lesegerät gehalten, wird abgeglichen ob dieser in der Whitelist enthalten ist. Da dieser in der Whitelist enthalten ist, wird Zugriff auf die SmartGarage gewährt.

Bei tatsächlicher Implementierung besteht hierbei neben Chipkarten und RFID-Karten auch die Möglichkeit sich einen NFC-Chip beziehungsweise RFID-Chip mittels Implantat unter die Haut einsetzen zu lassen. Dies bietet den Vorteil, dass man immer Zugriff auf seine Garage hat, ohne einen externen Gegenstand, den man verlieren kann, bei sich haben zu müssen. Während des Projektverlaufes konnten zwei weitere Funktionen festgestellt werden, die sich noch mit vertretbarem zeitlichen Aufwand in das Produkt integrieren ließen.

## 2.6 Weboberfläche

Mit Hilfe einer Weboberfläche lässt sich eine weitere Möglichkeit die SmartGarage zu öffnen implementieren. Der Vorteil dabei ist, dass diese unabhängig vom Endgerät aufgerufen werden kann und lediglich ein mit dem Internet verbundener Browser benötigt wird. Des weiteren lassen sich in einer Weboberfläche auch Informationen, die für den Besitzer der SmartGarage relevant sind, ausgeben. Hierzu gehört beispielsweise die Information, ob sich aktuell ein Fahrzeug in der Garage befindet. Hierzu wird ein Ultraschallsensor implementiert, der diese Information auf der Weboberfläche bereitstellt. Hierzu wird eine einfache wenn-Funktion verwendet, die angibt, dass die Garage belegt ist, falls sich innerhalb eines gewissen Abstands ein Objekt befindet, und dass die Garage frei ist, wenn sich innerhalb dieses Abstands kein Objekt befindet. Ein weiterer wichtiger Nutzen der Weboberfläche für die Bereitstellung von Informationen ist das Anzeigen von Logs. Hierzu wird eine Log-Datei erstellt, in die alle Dienste, mit denen sich die SmartGarage öffnen lässt, einen Log-Eintrag

schreiben, sobald sie diese öffnen. Damit wird Dokumentiert wann und wie die SmartGarage geöffnet wird. Durch die Implementierung dieser Logs in einer Weboberfläche steht diese Information dem Besitzer auch unabhängig vom Standort zur Verfügung.

Für die Realisierung einer Weboberfläche gibt es viele Realisierungsansätze und Möglichkeiten sowie Frameworks die zur Verwendung herangezogen werden können. So bieten sich beispielsweise beliebte Webframeworks wie React, Angular, NextJS, Nuxt, Vue oder viele weitere an. Da die Implementierung im Rahmen des Projekts jedoch auf einem Raspberry Pi stattfindet und Python bereits für viele Anwendungen innerhalb des Projekts verwendet wird, wird sich für das in Python geschriebene Webframework Flask entschieden. Dies hat den Vorteil, dass andere Komponenten, die bereits in Python geschrieben sind, direkt ohne Mehraufwand in die Weboberfläche integriert werden können.

Daher wird im Rahmen des Projekts für die Realisierung einer Weboberfläche eine einfache FlaskApp erstellt, die den üblichen Aufbau hat mit einer `flaskapp.py`, die die Weboberfläche steuert, und auf einzelne Templates, die wiederum in HTML geschrieben werden, zugreift. Hierbei wird in der `flaskapp.py`-Datei die Initialisierung der Weboberfläche sowie das Routing und dem Datentransfer zwischen den Seiten festgelegt sowie auch die Funktion, die die Garage öffnet. Als Templates die als Seite geladen werden, wird ein Template für die Startseite erstellt sowie ein Template für die Seite, auf der die Logs angezeigt werden sollen. Die Hauptseite, die bei Aufruf der Seite geladen wird, enthält einen Button, der mittels der in der `flaskapp.py` dafür festgelegten Funktion die SmartGarage öffnet und einen Eintrag in der Logs-Datei erstellt. Des Weiteren enthält diese Seite die Information ob in der Garage ein Fahrzeug steht, oder nicht. Dies geschieht über die Ultraschallmessung, wie bereits zuvor erläutert. Ebenfalls wird von dieser Seite aus auf die Logs-Seite verlinkt. Die Logs-Seite bezieht die Daten aus der Logs-Datei und gibt diese auf der Weboberfläche aus, wodurch der Besitzer der Garage die geloggten Öffnungen einsehen kann.

# 3 Kennzeichenerkennung

## 3.1 Datenbeschaffung

Die Güte eines Modells hängt stark von der Qualität und Quantität der beim Training verwendeten Datensätze ab. Jedoch sind Datensätze von deutschen Kennzeichen aufgrund der **DSGVO!** (DSGVO!) sehr schwer zu beschaffen. Zudem sind Anwendungen im Bereich Kennzeichenerkennung von großem kommerziellen Nutzen und dementsprechend nicht frei verfügbar. Die Recherche ergab, dass nur ein einziger Datensatz frei verfügbar war und für das Projekt in Frage käme. Es handelte sich um den TODO der TH Ingolstadt. Eine Anfrage beim dort zuständigen Prof. TODO blieb aber leider unbeantwortet, sodass eine andere Lösung gefunden werden musste. Da das Modell die Kernfunktion des Projekts ist, entschied sich das Projektteam dazu, selbst einen kleinen Datensatz zu erstellen. Hierzu wurden 108 Bildaufnahmen von Fahrzeugen in der Nachbarschaft angefertigt. Dabei wurde darauf geachtet, möglichst immer in der gleichen Perspektive zu fotografieren. Da sich die Kamera zur Kennzeichenerkennung nicht mittig im Garagentor selbst befestigen lässt, sondern entweder darüber oder seitlich versetzt montiert wird, wurden alle Aufnahmen von einer leicht seitlich versetzten Position aus getätigt.

## 3.2 Modelltraining und Implementierung

## 3.3 Geometrische Lösung

hier auf jeden noch die quelle

## 3.4 OCR

OCR (Optical Character Recognition) ist das automatische Erkennen von Text innerhalb eines Bildes. Diese Technologie ermöglicht es, den Text in einem Bild oder einem gescannten Dokument in für Menschen verständlichen Text umzuwandeln. Für diesartige Anwendungen

gibt es eine Vielzahl von Anwendungsfällen in der heutigen Zeit. Dazu gehört beispielsweise die Umwandlung Informationen, die ausschließlich in einem Print-Medium zu finden sind, in ein digitales Medium, die einfache Erkennung eines Coupons oder auch das Auslesen eines schriftlich ausgefüllten Formulars mit Hilfe des Computers.<sup>1</sup>

Für die Optische Texterkennung gibt es mittlerweile eine Vielzahl von angebotenen Algorithmen und Services. Darunter beispielsweise Textract, welches von Amazon bereitgestellt und verkauft wird. Allerdings gibt es auch Open-Source Algorithmen dafür wie EasyOCR und Tesseract. Diese ermöglichen es auch ohne einen Aufwand für die Kosten der Texterkennung diese durchzuführen.

## 3.5 Validierung

Nach erfolgreichem Proof-of-Concept musste unter den technisch möglichen Varianten der Verarbeitungspipelines die beste ausgewählt werden. Hierzu wurde ein Skript geschrieben, dass die Pipeline in leicht abgewandelter Form für jedes Bild des gesammelten Datensatzes durchführt und das Ergebnis oder eventuelle Fehler in den einzelnen Bearbeitungsschritten festhält. Damit konnten verschiedene OCR-Verfahren untereinander verglichen werden. Ebenso war es möglich die bereits festgestellten Unterschiede in der Bearbeitungsgeschwindigkeit und Qualität von der Berechnung am Laptop gegenüber der Bearbeitung auf dem RasPi zu quantifizieren. Anhand der festgestellten Metriken konnten schlussendlich auch weiterführende Optimierungen vorgenommen und Fehlerquellen lokalisiert werden.

---

<sup>1</sup>ocr1.



# 4 Technische Umsetzung

## 4.1 Implementierung Alexa

## 4.2 Schaltlogik

## 4.3 Weboberfläche

# 5 Risiken und Limitierungen

- Umlaute
- Schlechte Sicht / Schlechtes Wetter / Beleuchtung
- Verschmutzte Kennzeichen (muss rechtlich aber eh sauber sein)
- Text am Auto / Kennzeichenhalter
- Sicherheitsrisiko Einbruch / Fälschung (ist eh strafbar)
- Genauigkeit

## 5.1 Wirtschaftliche Aspekte

### 5.1.1 Kosten

Geschätzte Entwicklungskosten

Raspberry Pi 3	39,95 <sup>1</sup>
RFID-Modul RC552	5,00 <sup>2</sup>
Ultraschallmodul	8,94 <sup>3</sup>
Amazon Alexa Wifi Modul	13,99 <sup>4</sup>
Summe	

Alle Preise inkl. Versand

Materialkosten je Einheit Raspberry Pi 3 39,95 RFID-Modul RC552 5,00 Ultraschallmodul HC-SR04 4,99

# 6 Zusammenfassung

Dieses Kapitel enthält die Zusammenfassung der Arbeit mit Fazit und Ausblick.

## 6.1 Fazit

super toll 100 Punkte

## 6.2 Ausblick

Das Projekt SmartGarage bietet in vielen Punkten Potential und Ausblick auf weitere Features. Besonders der Ausblick auf die kommerzielle Nutzung des Systems ist vielversprechend. Grund hierfür ist, dass es aktuell noch kein Gesamtpaket kommerziell zu kaufen gibt, das alle Features, die im Projekt implementiert sind, enthält. Auch die kommerzielle Nutzung im Öffentlichen Raum ist möglich. So beispielsweise bei Tiefgaragen sowie Parkhäusern. Hierbei besteht die Möglichkeit das Projekt in einer abgewandten Form kommerziell nutzbar zu machen. Hierbei ist es beispielsweise denkbar, die Weboberfläche so umzugestalten, dass dort Parkplätze bereits vorab am Handy oder Tablet gebucht werden können und der Kunde sein Nummernschild angibt. Ist der Kunde dann vor Ort kann er aufgrund der Nummernschilderkennung einfach und ohne weiteren Aufwand in das Parkhaus oder die Tiefgarage fahren und hat bereits einen reservierten Parkplatz. Dies ermöglicht potentiellen Betreibern von diesem System, die Parkdauer des Kunden direkt zu berechnen, da das Loggen der Ein- und Ausfahrten in diesem Anwendungsbeispiel diese Information bereits mitliefert.

Einen weiteren Nutzen für die Öffentliche Sicherheit kann das Projekt mit einer Anbindung an die Polizei bieten. Durch die Nummernschilderkennung wird bei Einfahrt das Nummernschild erkannt und es ist im System ersichtlich, welches Auto in welcher Garage geparkt hat. Dies macht es wie im Anwendungsbeispiel der Tiefgaragen und Parkhäuser möglich, dass die Polizei direkt nachverfolgen kann, wo ein Nummernschild zuletzt geparkt hat. Wird so beispielsweise ein Verkehrsunfall mit Fahrerflucht gemeldet, bei dem der flüchtige Fahrer danach zu einem Kaufhaus fährt, welches über ein Parkhaus mit einem solchen System

verfügt, kann die Polizei den flüchtigen Fahrer nach Einfahrt in das Parkhaus direkt lokalisieren. Dies ist auch im Falle von Privatgaragen möglich, sollten diese auch die Information über Ein- und Ausfahrt mit der Polizei teilen. Im Falle dessen bietet das System hier auch den Vorteil, dass die Zustellung von Strafzetteln vereinfacht wird, da hier die Strafzettel immer an die richtige Adresse gesendet werden.

Potential zur Verbesserung weist das Projekt im Punkt Sicherheit auf. Hierbei ist die Rede von sowohl Sicherheit im Sinne von Schutz vor einem Einbruch als auch Sicherheit im Sinne von Unfallvermeidung. Die Vermeidung von Unfällen spielt vor allem eine Rolle, falls es sich um ein Garagentor handelt, welches sich nach außen öffnet. Hierbei ist es sinnvoll, auch eine Abstandsmessung außerhalb durchzuführen, um das Öffnen der Garage zu blockieren, falls sich etwas vor dem Tor befindet. Hierbei kann es sonst zu Kollisionen zwischen dem Tor der Garage und beispielsweise einem Auto kommen. Kommt es jedoch vor, dass vor der Garage Passanten laufen könnten, könnte es hierbei auch zu schwereren Unfällen kommen, im Falle dass das Garagentor mit einem vorbeifahrenden Fahrrad kollidiert. Die Sicherheit vor einem Einbruch lässt sich auch durch verschiedene Methoden noch optimieren. Eine Möglichkeit ist hierbei beispielsweise die Implementierung einer Sperre, die das Garagentor mittels Nummernschilderkennung nicht öffnet, sollte sich ein Auto bereits in der Garage befinden. Eine derartige Implementierung schützt nicht nur vor eventuell gefälschten Nummernschildern, sondern auch vor False-Positives im System, wodurch die Garage fehlerhafterweise bei der Einfahrt eines Kennzeichens, welches nicht in der Whitelist steht, geöffnet werden kann.

# Literaturverzeichnis

*Internet der Dinge - Weltweit* (2022). URL: <https://de.statista.com/outlook/tmo/internet-der-dinge/weltweit> (besucht am 20.01.2022).

*Smart Home - Weltweit* (2022). URL: <https://de.statista.com/outlook/dmo/smart-home/weltweit> (besucht am 20.01.2022).