

Projektdokumentation

im Studiengang

AIN/ITP

## **Convolutional Neural Networks**

Qualitätsanalyse und Vorhersage von Komponenten des  
Weinschorle-Automaten

Referent : Prof. Dr. Christoph Reich

Koreferent : Matthias Lermer

Vorgelegt am : 25. Juli 2018

Vorgelegt von : Sugirthan Arulsothy  
Marius Hirt  
Oliver Mattes  
Yasemin Türkmen



## **Abstract**

Im Herbst 2017 wurde als Einsendung für den Wettbewerb „CLOUD-MEISTER 2017“ der Weinschorle-Automat WISH 4.0 entwickelt. Im Rahmen des Semesterprojektes des Sommersemesters 2018 sollte dieser weiterentwickelt werden. Diese Arbeit dokumentiert die an das Weiterentwicklungsprojekt gestellten Anforderungen, sowie Umsetzung, Probleme und Ergebnisse von diesem.

Dabei haben wir uns mit drei funktionalen Anforderungen auseinandergesetzt, von denen wir sowohl die Klassifizierung verschiedener Glasformen, als auch das Feststellen von Verschmutzung des Drehellers mithilfe von Convolutional Neural Network (CNN) umsetzen konnten. Die Volumenberechnung von Gläsern wurde aufgrund von technischen Grenzen der Kinect nach ersten Tests nicht weiterverfolgt.



## Inhaltsverzeichnis

Abstract . . . . .	I
Inhaltsverzeichnis . . . . .	IV
Abbildungsverzeichnis . . . . .	V
Tabellenverzeichnis . . . . .	VII
Abkürzungsverzeichnis . . . . .	IX
1 Einleitung . . . . .	1
2 Gläser . . . . .	3
2.1 Anforderungen und Ziele . . . . .	3
2.2 Ausgangssituation . . . . .	3
2.3 Vorgehen . . . . .	4
2.3.1 Trainingsdaten sammeln . . . . .	4
2.3.2 Trainingsdaten und Script anpassen . . . . .	4
2.3.3 CNN-Architektur . . . . .	6
3 Volumenberechnung mit der Kinect . . . . .	7
3.1 Anforderungen und Ziele . . . . .	7
3.2 Vorgehen . . . . .	7
3.2.1 Limitationen der Kinect . . . . .	7
3.2.2 Vogelperspektive . . . . .	7
3.2.3 Frontalperspektive . . . . .	9
3.3 Fazit zur Kinect . . . . .	9

4 Verschmutzung des Drehtellers . . . . .	11
4.1 Anforderungen und Ziele . . . . .	11
4.2 Vorgehen . . . . .	11
4.2.1 Positionierung der Kamera . . . . .	11
4.2.2 Trainingsdaten sammeln . . . . .	12
4.2.3 CNN-Architektur . . . . .	12
5 Fazit . . . . .	15
5.1 Ausblick . . . . .	15
Eidesstattliche Erklärung . . . . .	17

## Abbildungsverzeichnis

Abbildung 1: Ursprünglich bekannte Gläser . . . . .	3
Abbildung 2: Position der Kamera . . . . .	3
Abbildung 3: Neue Gläser . . . . .	4
Abbildung 4: Foto vor (a) und nach (b) dem Zuschneiden . . . . .	5
Abbildung 5: Struktur des CNN zur Glaserkennung . . . . .	6
Abbildung 6: Platzierung der Kinect für Vogelperspektive . . . . .	8
Abbildung 7: Kinect Aufnahme aus Vogelperspektive, RGB-Modus links (a), Tiefen-Modus rechts (b) . . . . .	9
Abbildung 8: Platzierung der Kinect für Frontalpositionierung . . . . .	9
Abbildung 9: Positionierung der Kamera zur Schmutzerkennung . . . . .	11
Abbildung 10: Auszug aus den zugeschnittenen Trainingsdaten zur Verschmut- zungserkennung (a) sauber, (b) verschmutzt und (c) besetzt . . . . .	12
Abbildung 11: Struktur des CNN zur Schmutzerkennung . . . . .	13



## Tabellenverzeichnis

Tabelle 1: Umsetzung der Industrie 4.0 Konzepte im WISH 4.0 . . . . .	2
-----------------------------------------------------------------------	---



## **Abkürzungsverzeichnis**

**CNN** Convolutional Neural Network



## 1 Einleitung

Im Rahmen des Wettbewerbes „CLOUD-MEISTER 2017: DER WAGO - IDEENWETTBEWERB“ hat ein Team von Studierenden um Herrn Reich den Weinschorle-Automaten „WISH 4.0“ entwickelt. Am Smartphone, Tablet oder Computer lässt sich dabei das gewünschte Mischverhältnis des Schorles per Webseite angeben, das eingestellte Glas wird per Webcam bemessen, das Volumen bestimmt und anschließend über mehrere Ventile das Schorle gemischt.

Außerdem kann der Automat auch Barcodes erkennen, um früher festgelegte Mischungen wieder zu erkennen. Im Zentrum der Umsetzung standen dabei die in Tabelle 1 dargestellten Industrie 4.0 Konzepte.

Tabelle 1: Umsetzung der Industrie 4.0 Konzepte im WISH 4.0

Industrie 4.0 Konzept	WISH 4.0 Umsetzung
Individualisierte Produktion	Individuelles Glas, Mischverhältnis, Füllmenge und Glas-Label
Prozessoptimierung in der Cloud	Nutzungsabhängige Auswahl des Befüllers
Zustandsüberwachung (Condition Monitoring) in der Cloud	Parameter der Befüller wird überwacht
Optimierte Wartung	Je nach Nutzung wird der Bedarf einer Wartung bestimmt
Cloud Steuerung	Meta-Steuerbefehle (z.B. fülle Glas) werden von der Cloud an die Lokale Steuerung geschickt. Lokale Steuerung (WAGO-Controller) führt die Geräte-Befehle aus.
Visual Quality Management	Visuelle Erkennung von verschiedenen Gläsern in der Cloud
Smart Logistics	Erkennung des Barcodes, Kupplung zu Glas, und Speicherung für weitere Verarbeitung
Supply Chain Compliance	Einhaltung vordefinierter Compliance Vorgaben bezüglich des zu druckenden Labels

## 2 Gläser

### 2.1 Anforderungen und Ziele

Der Weinschorle-Automat soll eine Vielfalt verschiedener Glasformen erkennen um vor dem Befüllen dem Glas ein vordefiniertes Volumen zuordnen können. Ziel dieser Aufgabe ist das Vertraut machen mit Python an sich und insbesondere mit dem Skript, welches bereits in der Cloud für die Zuordnung zuständig ist.

### 2.2 Ausgangssituation

Der Automat kann zu Beginn des Projekts bereits zwischen zwei unterschiedlichen Glastypen unterschieden (siehe Abbildung 1). Dazu nimmt eine vertikal in der Fotokammer angebrachte Kamera (siehe Abbildung 2) ein Bild des eingestellten Glases auf, welches zunächst zur Performance-Verbesserung auf ein kleineres Format zugeschnitten wird. Dieses Bild wird als Eingabe für das CNN übergeben. Als Ausgabe wird das Glas im Automaten mit einer gewissen Wahrscheinlichkeit einem der beiden bekannten Glasformen zugeordnet. Basierend auf diesem Ergebnis wird die Befüll-Routine durchgeführt.



Abbildung 1: Ursprünglich bekannte Gläser



Abbildung 2: Position der Kamera

## 2.3 Vorgehen

### 2.3.1 Trainingsdaten sammeln



Abbildung 3: Neue Gläser

Zunächst wurde die Kamera von dem Weinschorle-Automat getrennt und direkt an einen Laptop angeschlossen. Über diesen haben wir automatisch eine Serie von 100 Bildern für insgesamt vier neue Glasformen (siehe Abbildung 3) aufgenommen. Zwischen den Aufnahmen war jeweils eine Verzögerung von drei Sekunden, so dass wir das Glas zwischen den Aufnahmen drehen konnten. Dadurch haben wir ein diverses Trainingsdataset erzeugt, was die Trainingseffektivität des CNN verbessert.

### 2.3.2 Trainingsdaten und Script anpassen

Bevor wir das CNN mit diesen Daten trainieren konnten mussten wir die Bilder zuschneiden (siehe Abbildung 4). Dies geschah mit dem selben Python-Script, welches bereits in der Routine des Weinschorle-Automaten verwendet wird. Weiterhin mussten wir das Trainingsscript dahingehend anpassen, dass wir die neuen Trainingsdaten als solche deklarieren und jedem Trainingsset ein entsprechendes Label zuordnen.

Damit hat sich die Anzahl der klassifizierbaren Gläser auf insgesamt sechs erhöht. Deshalb mussten wir im Predict-Script die Dimension des Output-Tensors an die neue Anzahl der Klassen entsprechend anpassen.



Abbildung 4: Foto vor (a) und nach (b) dem Zuschneiden

### 2.3.3 CNN-Architektur

Als Eingabe für das CNN verwenden wir ein 3 Channel 128x128 Bild. Für die convolution Operationen setzen wir drei convolutional layer ein, wobei die ersten beiden aus 32 und die dritte layer aus 64 Filtern besteht. Dabei werden gängige 3\*3 Filter verwendet.

Die tatsächliche Erkennung der Gläser übernehmen zwei FC-layer, die erste mit 128 Neuronen, die zweite mit 6 an die Anzahl der Gläser angepasst. Daraus ergibt sich die in Abbildung 5 gezeigte Struktur des CNN

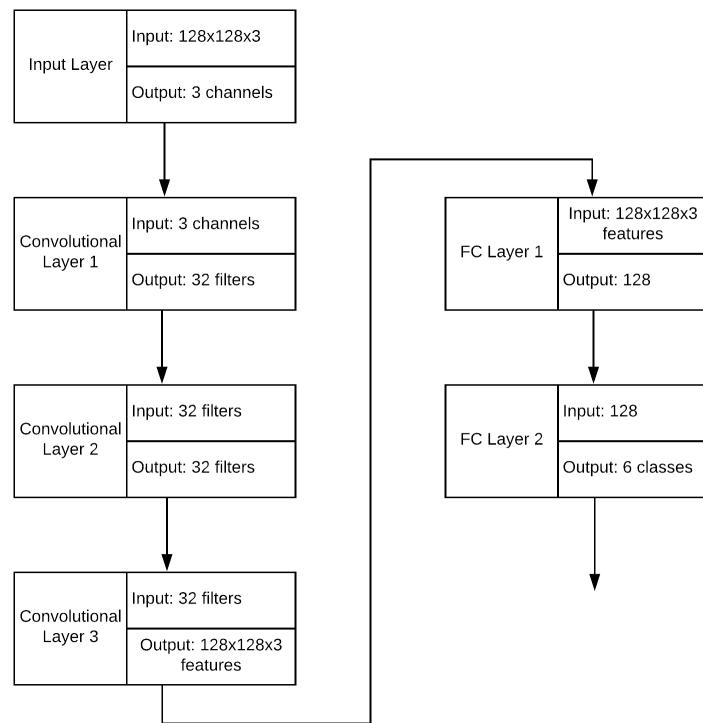


Abbildung 5: Struktur des CNN zur Gläserkennung

## 3 Volumenberechnung mit der Kinect

### 3.1 Anforderungen und Ziele

Um sich nicht mehr auf die Klassifizierung verschiedener Vordefinierter Gläsertypen verlassen zu müssen, soll der Weinschorle-Automat in der Lage sein das Volumen eines Glases mithilfe des Tiefensensors einer Microsoft Kinect zu berechnen.

### 3.2 Vorgehen

#### 3.2.1 Limitationen der Kinect

Bevor wir die Kinect in den Weinschorle-Automaten einbezogen haben, testeten wir deren Möglichkeiten. Nach der Installation von notwendiger Treiber-Software begannen wir mit den Verschiedenen Modi der Kinect zu experimentieren.

Nach ersten Tests und Recherchen mussten wir feststellen, dass der Tiefensensor der Kinect bei Objekten näher als ca. 50cm keine zuverlässigen Werte liefern kann. Diese Tatsache schränkte die Positionierungsmöglichkeiten am Weinschorle-Automaten deutlich ein. Übrig blieben die beiden Möglichkeiten:

1. Vogelperspektive
2. Frontalperspektive

#### 3.2.2 Vogelperspektive

Unsere erste Idee war es die Gläser von direkt über ihnen aufzunehmen. Dazu müssten wir allerdings einen Überbau konstruieren, um die Kinect mindestens 50cm über den Glasrändern anbringen zu können. (Abbildung 6) Vorteile dieser Positionierung sind zum einen die einfachere Messung des Volumens, da wir nur zwei Messpunkte im selben Bild benötigen. Einen für die Entfernung des Randes und einen für die Entfernung des Glasbodens zur Kinect. Daraus lässt sich dann einfach die Füllhöhe als Differenz der beiden Werte berechnen. Mithilfe des Radius könnten wir das Volumen des Glases abschätzen.

Zum andern könnten wir bei dieser Positionierung die RGB-Kamera der Kinect einsetzen, um beispielsweise die Sauberkeit des Drehtellers zu überprüfen (siehe Verschmutzung des Drehtellers).



Abbildung 6: Platzierung der Kinect für Vogelperspektive

Ein Nachteil wäre der zusätzliche Aufwand einen Überbau zu konstruieren, an dem wir die Kinect befestigen können. Außerdem muss bei diesem Ansatz das Glas zwischen den beiden Trichtern des Weinschorle-Automaten vermessen werden (siehe Abbildung 7a). Dies würde uns dazu zwingen den Drehteller nach der Vermessung gegen den Uhrzeigersinn zu drehen. Da zu diesem Zeitpunkt bereits ein neues Glas auf dem Drehteller stehen könnte, würde dessen Füllroutine behindert.

Zum Scheitern dieses Ansatzes führte schließlich die Untauglichkeit der Kinect für die Messung des Glasrandes, wie man in Abbildung 7b erkennen kann. Im besten Fall sollte man an der Stelle des Glasrandes einen dunkel-grauen Ring erkennen können. Jedoch nimmt die Kinect nur undefinierte Werte auf (repräsentiert durch weiß im Bild) Die Kinect gibt Tiefenwerte von 0 (schwarz) bis 2048(weiß) für jeden Pixel in ihrem Tiefenbild an.

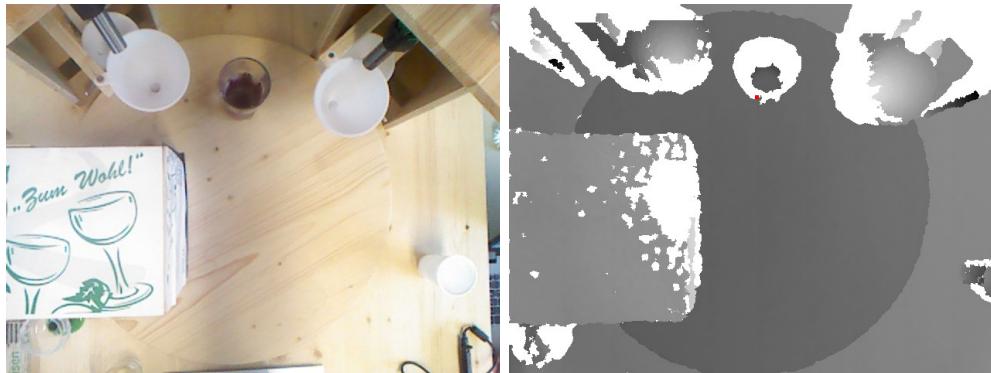


Abbildung 7: Kinect Aufnahme aus Vogelperspektive, RGB-Modus links (a), Tiefen-Modus rechts (b)

### 3.2.3 Frontalperspektive

Alternativ bliebe noch die Möglichkeit die Gläser frontal aufzunehmen, während sie unter dem ersten Trichter auf das Befüllen warten (siehe Abbildung 8a). Dazu positionieren wir die Kinect in der vorderen rechten Ecke des Tisches (siehe Abbildung 8b). Aus dieser Positionierung ergeben sich folgende Vorteile:

- Kein Überbau benötigt
- Keine Rotation des Drehtellers gegen den Uhrzeigersinn

Die Untauglichkeit der Kinect zuverlässige Werte zu liefern führt auch hier, ähnlich wie bei der Vogelperspektive, zum Scheitern des Ansatzes.



Abbildung 8: Platzierung der Kinect für Frontalpositionierung

## 3.3 Fazit zur Kinect

Die Kinect eignet sich nicht für Distanzen unter 50cm Entfernung, was schwer mit den Dimensionen des Weinschorle-Automaten zu vereinbaren ist. (z.B. mit einem Überbau).

Weiterhin reicht die Auflösung der Tiefenkamera nicht aus um schmale Flächen wie einen Glas- oder Becherrand zu erfassen, was zum Scheitern unserer ersten Idee führte.

Zuletzt scheitert diese Idee an der Tatsache, dass die Tiefenkamera der Kinect durch Projektionen von Infrarotlicht funktioniert, wobei aus den Reflexionen der Objekte ein Tiefenbild generiert wird. Durch fehlende Fähigkeit von Glas Licht zu reflektieren, ist es nahezu unmöglich dieses zu vermessen.

Zusammenfassend genügen die Fähigkeiten der Kinect nicht den Anforderungen des Projektes.

## 4 Verschmutzung des Drehtellers

### 4.1 Anforderungen und Ziele

In Regelmäßigen Abständen soll der Verschmutzungsgrad des Drehtellers von einer Kamera erfasst und ausgewertet werden. Dabei sollen alle anstehenden und laufenden Befüll-Routinen angehalten werden, bis eine entsprechende Säuberung durchgeführt wurde.

### 4.2 Vorgehen

Um mögliche Komplikationen zu minimieren führen wir die Schmutzerkennung direkt vor jeder Befüll-Routine durch. Dabei ist die Kamera auf die Endposition der Gläser ausgerichtet.

Grund dafür ist die erhöhte Wahrscheinlichkeit, dass man den Inhalt des Glases beim Entnehmen verschüttet, weshalb die größte Verschmutzungsdichte eben dort zu erwarten ist.

Vorteil davon ist, dass wir ebenfalls im Blick haben, ob das fertig befüllte Glas der vorherigen Routine beim Start einer neuen eventuell noch nicht entnommen wurde. Dies hätte zur Folge, dass ein volles Glas mit der Fotokammer kollidiert, was natürlich unerwünscht ist.

#### 4.2.1 Positionierung der Kamera

Um den Endbereich der Routine (Abbildung 9, rechts), optimal zu erfassen bringen wir die Kamera an der Seite der Fotokammer an (Abbildung 9, links). Durch diese Positionierung haben wir alle relevanten Bereiche im Blick, während wir den Hintergrund minimieren.



#### 4.2.2 Trainingsdaten sammeln

Für diesen Anwendungsfall definieren wir 3 Klassen:

- Sauber
- Verschmutzt
- Besetzt

Dazu haben wir zunächst für 6 unterschiedliche Gläser jeweils 50 Bilder aufgenommen, auf welchen die Gläser an der erwarteten Endposition stehen. Diese klassifizieren wir als „besetzt“ (Abbildung 10c).

Als zweites haben wir insgesamt 50 Fotos des sauberen Drehtellers in verschiedenen Positionen aufgenommen und diese als „sauber“ klassifiziert (Abbildung 10a).

Zum Schluss haben wir mit Weiß-, Rosé-, und Rotwein jeweils 50 Bilder aufgenommen, indem wir ein Glas befüllten und den Inhalt beim Aufnehmen verschütteten, wie es auch in Anwendung sehr wahrscheinlich passieren wird. Dieses Set klassifizieren wir als „verschmutzt“ (Abbildung 10b, Verschmutzung oben zu erkennen).



Abbildung 10: Auszug aus den zugeschnittenen Trainingsdaten zur Verschmutzungs-erkennung (a) sauber, (b) verschmutzt und (c) besetzt

#### 4.2.3 CNN-Architektur

Identisch zur Glaserkennung wird als Eingabe für das CNN ein 3 Channel 128x128 Bild verwendet. Für die convolution Operationen setzen wir jedoch vier convolutional layer ein, um aus den schwerer zu unterscheidenden Bildern detailliertere Features herauszufiltern. Dabei bestehen die ersten beiden layer aus 32, die anderen beiden layer aus 64 3\*3 Filtern.

Zur Erkennung sind gleich wie bei der Glaserkennung zwei FC-layer, die erste mit 128 Neuronen, die zweite mit 3 im Einsatz. Daraus ergibt sich die in Abbildung 11 gezeigte Struktur des CNN

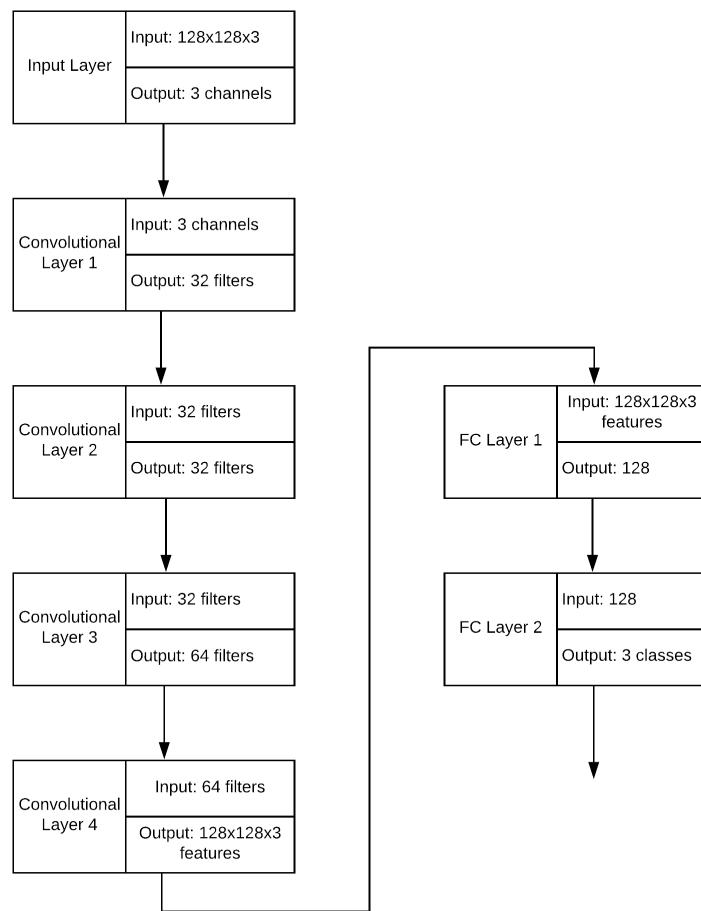


Abbildung 11: Struktur des CNN zur Schmutzerkennung



## 5 Fazit

Das Projekt ist abgeschlossen, ist aufgrund der nicht implementierten Volumenberechnung als funktionale Anforderung als kritisch einzustufen.

Bezogen auf unsere Planung waren unsere Arbeitspakete unproblematisch, da diese unabhängig voneinander waren, weshalb unser kritischer Pfad wesentlich kürzer als die vorgesehene Projektdauer war. Diese haben wir jedoch voll ausnutzen müssen, da wir zum parallel an verschiedenen Anforderungen nicht genug Teilnehmer waren.

### 5.1 Ausblick

Angesichts der in unserem Projekt nicht erfüllten funktionalen Anforderung, die Volumenberechnung der Gläser, wäre eine Weiterentwicklung des Automaten in dieser Richtung durchaus denkbar. Da sich die Kinect aus oben genannten Bedingungen dazu jedoch nicht eignet, müsste man sich auf diesem Gebiet nach passenderen Technologien umsehen.

Weiterhin könnte man sich mit der Analyse weiterer Komponenten beschäftigen, wie zum Beispiel mit dem Füllstand der Silos. Dazu könnte man die Befülltürme für einen Überbau vorbereiten (Obere Plattformen aktuell noch lo-se). Dies würde neue Möglichkeiten zur Analyse des Zustandes vom Automaten bieten, man könnte zum Beispiel den kompletten Drehteller überwachen, und nicht nur die Entnahmeposition der Gläser, wie es aktuell implementiert ist.

Eine weitere praktische Ergänzung wäre ein direkt im Automaten integrierter Label-Drucker, welcher nach Abschluss einer Bestellung in der Webanwendung direkt das zugehörige Label bereitstellt.

Auf Software-Ebene wäre wichtig das Frontend (die Webanwendung) an die neue Routine anzupassen. Mit der Erkennung des Verschmutzungsgrades haben wir einen neuen möglichen Zustand definiert, der vom Automaten eingegenommen werden kann, welcher auf der Status-Seite der Webanwendung jedoch noch nicht berücksichtigt wird. Die gilt auch für eventuelle andere Weiterentwicklungen, wie bereits oben angemerkt.



## **Eidesstattliche Erklärung**

Ich versichere, dass ich die vorstehende Arbeit selbständig verfasst und hierzu keine anderen als die angegebenen Hilfsmittel verwendet habe. Alle Stellen der Arbeit die wörtlich oder sinngemäß aus fremden Quellen entnommen wurden, sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt oder an anderer Stelle veröffentlicht.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben kann.

---

Furtwangen, den 25. Juli 2018 Sugirthan Arulsothy

---

Furtwangen, den 25. Juli 2018 Marius Hirt

---

Furtwangen, den 25. Juli 2018 Oliver Mattes

---

Furtwangen, den 25. Juli 2018 Yasemin Türkmen