HAW Hamburg

# Analyse dynamischer Szenen mit LSTM

Patrick Nagorski
patrick.nagorski@haw-hamburg.de
Department Informatik
Hochschule für Angewandte Wissenschaften Hamburg
Berliner Tor 7
20099 Hamburg

31. August 2017

# Inhaltsverzeichnis

# 1 Einleitung

## 1.1 Motivation

Jährlich passieren unzählige Verkehrsunfälle auf den deutschen Straßen. Der Abbildung 1 kann man entnehmen, dass sich seit 1996 die Anzahl der Unfälle, bei denen Menschen verletzt oder getötet wurden zwar reduziert hat, jedoch immer noch sehr groß ist. Dies ist zum Beispiel ein Grund warum Technologien wie die Fußgängererkennung durch Kameras wichtig sind.

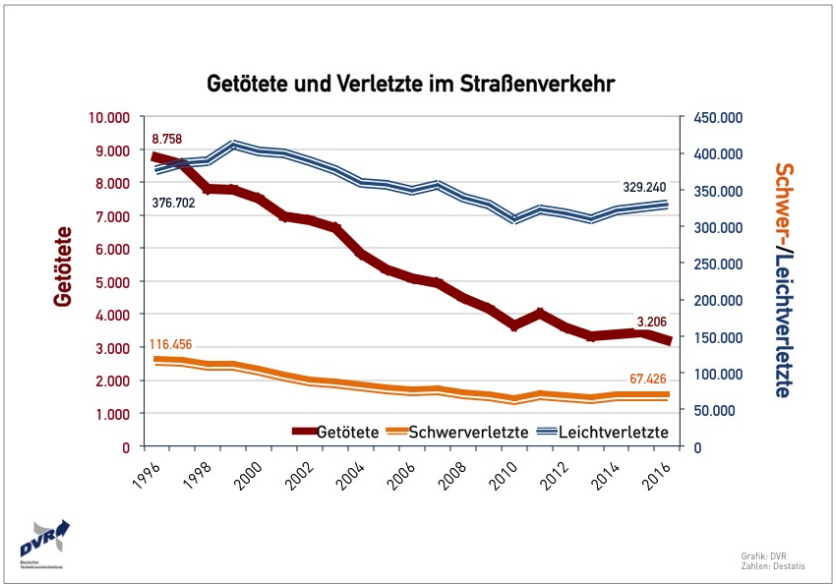


Abbildung 1: Anzahl der getöteten und verletzten Menschen im deutschen Straßenverkehr [10].

Eine weitere Anwendungsmöglichkeit die Erkennung von Kindern, die auf den Straßen spielen. Beispielweise rollt ein Ball auf die Straße und das Kind rennt diesem hinterher, um diesen wieder einzufangen, ohne auf den Straßenverkehr zu achten. Möglicherweise werden sie dann von einem Auto erfasst, welches nicht rechtzeitig reagieren und abbremsen kann. Daher könnte eine Verwendung von entsprechenden Erkennungssystemen in Fahrzeugen Unfälle verhindern. Dafür kann die Analyse dynamischer Szenen mit LSTM angewendet werden.

# 2 Grundlagen

## 2.1 Machine Learning

Beim Machine Learning geht es darum, dass neues Wissen durch ein künstliches System erworben wird. Der Computer generiert dabei selbstständig Wissen aus Erfahrung und versucht dabei eigenständig Lösungen für unbekannte Probleme zu finden. Dazu werden viele Beispiele analysiert und bestimmte Muster und Gesetzmäßigkeiten erkannt. Das Ziel dabei ist Zusammenhänge zwischen den Eingaben und Ausgaben zu erkennen.
Bei den algorithmischen Ansätzen unterscheidet man zwischen überwachtem Lernen (engl. supervised learning) und unüberwachtem Lernen (engl. unsupervised learning).

**Überwachtes Lernen:**
Beim überwachten Lernen wird ein Modell anhand von gegebenen Ein- und Ausgabe-Paaren erlernt. Dies wird durch einen "Lehrer" unterstützt, der zu einer Eingabe den korrekten Ausgabenwert bereithält. Das Lernen wird in zwei Schritten unterteilt. Im ersten Schritt wird ein Modell aus den Trainingsdaten gelernt. Im zweiten Schritt wird das Modell durch bisher nicht genutzte Daten getestet, um die Vorhersage-Genauigkeit zu testen.

**Unüberwachtes Lernen:**
Im Gegensatz zum überwachten Lernen gibt es beim unüberwachten Lernen keinen "Lehrer", der zu einer Eingabe den Ausgabewert kennt. Dabei erzeugt der Algorithmus für eine gegebene Menge von Eingaben ein Modell, das die Eingaben beschreibt und Vorhersagen ermöglicht. Dies wird durch sogenannte Clustering-Verfahren realisiert. Beim Clustering-Verfahren werden die Daten in mehrere Kategorien einteilt, die sich durch charakteristische Muster voneinander unterscheiden.

## 2.2 Künstliche neuronale Netze

Ein künstliches neuronales Netz besteht aus verschiedenen Schichten, welche unterschiedliche Aufgaben haben. Zuerst muss das künstliche neuronale Netz jedoch trainiert werden. Das bedeutet, dass das Netz mit Daten "gefüttert" wird, um diese anschließend klassifizieren zu können. In der Eingabeschicht werden ungelabelte Bilder in das trainierte Netz eingegeben. In der versteck-

ten Schicht werden die Eingangsdaten verarbeitet. Das heißt, die Gewichtung jedes einzelnen Neurons wird so angepasst, dass die Ausgabe möglichst genau dem bekannten Ergebnis entspricht. Anschließend werden in der Ausgabeschicht die Ergebnisse ausgegeben. Im Feedforward-Netz sind die Ausgangssignale nur von den Eingangssignalen abhängig. Dies ist in Abbildung 2 zu sehen. Außerdem werden die Neuronenausgaben nur in Verarbeitungsrichtung weitergeleitet und können nicht durch eine rekurrente Kante zurückgeführt werden.

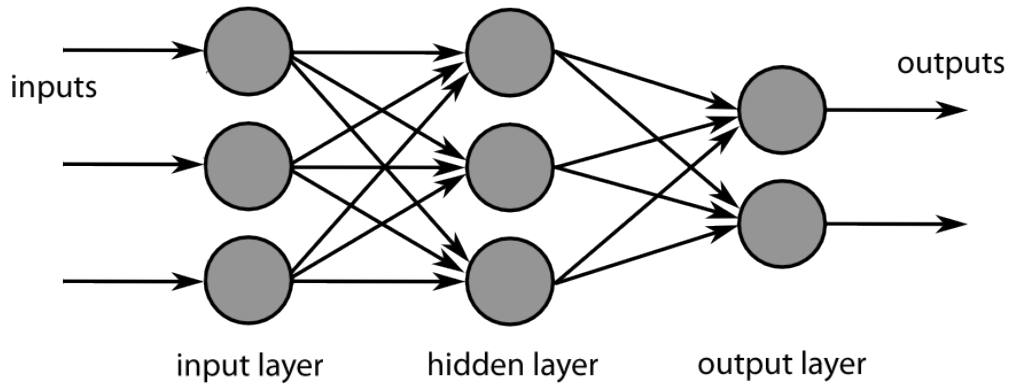


Abbildung 2: Feedforward-Netz [11].

Allerdings gibt es auch rekurrente neuronales Netze 3. Rekurrente Netze sind dadurch gekennzeichnet, dass Rückkopplungen von Neuronen einer Schicht zu anderen Neuronen derselben oder einer vorangegangenen Schicht existieren. Dies führt dazu, dass die Ausgangssignale sowohl von den Eingangssignalen, als auch von der zeitlichen Vorgeschichte abhängig sind. Rekurrente Netze werden zum Beispiel angewendet, wenn eine Prognose für die Zukunft bestimmt werden soll.

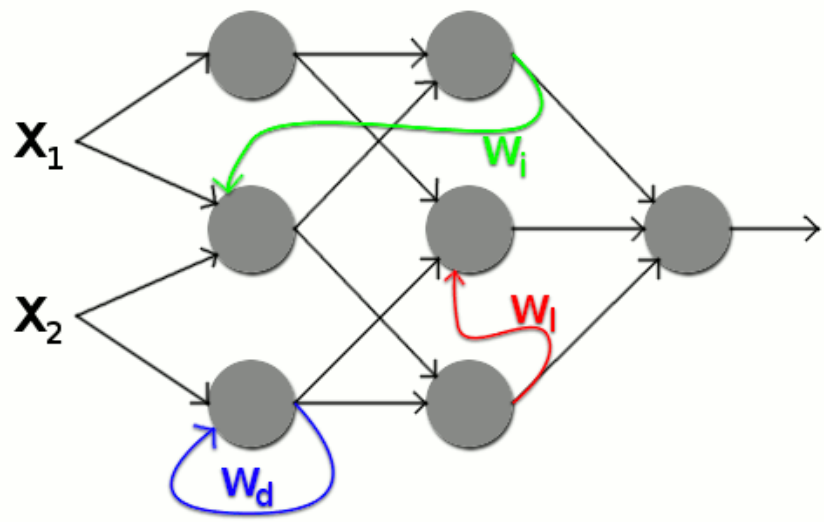


Abbildung 3: Rekurrentes neuronales Netz [12].

Das künstliches neuronale Netz besteht aus vielen Neuronen, die miteinander verbunden sind. Den Aufbau eines Neurons kann der Abbildung 4 entnommen werden.

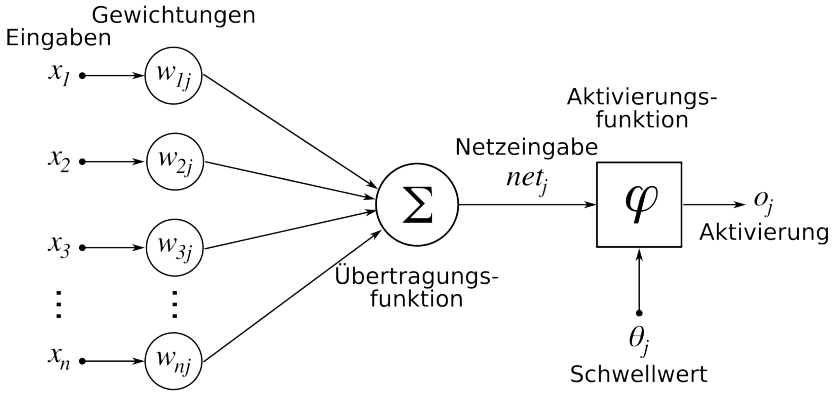


Abbildung 4: Aufbau eines künstlichen Neurons [13].

Die Gewichtung bestimmt den Grad des Einflusses für die Berechnung der späteren Aktivierung. Jeder Eingangswert hat eine Gewichtung. In der Übertragungsfunktion wird jeder einzelne Eingangswert mit dessen Gewichtung multipliziert. Anschließend werden alle Werte miteinander addiert. Auf das

Ergebnis wird anschließend eine Aktivierungsfunktion angewendet, um zum Schluss den Ausgangswert auszugeben. Das Addieren eines Schwellenwerts zur Netzeingabe verschiebt die gewichteten Eingaben. Mit der Schwellenwertfunktion als Aktivierungsfunktion wird das Neuron aktiviert, wenn der Schwellenwert überschritten ist.

## 2.3 LSTM-Netze

Long Short-Term Memory (LSTM) Netze sind eine besondere Art von rekurrenten neuronalen Netzen. Der wesentliche Unterschied zu normalen rekurrenten neuronalen Netzen ist, dass LSTM sowohl kurze, als auch lange Zeitabhängigkeiten verarbeiten kann. In der Abbildung 5 befindet sich auf der linken Seite ein Neuron eines üblichen rekurrenten Netzes. Auf der rechten Seite ein LSTM-Neuron. LSTM-Neuronen bestehen aus Speicherzellen, in die Informationen geschrieben und aus denen Informationen gelesen werden können. Außerdem gibt es zusätzliche Gates (Input-Gate, Output-Gate und Forget-Gate), die über die Zelle gesteuert werden. Die Gates besitzen, wie in den normalen rekurrenten neuronalen Netzen ebenfalls Gewichte, die während des Lernvorgangs angepasst werden. Dabei lernt die Zelle, wann Daten eingelesen, ausgegeben oder gelöscht werden sollen.

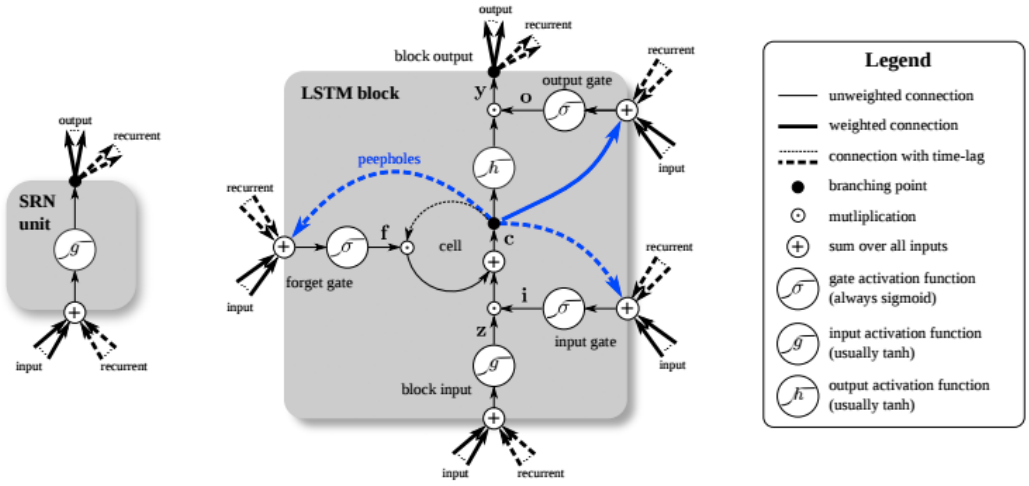


Abbildung 5: SRN (simple recurrent networks) Einheit und und LSTM (Long Short-Term Memory) Block im Vergleich[2].

# 3 Verwandte Arbeiten

## 3.1 Social LSTM: Human Trajectory Prediction in Crowded Spaces

In diesem Paper geht es um eine Vorhersage über dynamische Bewegungen von Menschenmassen. Jede Person hat ein unterschiedliches Bewegungsmuster, welches von der Geschwindigkeit, Beschleunigung und Gangart abhängig ist. Es wurden LSTM-Netze verwendet, um die allgemeine menschliche Bewegung zu erlernen und um den zukünftigen Bewegungsablauf zu ermitteln. Das Team hat sich für LSTM-Netze entschieden, da diese bereits für isolierte Sequenzen, wie zum Beispiel für die Analyse von Handschriften oder Sprachen erfolgreich eingesetzt wurden. Ein einzelnes LSTM-Model pro Person reicht nicht für die Erfassung der Bewegungsabläufe aus, da die Menschen miteinander interagieren, deshalb wurden die einzelnen LSTM-Modelle miteinander verbunden. Jede Fußgängerbewegung ist typischerweise von den Bewegungen der sich in der Nähe befindenden Personen abhängig. Ein Szenario zwischen vier Fußgängern befindet sich in der Abbildung 6. In diesem Szenario sind die verschiedenen vorhergesagten Bewegungenzu zu sehen. Person 1 bewegt sich einfach geradeaus, da diese Person keine Hindernisse vor sich hat. Die Personen 2, 3 und 4 interagieren miteinander. Person 4 macht eine Kurve um Person 3, um dieser auszuweichen. Person 3 weicht außerdem Person 2 aus. Nachdem alle Hindernisse überwunden wurden, werden die Bewegungen aller Personen linear fortgesetzt.

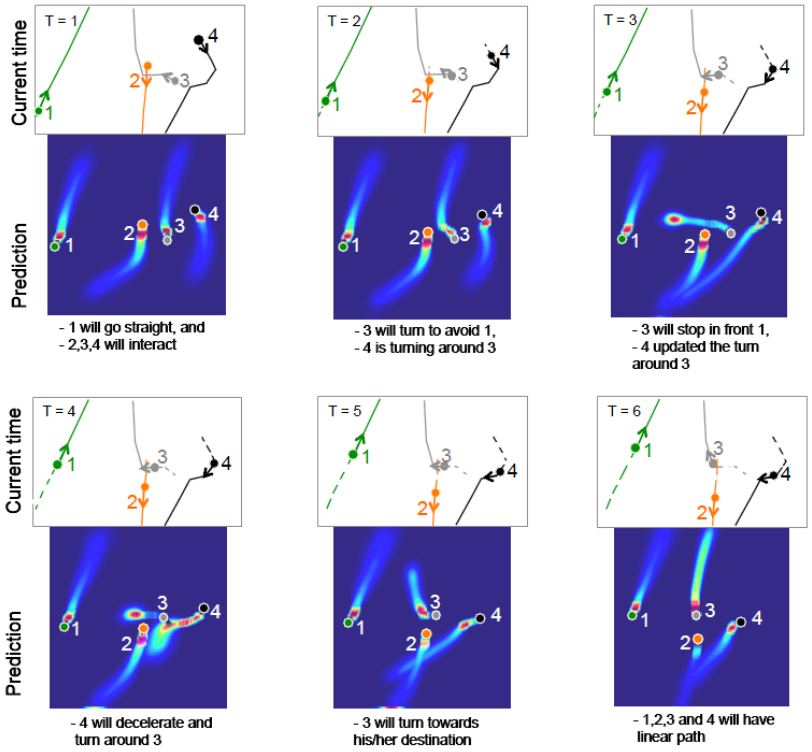


Abbildung 6: Testszenario zwischen vier Personen [6].

In der Abbildung 7 sind Vorhersagen verschiedener Algorithmen abgebildet. Die gelbe Linie (GT) entspricht dabei der tatsächlichen Bewegung. Die gestrichelte rote Linie (Social-LSTM) entspricht dem Algorithmus aus diesem Paper. In den ersten drei Zeilen schneidet der Social-LSTM Algorithmus besser ab, als die anderen. In der letzten Zeile waren andere jedoch besser.

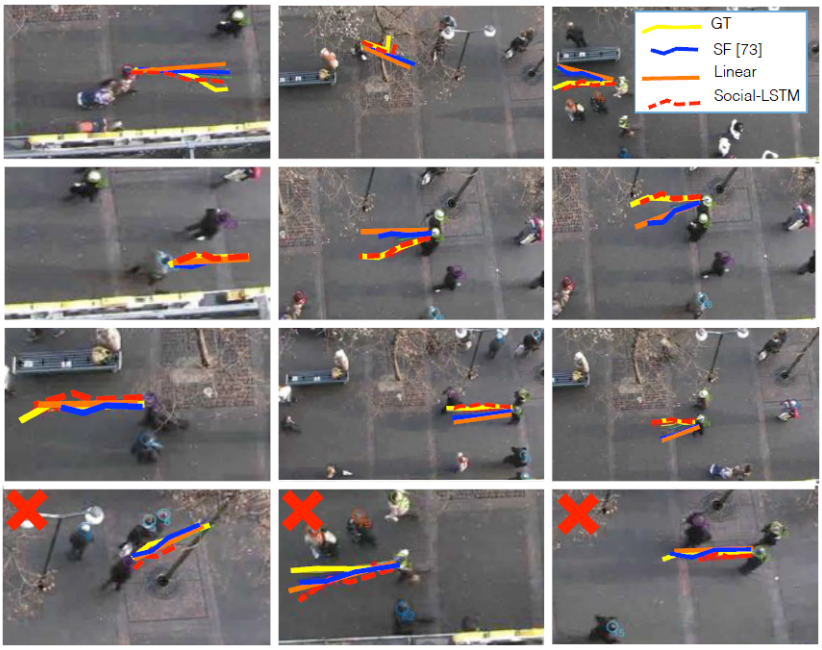


Abbildung 7: Verschiedene Algorithmen im Vergleich (GT, SF, Lienar und Social-LSTM) [6].

## 3.2 A Real-Time Pedestrian Detector using Deep Learning for Human-Aware Navigation

Das zweite Paperhandelt von einer Echtzeit-Fußgängererkennung. Der Ablauf ist in der Abbildung 8 abgebildet. Im ersten Schritt wird ein Testbild eingelesen. Anschließend wird der ACF (Aggregated Channel Features) Algorithmus angwendet, um mögliche Personen zu erkennen. Danach werden diese Vorschläge an das CNN (Convolutional Neural Network) weitergereicht. Dort findet dann die endgültige Klassifizierung statt. Man sieht im letzten Schritt, dass die Vorschläge, bei denen es sich nicht um Personen gehandelt hat ausgefiltert wurden.

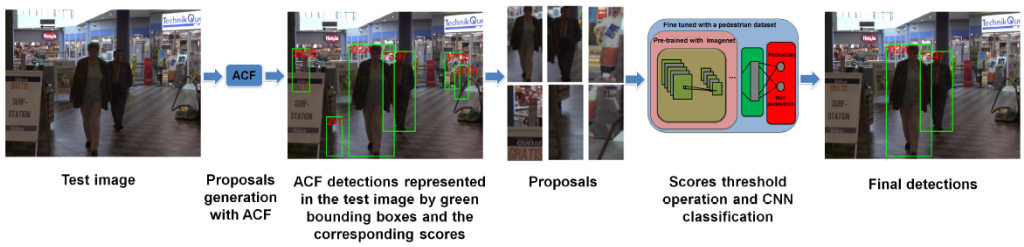


Abbildung 8: Ablauf der Personenerkennung [7].

Für die Tests wurde ein Roboter eingesetzt, der verschiedene Aufgaben lösen musste. In der Abbildung 8 ist beispielhaft ein Szenario zu sehen. In diesem Szenario geht es darum, dass der Roboter von links nach rechts gelangt. In der ersten Abbildung befindet sich noch kein Hindernis, sodass die geplante Route des Roboters linienförmig von links nach rechts verläuft. In der zweiten Abbildung taucht ein Hindernis auf, welches der Roboter wahrnimmt. Die Route ändert sich, da der Roboter links am Hindernis vorbeifahren soll, um an das Ziel zu gelangen. Anschließend bewegt er sich links am Hindernis vorbei und gelangt schließlich ans Ziel.

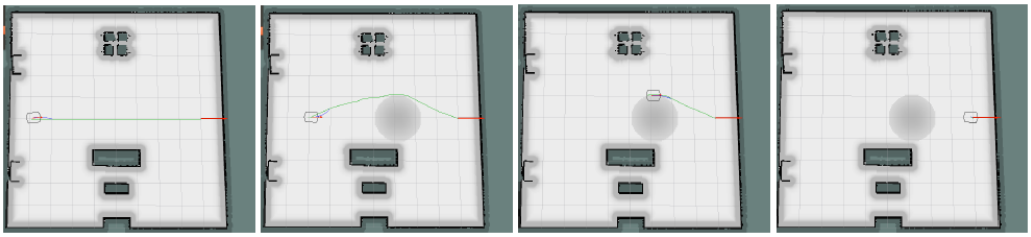

Abbildung 9: Testszenario zwischen Roboter und einem Hindernis [7].

# 4 Zusammenfassung und Ausblick

Zusammenfassend lässt sich sagen, dass das LSTM-Verfahren sehr gut geeignet ist, um Fußgänger auf der Straße zu erkennen und ihre Bewegung vorherzusagen. Das liegt daran, dass man mit LSTM im Gegensatz zu regulären rekurrenten neuronalen Netzen sowohl kürzere, als auch längere Sequenzen betrachten kann. Die Trefferquote ist bereits gut, kann aber noch besser werden, falls ein System für ein Auto entwickelt wird, welches automatisch reagieren soll, falls ein Hindernis plötzlich auftaucht. Zusätzlich könnten weitere Sensorinformationen verwendet werden, um die Genauigkeit der vorhergesagten menschlichen Bewegungen zu verbessern. Die Verwendung von Multi-LSTM-Netzen wäre ebenfalls interessant, indem jedes einzelne LSTM-Netz eine Vorhersage trifft und anschließend entschieden wird, welches am präzisesten ist. Die nächsten Schritte wären, sich mit den konkreten Technologien zu beschäftigen, um die Fußgängererkennung zu realisieren.
Viele große Unternehmen liefern Werkzeuge, um sich künstliche neuronale Netze zu erzeugen und zu trainieren, wie zum Beispiel:

- Tensorflow (Google)[14]
- CNTK - Microsoft Cognitive Toolkit (Microsoft)[15]
- Caffe[16]
- Keras[17]

Bezüglich der Trainingsdaten würden sich zum Beispiel die Daten von *Caltech Pedestrian Detection Benchmark*[18] eignen. Dort befindet sich eine Datensammlung von ca. 250.000 Bildern von Fußgängern, welche bereits aufgenommen wurden.

# Literatur


[1] C. Olah, Understanding LSTM Networks, https://colah.github.io/posts/2015-08-Understanding-LSTMs/, 2015, Zugriffsdatum: 28.07.2017

[2] A Beginner's Guide to Recurrent Networks and LSTMs, https://deeplearning4j.org/lstm.html, Zugriffsdatum: 28.07.2017

[3] A. Meisel, Vorlesungsfolien - Robot Vision, 2015

[4] Maschinelles Lernen, https://de.wikipedia.org/wiki/Maschinelles_Lernen, Zugriffsdatum: 28.07.2017

[5] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation, 1997

[6] A. Alahi et al., Social LSTM: Human Trajectory Prediction in Crowded Spaces, 2016

[7] D. Ribeiro, A Real-Time Pedestrian Detector using Deep Learning for Human-Aware Navigation, 2016

[8] David J. C. MacKay, Information Theory, Inference and Learning Algorithms. Cambridge University Press, Cambridge 2003

[9] M. Limbourg, Überforderte Kinder im Straßenverkehr - Welche Forderungen stellt die Kinderpsychologie an das Zivilrecht? https://www.uni-due.de/~qpd402/alt/texte.ml/Goslar.html, Zugriffsdatum: 05.08.2017

[10] Unfallstatistik aktuell, https://www.dvr.de/betriebe_bg/daten/unfallstatistik/de_jahre.htm, Zugriffsdatum: 05.08.2017

[11] Deep Learning, https://de.wikipedia.org/wiki/Deep_Learning

[12] Rekurrentes neuronales Netz, https://de.wikipedia.org/wiki/Rekurrentes_neuronales_Netz

[13] Künstliches Neuron, https://de.wikipedia.org/wiki/K%C3%BCnstliches_Neuron

[14] Tensorflow - https://www.tensorflow.org/

[15] CNTK - Microsoft Cognitive Toolkit - https://www.microsoft.com/en-us/research/product/cognitive-toolkit/

[16] Caffe - http://caffe.berkeleyvision.org/

[17] Keras - https://keras.io/

[18] Caltech Pedestrian Detection Benchmark - http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/