Informatics Inside

Informatics Inside
Herbst 2020

Tagungsband

Herausgeber:
Uwe Kloos, Hochschule Reutlingen
Natividad Martinez, Hochschule Reutlingen
Gabriela Tullius, Hochschule Reutlingen

Impressum

Hochschule Reutlingen
Reutlingen University

Anschrift:

Hochschule Reutlingen
Reutlingen University
Fakultät Informatik
Human-Centered Computing
Alteburgstraße 150
D-72762 Reutlingen

Telefon: +49 7121 / 271-4002
Telefax: +49 7121 / 271-4042

E-Mail: infoinside@reutlingen-university.de
Website: infoinside.reutlingen-university.de

Verlag: Hochschule Reutlingen
ISBN 978-3-00-066747-3

Vorwort

Liebe Leserinnen und Leser,

ich darf Sie herzlich zur ersten Herbstkonferenz der Informatics Inside begrüßen! Seit der ersten Informatics Inside 2009 fand sie ausschließlich im Frühjahr statt. Im Herbst fand eine deutlich kleinere Konferenz, die wvk, mit Posterpräsentationen statt. Aufgrund der steigenden Studierendenzahlen und der anhaltend hohen Qualität der Beiträge ist der nun vollzogene Schritt, zwei große Konferenzen im Jahr abzuhalten, folgerichtig. Masterstudierende des Studiengangs Human-Centered Computing organisieren mit sehr hohem Engagement und Aufwand diese Konferenz mit dem Ziel, ihre Forschungstätigkeiten im Rahmen des Studiums einer breiten Öffentlichkeit zu präsentieren und in den Diskurs zu gehen. Der Diskurs bietet allen Teilnehmenden die Gelegenheit, Neues zu erfahren, eigene Standpunkte und Auffassungen zu überdenken sowie die Vielfältigkeit der Themen, aber auch die Vortragenden facettenreich zu erleben.

Als Motto für die Herbstkonferenz Informatics Inside 2020 wurde KInside gewählt. Wieder einmal blicken die Studierenden inside und schauen sich Methoden, Anwendungen und Zusammenhänge genauer an. Die Beiträge sind vielfältig und entsprechend dem Studiengang human-centered. Es ist der Anspruch, dass sich die Themen um die Bedarfe der Menschen drehen und eingesetzte Methoden kein Selbstzweck sind, sondern am Nutzen für den Menschen gemessen werden.

Nachdem die Informatics Inside im Sommersemester bereits online durchgeführt werden musste, wird sie auch im Herbst ausschließlich digital erlebbar sein. Ich wünsche allen eine interessante Konferenz mit neuen spannenden Einsichten. Stellen Sie sich dem Diskurs und seien Sie damit ein Teil des digitalen Erlebnisses!

Reutlingen im November 2020

Prof. Dr. Gabriela Tullius

Inhaltsverzeichnis

Anforderungen an ein vision-basiertes System zur Erkennung epileptischer Anfälle

Anna Chiara Rosa
Hochschule Reutlingen
Anna_Chiara.Rosa @Student.Reutlingen-University.de

Abstract

Epileptische Anfälle können für den Betroffenen eine Gefahr darstellen, welche durch die Anfalls-Überwachung gemindert werden kann. Die automatisierte vision-basierte Erkennung von epileptischen Anfällen geht jedoch mit einigen Herausforderungen der Computer Vision einher. In dieser Arbeit werden grundlegende Kenntnisse über epileptische Anfälle und relevante Computer Vision-Methoden vorgestellt. Anhand dieser und weiterer Literatur werden Anforderungen an ein System zur Anfallserkennung abgeleitet und diskutiert.

CCS Concepts

•Computing methodologies~Artificial intelligence~Computer vision•Applied computing~Life and medical sciences

Keywords

Vision-Based Seizure Detection, Epilepsy, Computer Vision, Action Recognition, Requirements

---

Betreuer Hochschule: Prof. Dr.-Ing. Cristóbal Curio
Hochschule Reutlingen
Cristobal.Curio@Reutlingen-University.de
Betreuer Firma: Dr. Michael Gebhart
Iteratec GmbH
Michael.Gebhart@iteratec.com

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

1 Einleitung

1.1 Motivation und Ziel

Epilepsie ist eine neurologische Erkrankung, welche mit unterschiedlichen Arten von wiederkehrenden Anfällen einhergeht. Diese sind für die Betroffenen häufig eine Belastung und können eine Gefahr für deren Leben darstellen.

Ein System zur Erkennung epileptischer Anfälle kann Betroffene entlasten. So wird davon ausgegangen, dass viele Betroffene und deren Angehörige Schwierigkeiten haben, die Häufigkeit von Anfällen korrekt einzuschätzen. Da Ärzte in der Behandlung von diesen Angaben abhängig sind, kann es durch die fehlerhafte Einschätzung zu einer Über- oder Untermedikation des Patienten kommen. Dies kann zu einer Verstärkung von Nebenwirkungen führen oder bei Unterbehandlung zum Persistieren der Anfälle. [7]

Besonders generalisierte motorische Anfälle können eine Gefahr für das Leben des Patienten darstellen. Es wird davon ausgegangen, dass eine zeitnahe Erkennung von nächtlichen Anfällen mit einem verringerten Risiko für den Patienten einhergeht. [20]

1.2 Vorgehen

In der vorliegenden Arbeit wird untersucht, welche Anforderungen an ein vision-basiertes System zur Erkennung epileptischer Anfälle bestehen. Diese werden im Kontext des State of the Art zur Anfalls-, Posen- und Aktionserkennung diskutiert. Hierzu werden

die Erkrankung Epilepsie und die damit verbundenen Anfallsarten vorgestellt. Darauffolgend werden aktuell bestehende Systeme zur Anfallserkennung, sowie der State of the Art in Bezug auf die Aktions- und Posen-Erkennung dargestellt. Darauf aufbauend untersucht und diskutiert die Arbeit Anforderungen an vision-basierte Systeme zur Erkennung von Anfällen.

2 Epilepsie

2.1 Definition

Epilepsie ist eine neurologische Erkrankung, die eine heterogene Gruppe von Störungen zusammenfasst, bei denen der Patient eine erhöhte Wahrscheinlichkeit hat, epileptische Anfälle zu erleiden [12]. Diese Anfälle werden durch „abnorm exzessive oder synchrone neuronale Aktivität im Gehirn“ [12] ausgelöst. Es gibt über 30 Formen der Epilepsie und über 10 Anfallsformen [23]. Von Epilepsie spricht man, wenn es zu „wiederholten, nicht-provozierten“ [17] Anfällen kommt oder wenn durch EEG- oder MRT-Diagnostik auf eine erhöhte Anfallsbereitschaft geschlossen werden kann [17]. Goldstandard zur Diagnostik ist das Video-EEG, über das Anfälle in Video und EEG abgeglichen werden können [14].

Bei über 10% aller Menschen kommt es mindestens einmal im Leben zum Auftreten eines epileptischen Anfalls. Somit handelt es sich bei der Epilepsie um eine der häufigsten neurologischen Erkrankungen. Vor allem im Kindesalter und ab dem 60. Lebensjahr kommt es zu einem starken Anstieg der Neuerkrankungen. [5]

2.2 Anfalls-Arten

Epileptische Anfälle werden primär über ihren Ursprung im Gehirn unterteilt, je nachdem, ob sie sich auf eine Hemisphäre beschränken oder sich über beide Hemisphären ausbreiten [11]. Darüber hinaus werden die jeweiligen Arten über die Charakteristik des Anfalls unterteilt.

Bei motorischen Anfällen handelt es sich um Anfälle, bei denen die Muskulatur betroffen ist [6]. Dies kann sowohl bei fokalen, als auch bei generalisierten Anfällen der Fall sein. Es gibt einerseits einfache motorische Anfälle, bei denen es zu unnatürlichen Bewegungen kommt [4]. Andererseits treten komplexe Anfällen auf, bei denen Bewegungen vorkommen, welche in einem anderen Kontext als normale Bewegung eingeordnet werden können [4]. Bei einem atonischen Anfall kommt es zu einem Verlust oder einer Verminderung des Muskeltonus der betroffenen Region [6]. Bei tonischen Anfällen kommt es zur wenige Sekunden bis Minuten anhaltenden Verkrampfung von Muskelgruppen durch hochfrequente Muskelkontraktionen [6, 22]. Klonische Anfälle zeichnen sich durch regelmäßige, andauernde Zuckungen aus, welche bei einer Frequenz von 2-3 Kloni / Sekunde auftreten [6]. Bei myoklonischen Anfällen kommt es zu einzelnen oder mehreren kurzen Zuckungen [6].

Fokale Anfälle beschreiben Anfälle, die sich auf eine Hirnhälfte beschränken [5]. Die Symptomatik des Anfalls hängt davon ab, welche Hirnregion betroffen ist [17]. Fokale Anfälle können einerseits danach eingeteilt werden, ob der Anfall bewusst erlebt wird, andererseits darüber, ob es sich um einen motorischen oder nichtmotorischen Anfall handelt [11]. Motorische fokale Anfälle zeichnen sich durch eine einseitige Betroffenheit aus [17]. Zu den nichtmotorischen fokalen Anfällen zählen autonome, kognitive, emotionale und sensorische Anfälle [11]. Bei kognitiven Anfällen hat der Patient beispielsweise Einschränkungen in der Sprache, dem Denken oder Halluzinationen [11].

Als generalisierte Anfälle werden Anfälle bezeichnet, die sich über Teile beider Hirnhälften ausbreiten [5]. Diese Anfälle machen 1/3 alles Anfälle aus [17]. Sie werden ebenfalls in motorische und nichtmotorische Anfälle unterteilt [11]. Bei nichtmotorischen generalisierten Anfällen handelt es sich um Anfälle, bei denen es zu einer plötzlichen

Unterbrechung der Aktivität über wenige Sekunden bis zu einer Minute kommt und auf Ansprache nicht reagiert wird [11].

Generalisierte klonische Anfälle gehen einher mit „rhythmischen beidseitiger Zuckungen der Extremitäten sowie häufig von Kopf, Nacken, Gesicht und Rumpf" [11]. Bei generalisierten tonischen Anfällen kommt es zur beidseitigen Versteifung, sowie zum Anheben der Extremitäten und des Nackens [11]. Die Extremitäten können sich in unnatürlichen Stellungen der Beugung oder Streckung befinden [11]. Es tritt eine Frequenz tonischer und klonischer Phasen auf [6]. Sie zeichnen sich aus „durch Bewusstlosigkeit, Sturz, Verkrampfung am ganzen Körper, Zuckungen der Arme und Beine und einen nachfolgenden Erschöpfungs- oder Verwirrtheitszustand" [17].

Diese Anfälle müssen von anderen Krankheitsbildern mit ähnlichem Erscheinungsbild abgegrenzt werden. So ist beispielsweise eine Unterscheidung zu Ohnmachtsanfällen oder psychogenen nicht-epileptischen Anfällen nötig [22, 26].

Bei einem generalisierten klonisch-tonischen Anfall handelt es sich in jedem Fall um einen Notfall. Klingt der Anfall nicht innerhalb weniger Minuten ab, wird ein ärztliches Eingreifen nötig. Während eines Anfalls sollte man die Umgebung des Patienten sichern und die Dauer und Art des Anfalls beobachten. [23]

2.3 Komorbidität und Risiken

Zu den Komorbiditäten von Epilepsie zählen beispielsweise Osteoporose und gastrointestinale Beschwerden, aber auch psychische Erkrankungen und neuropsychologische Beeinträchtigungen. Sie können mit der Erkrankung, der Medikation, sowie gesellschaftlichen Stigmata zusammenhängen. Während eines epileptischen Anfalls ist der Betroffene außerdem einem erhöhten Verletzungsrisiko ausgesetzt. [23]

Bei Epilepsie-Patienten ist das Risiko, an einem nicht natürlichen Tod durch Suizid, Unfälle und gewünschte oder versehentliche Arzneimittelvergiftung zu sterben, erhöht [16]. Vor allem bei (nächtlichen) tonisch-klonischen generalisierten Anfällen kann der SUDEP (sudden unexpected death in epilepsy, dt. plötzlicher unerwarteter Tod bei Epilepsie) auftreten [21]. Dabei stirbt der Betroffene meist im Kontext eines Anfalls [21]. Das Risiko an SUDEP zu sterben, sinkt, wenn nächtliche Anfälle erkannt werden, beispielsweise wenn der Betroffene nicht allein schläft oder überwacht wird [20].

3 State of the Art

3.1 Posenschätzung

Posenschätzung behandelt das Problem des Auffindens anatomisch relevanter Regionen des menschlichen Körpers [8] zur Beschreibung einer 2D- oder 3D-Pose.

In der 2D-Erkennung finden häufig Convolutional Neural Networks (CNNs) Anwendung [19]. Hierbei gibt es Methoden, welche aus einzelnen RGB-Bildern erst die Gelenke detektieren und deren Zusammenhänge dann modellieren, beispielsweise in [8]. Daneben existieren holistische Ansätze, bei denen die Pose im Bild direkt detektiert wird, beispielsweise durch CNNs [27].

Die 3D-Posen-Schätzung aus einzelnen Bildern kann unterschieden werden zwischen Methoden, welche aus Bildern erst die 2D-Pose erkennen und aus dieser die 3D-Pose ableiten und solchen, die die Pose direkt ableiten. Bei der Ableitung aus der 2D-Pose finden unterschiedlich komplexe Verfahren zum Schätzen der Tiefeninformationen Anwendung, beispielsweise Nearest-Neighbour-Verfahren und neuronale Netze. Bei der direkten Ableitung von 3D-Posen aus RGB-Bildern finden ebenfalls häufig CNNs Anwendung. Das Training der CNNs benötigt eine große Menge an Trainingsdaten, die im Fall von 2D-Schätzung manuell gelabelt werden können. In der 3D-Anwendung werden die Posen hingegen über Motion Capture

Systeme erfasst. Somit findet die Erfassung in einer eingeschränkten Umgebung statt, wodurch die Übertragbarkeit auf Szenen in der echten Welt begrenzt ist. [27]

Zur Erkennung der Pose in 3D werden neben RGB-Bildern auch Daten von Tiefen-Kameras verwendet. Sie bringen bei Verdeckungen und anspruchsvollen Belichtungs-Verhältnissen Vorteile und ermöglichen somit eine robuste, schnelle Posen-Erkennung [29]. Allerdings geht die Verwendung von Tiefen-Daten mit erhöhten Kosten, geringer Sensor-Genauigkeit und beschränkter Einsetzbarkeit einher [29]. So ist die Kinect nur im Innenraum nutzbar [19].

Die Posen-Erkennung ist anspruchsvoll, wenn es zu Verdeckung von Körperteilen durch den Körper selbst oder andere Gegenstände und Personen kommt und die Körperteile in dem nicht sichtbaren Raum geschätzt werden müssen. Ein Verfahren zur Erkennung mehrerer, auch verdeckender, Personen in einem Bild wird in [8] beschrieben.

3.2 Aktionserkennung

Die Erkennung von menschlichen Aktionen (Action Recognition) ist ein anspruchsvolles Problem, einerseits durch die Diversität des menschlichen Körpers, andererseits durch die Komplexität und Variabilität menschlicher Bewegungen und Aktionen [9]. So kann sich die Ausführung der gleichen Aktion durch die gleiche Person bei mehrfacher Wiederholung unterscheiden. Vor allem bei unterschiedlichen Personen sind Unterschiede in der Ausführung zu erwarten.

Man unterscheidet in der Action Recognition zwischen der Aktions-Klassifikation von Bild-Sequenzen mit einer einzigen Aktion und dem Auffinden von Aktionen in einer Bildsequenz, die auch andere Aktionen enthalten kann, der Aktions-Detektion. Die Detektion von Aktionen basiert meist auf dem Konzept des Sliding Window und zeichnet sich durch eine hohe rechnerische Komplexität aus. [29]

Verfahren zur Aktionserkennung basieren auf Handcrafted Action Features oder End-To-End Deep Learning. Unter Handcrafted Action Features versteht man unterschiedliche Ansätze, menschliche Bewegungen über Raum und Zeit hinweg zu kodieren. Hierzu können im RGB- und Tiefenbilder-Bereich Trajektorien der abgeleiteten Joints verwendet werden, aber auch andere Features. Wichtig ist, dass Features extrahiert werden über die Aktivitäten robust erkannt werden können. Skeleton-basierte Verfahren sind abhängig von der Performanz der Posen-Erkennung. Auch bei der End-To-End Action Recognition mit tiefen neuronalen Netzwerken, die die Features eigenständig lernen, gibt es unterschiedliche Ansätze zum Verwenden von RGB-Daten. So werden beispielsweise Bilder und Optical Flow gemeinsam in Two Stream Convolutional Networks genutzt oder die Bewegungen von detektieren Skeletons gelernt. Um zeitliche Abläufe in die Erkennung zu integrieren, werden beispielsweise LSTMs (long short-term memory) und 3D Convolution Networks verwendet. [29]

Daneben gibt es auch Ansätze, die Tiefen-Daten und daraus abgeleiteten Skeletons verwenden [19] oder direkt neuronale Netzwerke anhand der Tiefen-Daten trainieren [29]. Die Verwendung von Tiefen-Daten in der Aktionserkennung hat den Vorteil, dass das Verfahren robuster bei Änderungen und Bewegungen im Hintergrund, sowie gegenüber Belichtungs-Verhältnissen ist [9, 29].

Neben der Sequenz-Erkennung können Aktionen statisch in einzelnen Bildern erkannt werden. Es hängt von der Aktion ab, ob einzelne Bilder zur Erkennung ausreichen oder ob Videos besser geeignet sind. [13]

3.3 Geräte zur Erkennung epileptischer Anfälle

In [7] werden bestehende, medizinisch zugelassene oder peer-reviewte Geräte zur Erkennung epileptischer Anfälle vorgestellt. Es

wurden dabei keine auf Video-Daten basierenden Systeme gefunden. Stattdessen gibt es Geräte zum Erkennen von Absence-Anfällen über EEG und motorischen Anfällen über Bewegungsmesser-Armbänder / -Uhren, Bewegungs- und Audio-Sensoren unter der Matratze für nächtliche Anfälle, sowie Oberflächen-Elektromyographie und weitere tragbare, multimodale Geräte. Die Geräte besitzen teilweise die Zulassung als Medizinprodukt in den USA oder der EU. [7]

In einer Auswertung unterschiedlicher, teils multimodaler Geräte zur Erkennung motorischer Anfälle in [18] zeigt sich, dass nahezu alle in der Arbeit betrachteten Geräte in Epilepsy Monitoring Units (EMU), also im klinischen Umfeld, entwickelt und evaluiert werden, da dort Video-EEG Daten als Goldstandard der Diagnostik erfasst werden. Daraus folgt die Frage der Übertragbarkeit der Ergebnisse, da nicht klar ist, ob die in den EMUs gemessene Sensitivität und False Detection Rate der Geräte für den heimischen Einsatz repräsentativ sind. Die Autoren gehen davon aus, dass dies nicht gegeben ist, da das heimische Verhalten der Nutzer von dem Verhalten im klinischen Umfeld abweicht, beispielsweise in Bezug auf nächtliches Aufstehen. [18]

3.4 Vision-basierte Krampferkennung

Bestehende Verfahren zur vision-basierten Krampferkennung können unterteilt werden in jene, welche konventionelle Motion Analysis Verfahren anwenden und aktueller solche, welche Machine Learning und insbesondere Deep Learning verwenden [28].

Die ausgewerteten Arbeiten in [1–3] basieren auf in EMUs in Krankenhäusern erfassten Daten.

In [3] wurden 161 RGB-Videos von Anfällen aufgezeichnet, um zwei Anfallsformen automatisiert zu unterscheiden. Dabei wird eine möglichst hohe Variabilität der Anfalls-Muster angestrebt. Trainings-, Test- und Validierungsdatensätze bestehen aus Daten von unterschiedlichen Patienten. In den Videos werden das Gesicht, der Oberkörper mit Kopf-Drehung, sowie die rechte und linke Hand getrennt detektiert. Aus der Detektion des Oberkörpers mit Kopf wird die 3D-Pose abgeleitet und als Input für ein LSTM-Netzwerk verwendet, bei Gesicht und den Händen werden die RGB-Daten direkt für ein LSTM genutzt. Hierbei zeigt sich, dass das Einbeziehen unterschiedlicher Körper-Regionen für Klassifikation der Formen der Epilepsie sinnvoll ist. [3]

In [1] werden epileptische Anfälle anhand von Tiefen- und Infrarot-Daten unter Verwendung von CNNs erkannt. Die Ergebnisse übertreffen dabei klassische Methoden und sind bei getrenntem Training der CNNs für Infrarot- und Tiefen-Daten am besten, eine späte Fusion bringt somit die besten Ergebnisse hervor. Darüber hinaus ist das System echtzeitfähig. [1]

In [2] wird bei der Erkennung epileptischer Anfälle der Datensatz um Videos von psychogenen nicht-epileptischen Anfällen erweitert (40 epileptische Anfälle, 10 nicht-epileptische Anfälle), die nicht als Anfall klassifiziert werden sollen. Dabei wird einerseits eine landmark-basierte Erkennung umgesetzt, bei der ein Skeleton-Modell des Patienten berechnet wird und gemeinsam mit Optical Flow für das Training eines LSTM-Netzwerks genutzt wird. Daneben wird eine region-basierte Erkennung umgesetzt, bei der ein nicht tiefes CNN als Basis für ein LSTM verwendet wird. Die Verwendung eines tiefen State-of-the-Art CNN ist aufgrund der verfügbaren Datenmenge nicht möglich, da dies mit einer hohen Anzahl an Parametern einhergeht und Overfitting zu erwarten ist. Der region-basierte Ansatz bringt dabei bessere Ergebnisse hervor als der landmark-basierte Ansatz. [2]

4 Anforderungen

Die Anforderungen an ein System zur visuellen Erkennung von epileptischen Anfällen sind geknüpft an den Anwendungsfall des

Systems. Grundsätzlich ist davon auszugehen, dass nur motorische Anfälle visuell detektierbar sind, andere Anfälle benötigen zur Erkennung meist ein EEG. [18]

Ein Anwendungsfall ist das Tracking der Häufigkeit von Anfällen, um darauf aufbauend die Therapie anpassen zu können [7]. Ebenso kann die Erkennung von Anfällen genutzt werden, um einen Alarm auszulösen, so dass vor allem bei Nacht dem Betroffenen, wenn nötig, geholfen werden kann [7]. Dabei ist eine Unterscheidung zwischen fokalen und generalisierten Anfällen notwendig, da vor allem von generalisierten Anfällen eine Gefahr für den Betroffenen ausgeht. Ein weiterer Anwendungsfall ist die Unterstützung bei der Anamnese. Hierzu muss die Art des Anfalls über die betroffene Seite (generalisiert / fokal) und die ausgeführten Bewegungen unterschieden werden, aber auch die Dauer, Frequenz und Entwicklung des Anfalls sind relevant [24]. Darüber hinaus können auch die Umstände des Anfalls (z.B. ob es sich um einen nächtlichen Anfall handelt und ob der Patient vor oder nach dem Anfall schläft) für den Arzt interessant sein [24].

Um die Bewegungen robuster detektieren zu können, kann die technische Einbeziehung von Tiefendaten sinnvoll sein. Oft ist auch die nächtliche Überwachung nötig [24]. Dies ist beispielsweise mit Hilfe von Infrarot-Sensoren möglich.

In Abhängigkeit von der zu detektierenden Anfallsart und den betroffenen Körperteilen können unterschiedliche Anforderungen an die Auflösung der Sensoren entstehen. Dies ist beispielsweise der Fall, wenn auch feine Bewegungen des Gesichts oder der Finger, anstatt nur grobe Bewegungen von Armen und Beinen, erfasst werden sollen [24]. Bei besonders heftigen und schnellen Anfalls-Bewegungen kann die verbreitete Bildfrequenz von 25 fps nicht mehr ausreichen und es wird eine höhere Bildfrequenz empfohlen [10]. Dies kann jedoch mit Problemen bei der Speicherung und Verarbeitung der Daten einhergehen [24], was vor allem in Bezug auf die Echtzeitfähigkeit relevant ist.

Eine für Betroffene relevante Anforderung ist die hohe Zuverlässigkeit des Geräts. Konkrete akzeptable Werte hängen dabei vom Anwendungsfall ab. Die Untersuchung unterschiedlicher Studien zur benötigten Zuverlässigkeit von Systemen zur Anfallserkennung für Alarme in [7] zeigt, dass die Systeme eine Sensitivität über 90% benötigen und über eine geringe False Alarm Rate von 0,14 / Tag oder 1 / Woche verfügen sollen. Die hohe Sensitivität ist wichtig, damit möglichst viele der Anfälle korrekt identifiziert werden können [7]. Die niedrige False Alarm Rate dient dazu, dass der Betroffene und die pflegenden Personen selten durch falsche Alarme gestört und verunsichert werden [7]. Dahingegen ist bei einer Prüfung der Therapie-Erfolge die reine Erkennbarkeit einer Abnahme der Anfälle ausreichend [7].

Da es sich bei den Anfalls-Daten um Patientendaten handelt, ist die Vertraulichkeit der Erfassung, Speicherung, Nutzung und Sicherheit der Daten für den Betroffenen von Interesse [7]. Ebenso müssen gesetzliche Vorgaben zum Datenschutz eingehalten werden.

Die Erkennung muss robust gegenüber Verdeckung durch helfende Personen oder Gegenstände, z.B. Decken, sein. Darüber hinaus ist für eine Erkennung im heimischen Umfeld des Betroffenen wichtig, dass die Erkennung robust gegenüber unterschiedlichen Umgebungen ist.

Die Echtzeitfähigkeit ist eine elementare Anforderung an ein System zur visuellen Krampferkennung zur Alarmierung Pflegender [7]. Bei der Überwachung von Behandlungserfolgen ist dies weniger relevant.

Durch den verwendeten Datensatz kann ein Bias in der Erkennung entstehen. So können durch unausgeglichene Datensätze beispielsweise in Bezug auf die Hautfarbe oder das Geschlecht diskriminierende Algorithmen

entstehen [15]. Das System muss Krampfanfälle zuverlässig und unabhängig von Altersgruppe, Herkunft und Ethnien, Geschlecht und Behinderung erkennen. Da vor allem bei älteren oder behinderten Menschen durch Begleiterkrankungen Posen auftreten, die Verkrampfungen ähnlichsehen können, ist es wichtig, dass ein System diese von Anfällen unterscheiden kann.

5 Diskussion

Die visuell detektierbaren motorischen Anfälle sind von einer hohen Komplexität und einer hohen Variabilität zwischen Betroffenen und Anfällen gekennzeichnet. Somit ist der Einsatz nachgestellter Daten fragwürdig [25]. Entsprechend herausfordernd ist die fehlende Verfügbarkeit größerer visueller Anfalls-Datensätze.

Die vorgestellten Arbeiten verwenden im klinischen Umfeld aufgenommene Anfalls-Videos, deren Umfang nicht mit großen Action-Recognition-Datensätzen vergleichbar ist. State-of-the-Art-Verfahren zur Erkennung von Aktionen und Krampfanfällen basieren auf neuronalen Netzen, die für ihr Training große Datenmengen benötigen.

Eine Möglichkeit damit umzugehen ist die Analyse der statischen Pose zur Erkennung, ob in einem Bild ein Anfall zu sehen ist. Dies ist möglich, da die Haltung einzelner oder mehrere Körperteile bei Anfällen häufig, jedoch nicht immer, von alltäglichen Posen abweicht. Es ist jedoch möglich, dass Posen Spasmen, welche durch anderweitige Erkrankungen entstehen, ähneln und erst Bewegungsmuster diese unterscheiden. Darüber hinaus sind Charakteristika wie Frequenz und Entwicklung des Anfalls nur begrenzt anhand von Einzelbildern erkennbar. Somit erscheint das Einbeziehen zeitlicher Informationen sinnvoll.

Es stellt sich die Frage, ob die Verwendung von klinischen Daten auf die Anwendung im heimischen Kontext übertragbar ist. Es ist außerklinisch eine höhere Variabilität der Aktionen des Betroffenen, die nicht erkannt werden sollen, zu erwarten. Außerdem besteht dort eine höhere Variabilität der Umgebung. Es ist es denkbar, dass Anfälle im heimischen Umfeld anders aussehen als im klinischen, beispielsweise wenn der Patient einen Anfall sitzend erleidet oder stürzt, während in klinischen Daten meist Anfälle im Klinikbett enthalten sind.

6 Fazit

Die Erkennung von motorischen Anfällen kann für Betroffene, Pflegende und Behandelnde eine Erleichterung im Alltag und der Behandlung darstellen. Es gibt dabei unterschiedliche Anwendungsfälle, von der Alarmierung Pflegender bis zur Unterstützung von Ärzten bei der Behandlung. Damit gehen variable Anforderungen in Bezug auf die Zuverlässigkeit der Erkennung, die Performance, sowie die Auflösung und Form der Daten-Erfassung einher. Übergreifend ist jedoch der Datenschutz eine wichtige Anforderung.

Aktuell ist die Verfügbarkeit von Daten ein Problem für die visuelle Erkennung der Anfälle, da für State-of-the-Art Verfahren zur Aktionserkennung, welche meist auf Deep Learning basieren, große Datenmengen benötigt werden. Auch ohne die Verwendung von Deep Learning werden für das Abbilden der Variabilität der Anfalls-Ausprägung viele Daten benötigt. Selbst wenn ein solcher Datensatz aus klinischen Daten zur Verfügung steht, stellt sich die Frage, inwiefern dieser in den heimischen Bereich übertragbar ist und ob somit die Anforderung der Zuverlässigkeit erfüllt werden kann.

Darüber hinaus gibt es einen Konflikt zwischen der Genauigkeit der Ergebnisse und der Echtzeitfähigkeit des Systems. Für eine hohe Genauigkeit kann es notwendig sein, unterschiedliche Daten-Quellen mit hoher Auflösung zu analysieren, um beispielsweise auch Änderungen im Gesicht erkennen zu können. Dies kann das vor allem in der Aktions-Detektion vorhandene Problem der hohen rechnerischen Komplexität weiter verstärken.

Für die zukünftige Entwicklung ist der Aufbau eines großen Datensatzes epileptischer Anfälle wichtig, um so die Erkennung und Klassifizierung der durch hohe Variabilität gekennzeichneten Anfälle zu verbessern.

Literaturverzeichnis

[1] Felix Achilles, Federico Tombari, Vasileios Belagiannis, Anna M. Loesch, Soheyl Noachtar, and Nassir Navab. 2018. Convolutional neural networks for real-time epileptic seizure detection. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, 6(3), 264-269. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization* 6, 3, 264–269. DOI: https://doi.org/10.1080/21681163.2016.1141062.

[2] David Ahmedt-Aristizabal, Simon Denman, Kien Nguyen, Sridha Sridharan, Sasha Dionisio, and Clinton Fookes. 2019. Understanding Patients' Behavior: Vision-Based Analysis of Seizure Disorders. *IEEE journal of biomedical and health informatics* 23, 6, 2583–2591. DOI: https://doi.org/10.1109/JBHI.2019.2895855.

[3] David Ahmedt-Aristizabal, Clinton Fookes, Simon Denman, Kien Nguyen, Tharindu Fernando, Sridha Sridharan, and Sasha Dionisio. 2018. A hierarchical multimodal system for motion analysis in patients with epilepsy. *Epilepsy & behavior : E&B* 87, 46–58. DOI: https://doi.org/10.1016/j.yebeh.2018.07.028.

[4] David Ahmedt-Aristizabal, M. S. Sarfraz, Simon Denman, Kien Nguyen, Clinton Fookes, Sasha Dionisio, and Rainer Stiefelhagen. 2019. Motion Signatures for the Analysis of Seizure Evolution in Epilepsy. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference* 2019, 2099–2105. DOI: https://doi.org/10.1109/EMBC.2019.8857743.

[5] Christoph Baumgartner, P. Gallmetzer, S. Pirker, and B. Schimka. 2010. Epilepsie: Aktuelles zu Diagnostik und Therapie. *Psychopraxis* 13, 3, 30–33. DOI: https://doi.org/10.1007/s00739-010-0228-2.

[6] W. T. Blume, H. O. Lüders, E. Mizrahi, C. Tassinari, W. van Emde Boas, and J. Engel. 2001. Glossary of descriptive terminology for ictal semiology: report of the ILAE task force on classification and terminology. *Epilepsia* 42, 9, 1212–1218. DOI: https://doi.org/10.1046/j.1528-1157.2001.22001.x.

[7] Elisa Bruno, Pedro F. Viana, Michael R. Sperling, and Mark P. Richardson. 2020. Seizure detection at home: Do devices on the market match the needs of people living with epilepsy and their caregivers? *Epilepsia. DOI: https://doi.org/*10.1111/epi.16521.

[8] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2016. *Real-time Multi-Person 2D Pose Estimation using Part Affinity Fields*.

[9] Lulu Chen, Hong Wei, and James Ferryman. 2013. A survey of human motion analysis using depth imagery. *Pattern Recognition Letters* 34, 15, 1995–2006. DOI: https://doi.org/10.1016/j.patrec.2013.02.006.

[10] J. P. S. Cunha, C. Vollmar, J. M. Fernandes, and S. Noachtar. 2009. Automated Epileptic Seizure Type Classification through Quantitative

Movement Analysis. In *World Congress on Medical Physics and Biomedical Engineering 2009. 7 - 12 September, 2009, Munich, Germany*, Olaf Dössel, Ed. IFMBE Proceedings, 25. Springer, Berlin [u.a.], 1435–1438. DOI: https://doi.org/10.1007/978-3-642-03882-2_380.

[11] Robert S. Fisher, J. H. Cross, Carol D'Souza, Jacqueline A. French, Sheryl R. Haut, Norimichi Higurashi, Edouard Hirsch, Floor E. Jansen, Lieven Lagae, Solomon L. Moshé, Jukka Peltola, Eliane Roulet Perez, Ingrid E. Scheffer, Andreas Schulze-Bonhage, Ernest Somerville, Michael Sperling, Elza M. Yacubian, and Sameer M. Zuberi. 2018. Anleitung („instruction manual") zur Anwendung der operationalen Klassifikation von Anfallsformen der ILAE 2017. *Z. Epileptol.* 31, 4, 282–295. DOI: https://doi.org/10.1007/s10309-018-0217-7.

[12] Robert S. Fisher, Walter van Emde Boas, Warren Blume, Christian Elger, Pierre Genton, Phillip Lee, and Jerome Engel. 2005. Epileptic seizures and epilepsy: definitions proposed by the International League Against Epilepsy (ILAE) and the International Bureau for Epilepsy (IBE). *Epilepsia* 46, 4, 470–472. DOI: https://doi.org/10.1111/j.0013-9580.2005.66104.x.

[13] Guodong Guo and Alice Lai. 2014. A survey on still image based human action recognition. *Pattern Recognition* 47, 10, 3343–3361. DOI: https://doi.org/10.1016/j.patcog.2014.04.018.

[14] Beeke E. Heldberg, Thomas Kautz, Heike Leutheuser, Rudiger Hopfengartner, Burkhard S. Kasper, and Bjoern M. Eskofier. 2015. Using wearable sensors for semiology-independent seizure detection - towards ambulatory monitoring of epilepsy. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference* 2015, 5593–5596. DOI: https://doi.org/10.1109/EMBC.2015.7319660.

[15] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. 2017. *The Kinetics Human Action Video Dataset*.

[16] P. Kotagal, A. Bleasel, E. Geller, P. Kankirawatana, B. I. Moorjani, and L. Rybicki. 2000. Lateralizing value of asymmetric tonic limb posturing observed in secondarily generalized tonic-clonic seizures. *Epilepsia* 41, 4, 457–462. DOI: https://doi.org/10.1111/j.1528-1157.2000.tb00189.x.

[17] Eva Lehner-Baumgartner, Simone Geiblinger, and Christoph Baumgartner. 2011. Epilepsien. In *Klinische neuropsychologie. Grundlagendiagnostik - rehabilitation*, W. Pies, Ed. Springer, [Place of publication not identified], 357–374. DOI: https://doi.org/10.1007/978-3-7091-0064-6_26.

[18] Frans S. S. Leijten. 2018. Multimodal seizure detection: A review. *Epilepsia* 59 Suppl 1, 42–47. DOI: https://doi.org/10.1111/epi.14047.

[19] Diogo C. Luvizon, David Picard, and Hedi Tabia. 2018. *2D/3D Pose Estimation and Action Recognition using Multitask Deep Learning*.

[20] Theodor W. May and Carsten W. Israel. 2019. Plötzlicher unerwarteter Tod bei Epilepsie (SUDEP) : Epidemiologie, kardiale und andere Risikofaktoren. *Herzschrittmachertherapie & Elektrophysiologie* 30, 3, 274–286. DOI: https://doi.org/10.1007/s00399-019-00643-0.

[21] Theodor W. May and Margarete Pfäfflin. 2006. Epidemiologische Daten zum plötzlichen, unerklärbaren Tod bei Epilepsie. *Z. Epileptol.* 19, 2, 60–70. DOI: https://doi.org/10.1007/s10309-006-0193-1.

[22] S. Noachtar, F. Rosenow, S. Arnold, C. Baumgartner, A. Ebner, H. Hamer, H. Holthausen, H. J. Meencke, A. Müller, A. C. Sakamoto, B. J. Steinhoff, I. Tuxhorn, K. J. Werhahn, P. A. Winkler, and H. O. Lüders. 1998. Die semiologische Klassifikation epileptischer Anfälle. *Der Nervenarzt* 69, 2, 117–126. DOI: https://doi.org/10.1007/s001150050247.

[23] Petra Ott-Ordelheide. 2019. Epilepsie: Formen und Klassifikationen. *Heilberufe* 71, 3, 25–28. DOI: https://doi.org/10.1007/s00058-019-0019-y.

[24] M. Pediaditis, M. Tsiknakis, P. Vorgia, D. Kafetzopoulos, V. Danilatou, and D. Fotiadis. 2010. Vision-based human motion analysis in epilepsy - Methods and challenges. In *Information Technology and Applications in Biomedicine (ITAB), 2010 10th IEEE International Conference on*. IEEE, 1–5. DOI: https://doi.org/10.1109/ITAB.2010.5687733.

[25] Matthew Pediaditis, Manolis Tsiknakis, and Norbert Leitgeb. 2012. Vision-based motion detection, analysis and recognition of epileptic seizures--a systematic review. *Computer methods and programs in biomedicine* 108, 3, 1133–1148. DOI: https://doi.org/10.1016/j.cmpb.2012.08.005.

[26] Markus Reuber and Christian E. Elger. 2003. Psychogenic nonepileptic seizures: review and update. *Epilepsy & behavior : E&B* 4, 3, 205–216. DOI: https://doi.org/10.1016/S1525-5050(03)00104-5.

[27] Gregory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. 2020. LCR-Net++: Multi-Person 2D and 3D Pose Detection in Natural Images. *IEEE transactions on pattern analysis and machine intelligence* 42, 5, 1146–1161. DOI: https://doi.org/10.1109/TPAMI.2019.2892985.

[28] Jing Tian, Weiyu Yu, Jinquan Chen, Junke Lin, Mingfeng Wen, Yingxin Li, Jianxin Zhong, Keqiang Chen, and Xuchu Feng. 2020 - 2020. Automated Analysis of Seizure Behavior in Video: Methods and Challenges. In *2020 2nd World Symposium on Artificial Intelligence (WSAI)*. IEEE, 34–37. DOI: https://doi.org/10.1109/WSAI49636.2020.9143279.

[29] Hong-Bo Zhang, Yi-Xiang Zhang, Bineng Zhong, Qing Lei, Lijie Yang, Ji-Xiang Du, and Duan-Sheng Chen. 2019. A Comprehensive Survey of Vision-Based Human Action Recognition Methods. *Sensors (Basel, Switzerland)* 19, 5. DOI: https://doi.org/10.3390/s19051005.

Evaluation eines Algorithmus zum Upscaling durch Single-Image Verfahren anhand von Nummernschildern

Mario Werner
Hochschule Reutlingen
Mario.Werner@Student.Reutlingen-University.de

Abstract

Der Bereich Single-Image Super-Resolution, mit dem Bilder auf eine höhere Auflösung skaliert werden können, zeigt viel Potential, um damit Bilder oder Videos schärfer aussehen zu lassen. Es gibt dabei viele Anwendungsmöglichkeiten, welche auch heute schon genutzt werden, wie z. B. der Multimediabereich, welcher alte Filme in einer besseren Auflösung abspielen lassen kann oder Urlaubsbilder, die nach dem Upscaling schärfer aussehen. In dieser Ausarbeitung wird die Frage geklärt, ob es möglich ist, Single-Image Super-Resolution im Bereich der automatisierten Kennzeichenerkennung zu nutzen und ob die überarbeiteten Bilder dabei helfen, die Erkennung zu vereinfachen. Dabei wird ein Versuch durchgeführt mit der ein Upscaling Algorithmus Bilder mit Kennzeichen verbessert und ein weiterer Algorithmus versucht, diese Kennzeichen automatisiert zu erkennen. Ziel ist es, die automatisierte Erkennung noch weiter zu verbessern und die Trefferrate der erkannten Kennzeichen auf den Bildern zu erhöhen.

CCS Concepts

Computing methologies → Image manipulation

Keywords

Algorithm, Classification, Super-Resolution, Single-Image, Neuronal Network, License Plates, Upscaling

1 Einleitung

Neuronale Netze (NN) sind mittlerweile schon fast in allen Bereichen angekommen. Der Vorteil von Machine Learning und deren Lernfähigkeit wird in immer mehr Bereichen eingesetzt. Darunter auch die automatische Erkennung von Mustern in Bildern. Die Klassifizierung von Objekten kann nützlich sein für z. B. das autonome Fahren, Gesichtserkennung oder in Produktionsstätten, durch welche die Maschinen die Objekte eigenständig erkennen können. Das Potential ist bereits riesig und wächst weiter.

Die Herausforderung, welche die neuronalen Netze überwinden müssen, ist in diesem Fall das Upscaling von Bildern. Das bedeutet, dass Bilder mit einer schlechten Auflösung durch die künstliche Intelligenz (KI) verbessert werden. Im Folgenden wird das Up-Downscaling genannt. Dabei ist darauf zu achten, dass nicht nur die Anzahl der Pixel erhöht wird, sondern dass der Inhalt auf dem Bild auch beibehalten wird. Die Schwierigkeit dabei ist, dass die Pixelfarbe, welche mit den Pixeln hinzugefügt wird, geschätzt werden muss. Orientieren kann sich die KI an

---

Betreuer Hochschule: Prof. Dr. Uwe Kloos
Hochschule Reutlingen
Uwe.Kloos@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

den darum liegenden Nachbar Pixeln, welche im Normalfall einen ähnlichen Farbwert wiedergeben.
Upscaling kann auf Bilder und Videos angewendet werden. Auch wenn Videos im Gegensatz zu einzelnen Bildern mehr Informationen beinhalten, beschränkt sich der Versuch auf die einzelnen Bilder. Das hat zum einen den Vorteil, dass die Anwendung dadurch in mehr Szenarien genutzt werden kann, da die Auswertung auf einzelnen Bildern auch in Videos genutzt werden kann. Zum anderen sollen gezielt Szenarien dafür in Frage kommen, welche nur ein Bild zur Verfügung haben, wie z. B. bei Radarfallen. Die Berechnungen beider Methoden funktionieren in Echtzeit, auch wenn die zweite Möglichkeit entsprechend mehr Rechenleistung und damit auch Zeit braucht. Die häufigste Anwendung heutzutage ist im Bereich der Medien zu finden. Dabei wird bei älteren Filmen oder allgemein Bildmaterial, das nicht die volle Auflösung unterstützt, die Qualität verbessert. Dadurch können auch noch alte Filme in hoher Auflösung angesehen werden, ohne dass diese in dieser Auflösung aufgenommen wurden.

Angesichts der Tatsache, dass die Nummernschilderkennung ein entscheidender Baustein der Strafverfolgung darstellt, soll mithilfe dieser Arbeit evaluiert werden, ob ein Upscaling der Bilder vor der Kennzeichenerkennung die Trefferrate erhöhen kann. Ziel dabei ist es, die automatisierte Erkennung von Kennzeichen zu verbessern, bzw. die Frage zu klären, ob es der Auswertung der Kennzeichen weiterhilft, wenn den Bildern weitere Pixel hinzugefügt werden. Dadurch wird die Upscaling-KI in der Anwendung der Kennzeichen getestet und herausgefunden, ob diese in diesen Bereich einen Nutzen findet.

In dieser Arbeit wird deswegen der Kennzeichentext als Merkmal der korrekten Detailrekonstruktion genutzt. Dabei dürfen die Zahlen und Buchstaben der Kennzeichen nicht einfach nur richtig oder qualitativ hochwertig aussehen, sondern vor allem müssen diese, dem Original übereinstimmen. Sollten diese von den eigentlichen Zeichen abweichen, würde ein anderer Text auf dem Bild ermittelt werden und zu Fehlern bei der Erkennung der Kennzeichen führen.

1.1 Upscaling Single-Image

Für das Single-Image-Upscaling, auch Super-Resolution genannt, gibt es mehrere Verfahren [1, 3, 5–7]. Die Vorgehensweise der Autoren ist dabei ähnlich. Es wird ein neuronales Netz erstellt und trainiert. Die Unterschiede liegen jeweils im Aufbau der Netze und in den Trainingsdaten, welche dafür verwendet wurden. Da die neuronalen Netze nur so gut sind, wie die Bilder, welche zum Training verwendet wurden, wird versucht aus vielen Unterschiedlichen Bereichen Bilder zu nutzen. Damit wird versucht einen möglichst großen Trainingsbereich abzudecken, um das Netz nicht anfällig für Fehler zu machen, welche auf Grenzfälle von Bildern beruhen. Darunter zählen dann Besonderheiten in Bildern, welche das Netz zuvor nicht gesehen, bzw. gelernt hat und anhand davon eine falsche Entscheidung treffen kann.

Das Prinzip ist, das Bild, welches aus einer deutlich schlechteren Qualität besteht, mit weiteren Pixeln zu füllen, um damit nicht nur die Anzahl der Pixel zu erhöhen, sondern um auch die Details hervorzuheben. Dadurch sehen die Bilder schärfer und detailreicher aus. Dieses Verfahren wird auch in diesem Versuch von Nutzen sein, da die folgenden Algorithmen, welche danach auf die Bilder angewendet werden können, davon profitieren. Der Versuch wird später genauer beschrieben. Die Upscaling-Algorithmen bestehen in den meisten Fällen aus neuronalen Netzen (NN), welches sich das Wissen über die Pixel in den Bilder während des Trainings selbstständig aneignen können. Jedoch gibt es dabei Unterschiede im Aufbau und zum anderen in der Trainingsweise der NN. Das beeinflusst das Verhalten und damit das Ergebnis der NNs erheblich [3].

Fehler, die dabei entstehen könnten, sind z. B. Pixel, die zwar in dem Bild sinnvoll aussehen, jedoch nicht korrekt sind.

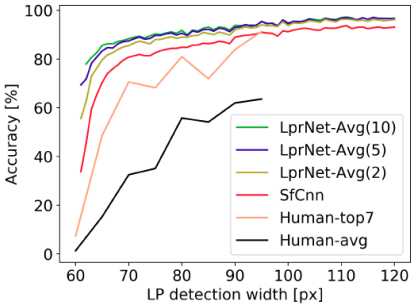

Abbildung 1: Vergleich der Erkennung von Kennzeichen zwischen Mensch und neuronalen Netzen [11]

Bei einer gleichmäßigen Oberfläche, wie einer Wand oder Gras, macht das nahezu keinen Unterschied, da sich die Muster immer wiederholen. Nur wenn ein neues Objekt oder Muster auftaucht, dass sich von den umherliegenden unterschiedet, kommt es auf die Details an.

1.2 Themenkontext

Die Methode, Bilder in eine höhere Auflösung zu konvertieren, gibt es allgemein schon in vorherigen Projekten und das wurde auch schon in einigen Versuchen unter Beweis gestellt. Auch in der Quelle [11] wird ein neuronales Netz genutzt, um Videos bzw. Bilder, in eine höhere Auflösung zu konvertieren. In dieser Arbeit wurde auch ein Vergleich zwischen der Erkennung von Kennzeichen durch Personen und einem Algorithmus durchgeführt [11].

In Abbildung 1 wird das Ergebnis dieses Vergleichs durch ein Diagramm verdeutlicht. Zu sehen ist, dass jede Umsetzung der NNs besser abschneidet als die Personen in dem Test [11].

2 Herausforderungen

Das Upscaling von Bildern durch neuronale Netze gibt es zwar schon eine Weile und diese erzielen in vielen Fällen auch gute Ergebnisse, jedoch ist es gerade bei Ziffern und kleineren Details sehr schwer diese schärfer darzustellen.

Abbildung 2: Originalbild links im Vergleich mit einem falschen Schriftzug des Upscalings rechts [10]

Dabei handelt es sich in den meisten Fällen um Bereiche in den Bildern, welche nicht erlernt werden können.

Der große Vorteil von den neuronalen Netzen ist, durch die Training-Phase neue Möglichkeiten zu einem bestimmten Teilbereich zu erlernen. Bei diesem Versuch werden vor allem auch die Bereiche der Bilder getestet, welche nicht direkt erlernt werden können.

Das neuronale Netz ist weder auf Kennzeichen noch auf Fahrzeuge spezialisiert und funktioniert damit vergleichbar wie andere NN. Das beschriebe Problem wird unter anderem in Abbildung 2 deutlich gemacht.

Dabei ist die Rückseite eines Fahrzeugs sichtbar, welche einen Schriftzug aufweist. Bei Betrachtung des Fahrzeugs wird sichtbar, dass dieses bei dem Upscaling gut gelungen ist und nur wenig Unterschiede festgestellt werden können. Der Lack und auch der Hintergrund sind größtenteils identisch. In den blauen Kreisen sind jeweils die Unterschiede sichtbar gemacht, um welche es unter anderem in dieser Arbeit geht. Auch wenn die beiden Bilder sich auf den ersten Blick ähnlich sehen, sind gerade die kleinen Details schwierig für den Algorithmus. Das Erkennen von Kennzeichen erfordern aber genau diese Details auf den Kennzeichen, damit diese korrekt erkannt werden. Ein weiteres Beispiel wird in Abbildung 9 gezeigt. Dabei wird das Problem bei einem Kennzeichen sichtbar, welches nicht erkannt werden kann, auch wenn das restliche Bild gut gelungen ist und für das menschliche Auge identisch aussieht.

3 Auswahl des Algorithmus

Bei der Auswahl des Algorithmus wird sich an einem Benchmark [8] orientiert, welcher unterschiedliche NN getestet hat und die Ergebnisse mit einer Score versehen hat.

Es werden dabei die besten NNs in eine engere Auswahl genommen und getestet. Das gewählte NN hat den Vorteil gegenüber den herkömmlichen, die Details nicht nur so aussehen zu lassen, als ob diese echt wären. Gerade bei Texturen wird das oft angewendet, da der Unterschied nicht auffällt und diese realitätsnah nachgebildet werden. Da es sich dabei um Quellcode offene Algorithmen handelt, gibt es keine Garantie, dass diese auch wirklich so funktionieren wie angegeben. Einige waren auf den zu testenden Computerumgebungen nicht lauffähig und wurden deswegen aussortiert. Das gewählte NN der Autoren Zheng Hui, Jie Li, Xinbo Gao und Xiumei Wang [3] ist in diesem Fall auch Quellcode offen und wurde durch die Github-Seite, bzw. ein eine eigene Arbeit darüber, dokumentiert [2, 3].

4 Erläuterung des Algorithmus

Der für diesen Versuch genutzte Algorithmus ist aus der Quelle [3] und unter dem folgenden Github-Link erreichbar [2]. Dieser besteht aus einem Progressive Perception-Oriented Network (PPON).

In Abbildung 3 wird die Architektur des Modells aufgezeigt, welches sich nach dem Peak signal-to-noise ratio (PSNR) richtet. Zu Beginn wird das Bild, welches eine höhere Auflösung bekommt, in das NN gegeben.

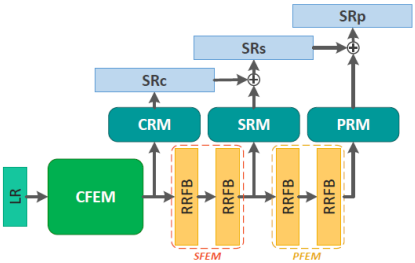

Abbildung 3: Architektur des PPON [2]

Das wird mit dem LR-Block dargestellt. Es werden in diesem Modell 24 Residual-in-Residual Fusion Blocks (RRFB) verwendet. Diese werden alle nacheinander durchlaufen und mit der Faltung des ursprünglichen Bildes Elementweise addiert.

Das wird durch das kleine Plus Symbol dargestellt. Am Ende kommt dabei dann die Super-Resolution heraus, also das Bild mit der besseren Auflösung [3]. Die Abkürzungen stehen dabei für:

- CRM = Content reconstruction module
- SRM = Structural reconstruction module
- PRM = Photo-realism reconstruction module
- LR = Low resolution
- SR = Super resolution

Die einzelnen Schritte werden dabei einzeln berechnet und als Bild ausgegeben. Die Ausgaben davon, SRc und SRs, werden dann Element für Element mit der Ausgabe des PRM addiert [3]. Die Ausgabe ist dann das fertige Bild mit der gewünschten Auflösung. Das Verfahren kann das Bild entweder in der Auflösung verdoppeln oder vervierfachen.

5 Auswahl der Daten

Bei der Auswahl der Daten wird auf eine entsprechende freie Lizenz geachtet, da es sich bei den Kennzeichen um sensible Daten handelt. Die in der Einleitung erwähnten Bilder von Radarfallen werden nicht verwendet, da diese nicht unter einer freien Lizenz verfügbar sind.

Es gibt zwar eine große Auswahl an Datenbanken mit Kennzeichen, jedoch sind davon nicht alle für die Auswertung geeignet. Die meisten Bilder in diesen Datenbanken haben nur das Kennzeichen auf dem gesamten Bild, was für das Training, nicht aber für diese Auswertung nützlich ist. Hintergrund dazu ist, dass eine reale Situation mit den Bilder simuliert werden soll.

Abbildung 4: Unterschied zu anderen Kennzeichen mit dem Schriftzug der Provinz [10]

Gerade bei Aufnahmen im Straßenverkehr bei fahrenden oder parkenden Fahrzeugen werden die Bilder nicht ausschließlich von den Kennzeichen geschossen. Das ist deshalb wichtig, da es bei den Bildern nicht direkt auf die Auflösung ankommt, um die es in dieser Arbeit hauptsächlich geht, sondern nur um die Pixel, welche das Nummernschild abbilden. Bilder mit einer höheren Auflösung bilden zwar das Kennzeichen auch hochauflösender ab, sollte das Fahrzeug hingegen weiter von den Kamera entfernt sein, wird die Auflösung des Nummernschild schlechter (weniger Pixel), die der Aufnahme bleibt dabei jedoch gleich. Es wird bei der Auswahl der Bilderdatenbank deshalb darauf geachtet, einen gewissen Abstand zu dem Fahrzeug zu halten, bzw. die Blickwinkel der Kamera zu dem Kennzeichen, welche variieren sollten.

Für diese Arbeit wird ein Datensatz [10, 12] mit chinesischen Kennzeichen gewählt. Das Land spielt zwar keine Rolle, jedoch sind die Kennzeichen je nach Land unterschiedlich. In diesem Teil der Bilderdatenbank befinden sich ca. 5000 Bilder, wobei nur ein Teil davon verwendet wird. Für diesen Versuch werden die ersten 181 Bilder verwendet. Die in der Quelle verwendete Datenbank wurde auch von den Herausgebern in einem eigenen Algorithmus verwendet, wobei dieser in dieser Arbeit nicht genutzt wird und der Verweis sich dabei auf die Bilder beschränkt. Die Bilder haben eine einheitliche Auflösung von 720 x 1160 Pixeln. Die Bilder wurden von parkenden Fahrzeugen aufgenommen. In Abbildung 4 ist der Unterschied der Kennzeichen abgebildet.

In dem rot markierten Bereich befinden sich noch ein Schriftzug, welcher auf jedem Kennzeichen vorhanden ist und die Provinz angibt. Das ist insofern relevant, da diese Zeichen oft fälschlicherweise als Ziffer auf dem Kennzeichen erkannt werden. Gerade feine Details in Bilder mit einer geringen Auflösung sind schwer zu erkennen und führen dadurch auch oft zu Fehlern in der Erkennung oder wie in diesem Fall, zu Fehlern in dem Upscaling. Der Rest des Kennzeichens besteht, wie europäische Kennzeichen auch, aus Buchstaben und Zahlen.

Der zweite Datensatz [4] beschränkt sich nicht auf ein Land und beinhaltet Kennzeichen aus unterschiedlichen Ländern. Es gibt bei dem Vorgehen der Aufnahmen zwischen den Datensätzen keinen Unterschied.

Der Algorithmus [3] selbst ist nicht für eine bestimme Art von Länderkennzeichen entworfen. Es gibt also weder einen Vorteil noch einen Nachteil bei den Kennzeichen unterschiedlicher Länder

6 Durchführung Versuch

Der allgemeine Ablauf sieht wie folgt aus:

1. Die Daten werden so vorbereitet, dass diese für den Versuch genutzt werden können. Nicht erkennbare oder verschwommene Bilder werden aussortiert. Des Weiteren wird die Auflösung auf die gewünschte Pixelanzahl angepasst.
2. In dem zweiten Schritt wird getestet, ob die Bilder bereits von dem Algorithmus erkannt werden. Sollte das der Fall sein, werden diese aussortiert. Der Algorithmus darf die Bilder im Voraus nicht erkennen, da das erst nach dem Upscaling möglich sein darf.
3. In dem folgenden Schritt findet das Upscaling statt. Dabei wird die Pixelanzahl erhöht, um so dem ALPR-Algorithmus das Erkennen der Kennzeichen zu ermöglichen.

4. In dem letzten Schritt wird die endgültige ALPR durchgeführt. Dabei werden alle Bilder überprüft und nach möglichen Kennzeichen gesucht. Die erfolgreich erkannten Kennzeichen werden von Hand geprüft und zusammengezählt.

Abbildung 5: Bildvergleich vor und nach dem Downscaling

6.1 Vorarbeit der Daten

Die Bilder werden auf eine geringere Auflösung reduziert. Da die Auflösung von 720 x 1160 Pixeln schon so hoch ist, dass darauf das Kennzeichen problemlos erkennbar ist, werden diese auf 10 % der Pixel in dem Bild reduziert. Das heißt das ganze Bild hat nur noch 10 % der Pixel des ursprünglichen Bildes und damit eine Auflösung von 72 x 116 Pixeln. In diesem Fall werden die 10 Pixeln in dem hochaufgelösten Bild nur noch durch einen Pixel dargestellt. Das Bild wird dadurch deutlich kleiner, wenn es betrachtet wird, sollte man hineinzoomen, wird es sichtbar unscharf. Das Prinzip des Downscalings wird in Abbildung 5 verdeutlicht. Dabei wird die Auflösung von vier auf einen Pixel verringert. Das wäre in diesem Fall ein Scaling auf 25 % des ursprünglichen Bildes. Da in diesem Verfahren kein Filter angewendet wird, wird einer der Pixel ausgewählt und als neuer Pixel in das kleinere Bild eingetragen. In diesem Versuch wird kein Filter verwendet, da dieser schon von dem Folgealgorithmus durchgeführt wird.

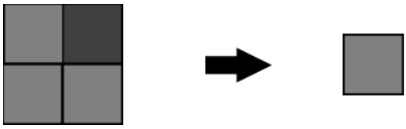

Abbildung 6: Prinzip des Downscalings auf 25 %

Die Idee dahinter ist, die Situation mit einem weit entfernten Fahrzeug oder einer schlechten Kamera zu simulieren.

Auch wenn in realen Situationen die Bedingungen besser sind was das gesamte Bild angeht, so wird das Kennzeichen nur durch ein paar der Pixel dargestellt. Das Kennzeichen ist in dieser Auswertung der relevante Teil des Bildes, da die anderen Objekte darum keine Rolle spielen. Bei Werten über den 10 % war das Problem, dass die Bilder, bzw. die Kennzeichen darauf schon vor dem Upscaling erkannt werden. Das darf nicht sein, da sonst das Upscaling keinen Sinn mehr ergibt. Ein Wert unter den 10 % war von der Qualität des Bildes, also der Anzahl der Pixel so schlecht und ungenau, dass darauf nichts mehr erkannt werden konnte. Die 10 % sind damit das Minimum, welches der Algorithmus noch verarbeiten kann. Da die Bilder aber sogar noch unter der Full-HD (1080 x 1920) Auflösung liegen, ist eine Reduzierung auf 10 % und die damit erreichte Auflösung für die heutige Zeit niedrig genug für diesen Versuch. In Abbildung 6 wird der Effekt sichtbar gemacht. Das Bild mit der geringen Auflösung, in diesem Fall dann die 72 x 116 Pixel, ist deutlich kleiner. Die Seiten sind dadurch nur noch ein Zehntel so lang wie in dem Originalbild, welches im oberen Bildbereich zu sehen ist. Um die Qualität besser darzustellen wird das Bild auf die gleiche Größe wie das Original vergrößert, indem die Pixel angepasst werden. Aus geringer Distanz ist damit das Kennzeichen auch mit dem menschlichen Auge nur noch schwer zu erkennen. Die Anzahl der Pixel, welche ein Kennzeichen abbilden ist dabei nicht in allen Bildern identisch. Je nach Entfernung und Winkel zu dem Kennzeichen wird dieser unterschiedlich groß dargestellt.

6.2 Erste Auswertung durch Automatic License Plate Recognition

Die Automatic Lincese Plate Recognition, kurz ALPR, ist ein Algorithmus zur automatischen Erkennung von Kennzeichen.

Dabei wird ein neuronales Netz aufgesetzt und mit Bilder von Kennzeichen trainiert. Um Kennzeichen auf den Bildern bei einem trainierten Netz zu erkennen, wird ein Bild eines Fahrzeugs mit sichtbarem Kennzeichen in das Netz gegeben und automatisch ausgewertet. In dieser Arbeit wird Sighthound [9] als Erkennung der Nummernschilder verwendet. Dabei handelt es sich um ein kommerzialisiertes Produkt, welches zu einem gewissen Grad kostenfrei genutzt werden kann. Es handelt sich dabei um ein bereits trainiertes neuronales Netz, welches die Bilder automatisch auswertet. Es wird nicht ersichtlich wie das neuronale Netz aufgebaut ist, das spielt in dieser Arbeit aber auch keine wichtige Rolle, da das Netz sich bei den Bildern immer gleich verhält und nicht durch den Nutzer weiter trainiert wird. Das ist die Erkennung oder auch die Möglichkeit, dass das Kennzeichen nicht erkannt wird und die damit mögliche Einteilung, ob ein Kennzeichen auf einem Bild erkannt wird oder nicht. Dieser Algorithmus dient hierbei als Referenz und als objektive Beurteilung der Qualität der verbesserten Bilder. Sollten diese durch eine Person ausgewertet werden, wäre dass durch die subjektive Meinung nicht sehr aussagekräftig. Die Auswertung durch den Algorithmus entscheidend bei den gleichen Bildern auch immer gleich und ist damit besser zur Beurteilung geeignet. Das Ergebnis besteht aus einer von zwei Möglichkeiten. Entweder wird das Kennzeichen erkannt oder es wird keins erkannt. Ein falsch erkanntes Kennzeichen, das zwar den Großteil der Ziffern erkannt hat, gilt auch als nicht erkannt. Es werden nur vollkommen richtig erkannte Kennzeichen als solche eingestuft. Nachdem die Bilder eine geringere Auflösung haben, werden diese zunächst ohne das Upscaling getestet. Sollten diese zu diesem Zeitpunkt schon korrekt erkannt werden, werden diese aus der Auswertung entfernt und durch andere Bilder ersetzt. Der Hintergrund dazu ist, dass es zu diesem Zeitpunkt keine Rolle spielt ob der Algorithmus das Bild, bzw. die Auflösung verbessern kann, da es ab diesem Punkt kein verbessern mehr gibt. Es werden für diesen Versuch nur Bilder genutzt, welche Kennzeichen beinhalten, die nicht schon zuvor von dem Algorithmus erkannt werden. Sollten die Kennzeichen falsch erkannt worden sein, werden sie in diesem Kontext auch genutzt.

6.3 Upscaling der Bilder

Der bereits beschriebe Algorithmus in dem Kapitel “4 Erläuterung des Algorithmus“ wird an diesem Punkt auf die Bilder angewendet. Die Bilder haben alle die gleiche Auflösung und werden durch den Algorithmus überarbeitet. Der Scaling-Faktor beträgt bei allen Bildern 4. Das heißt das Bild wird nach der Überarbeitung viermal so viele Pixel beinhalten.

6.4 Zweite Auswertung durch Automatic License Plate Recognition

Der zweite Durchlauf erfolgt nach dem Downscaling der Bilder. Es werden dann alle Bilder, welche davor nicht schon von dem Algorithmus erkannt werden, in den Algorithmus geladen. Die Bilder haben danach eine Auflösung von 288 x 464 Pixeln. Durch die erhöhte Auflösung wird versucht dem Algorithmus das Erkennen der Kennzeichen zu ermöglichen. Werden die Kennzeichen auf Bilder nach dem Upscaling erkannt und waren davor für den Algorithmus nicht sichtbar, so hat das Upscaling dies ermöglicht. Die erkannten Bilder werden dann von Hand geprüft. Das Referenzbild ist dabei das Originalbild in voller Auflösung. Sollte auch nur eine Ziffer falsch sein, werden diese auch hier als nicht erkannt einsortiert. Die erkannten Bilder werden gezählt und der Prozentsatz der erkannten Bilder abhängig aller Bilder für die Auswertung berechnet.

7 Ergebnisse

Da es unter anderem das Ziel dieser Arbeit ist, die Erfolgschance dieser Algorithmen weiter zu verbessern, sollten entsprechend die Ergebnisse dazu beitragen, auch nicht erkennbare Bilder auswerten zu können. Das neuronale Netz dieses Anbieters erlangt nach Hersteller bereits eine Trefferrate von 99% [9], diese kann aber noch weiter verbessert werden. Die Anwendung zielt dabei auf die verbleibenden 1 % ab, welche nicht erkannt wurden und versucht, auch diese noch erkennbar zu machen. Nach der Analyse von 181 Bildern kam folgendes bei dem ersten Datensatz heraus. Es wurden 92 Bilder erfolgreich nach dem Upscaling erkannt, was ca. 51 % der Bilder ausmacht. Damit wird ungefähr jedes zweite Bild, welches nicht erkannt wurde, durch das Upscaling erkennbar. Bei dem zweiten Datensatz wurden von 64 Bildern 43 richtig erkannt, was 67 % ausmacht.

In Abbildung 8 werden alle drei Auflösungsstufen sichtbar. Im zweiten Teil des Bildes ist das Ergebnis des Upscalings sichtbar und es sind auch deutlich mehr Details erkennbar. Die allgemeine Auflösung hat sich erhöht, jedoch werden bei genauem Betrachten kleine Artefakte sichtbar. Die Zeichen ganz links auf dem Bild sind nicht mehr lesbar und können zu keinem sinnvollen Symbol zusammengesetzt werden. Das ist in diesem Fall kein Problem, da der Algorithmus ohnehin nur die Ziffern danach erkennen muss. Des Weiteren sind kleine Bildfehler bei der ersten Acht sichtbar. Bei einer sehr schlechten Auflösung ist die Trennung zwischen zwei Zeichen oft nicht gut sichtbar, sodass es für den Algorithmus so aussieht, als ob es keine klare Trennung gibt. Das wird im Ergebnis fälschlicherweise mit verschmolzenen Symbolen interpretiert, ähnlich wie in der Abbildung 8 in dem mittleren Bildbereich. In diesem Fall hat die Erkennung trotzdem funktioniert und auch für das menschliche Auge ist ein klares Kennzeichen mit allen Ziffern sichtbar. Das erkannte Kennzeichen, so wie die einzelnen Ziffern werden im

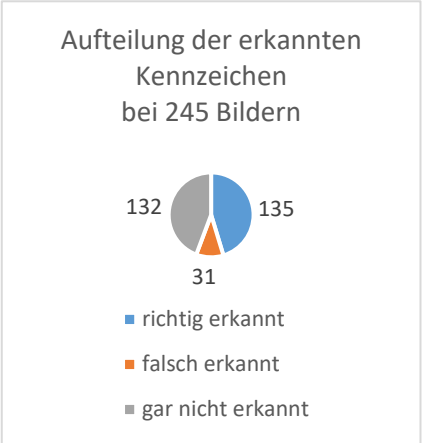

Abbildung 7: Aufteilung der Ergebnisse der Kennzeichenerkennung

unteren Bildrand durch das Programm eingetragen und die Position markiert.

Da es aber nur ungefähr bei der Hälfte der Bilder funktioniert hat, wird in Abbildung 9 der Fall aufgezeigt, bei welchem es nicht funktioniert hat. Im oberen Bildbereich gibt es bei dieser Auflösung nahezu keine Unterschiede mehr. Das Fahrzeug, welche abgebildet wird, sieht dem Original was die Karosserieteile angeht, zum Verwechseln ähnlich. Bei genauerem Hinsehen, gerade bei dem Kennzeichen, werden aber die Artefakte sichtbar. Diese sind in der Abbildung 9 im unteren Bildrand sichtbar. Das neuronale Netz hat sich bei dem Training die Fahrzeuge angesehen und kann diese Detailgetreu nachbilden. Die Form der Karosserie unterscheidet sich von Fahrzeug zu Fahrzeug nur leicht und auch die Farbe und Oberfläche des Lacks bleibt bei einem Modell immer ähnlich. Bei dem Kennzeichen hingegen ist es schwieriger diese Nachzustellen, da diese niemals gleich sind. Auch wenn das neuronale Netz alle vorherigen Kennzeichen kennen würde, wäre es trotzdem nicht das gesuchte, welches in diesem Bild dargestellt werden muss.

Abbildung 8: Alle drei Stufen der Auflösung im Vergleich

8 Weitere Anwendungsfelder

Durch das Überarbeiten der Auflösung können weitere Anwendungsfelder in Betracht gezogen werden. Das kann von einfachen Apps, welche Notizen einscannen können, bis hin zu dem Scannen von Dokumenten reichen, die dann in ein anderes Format umgewandelt werden. Ein weiteres Anwendungsfeld, das an sich in der Umsetzung noch umstritten ist, aber vereinzelt in Ländern [1] schon genutzt wird, ist das automatisierte Erkennen von Gesichtern. Durch den Vorteil, welcher in dieser Arbeit aufgezeigt wird, also das Erkennen von Details in qualitativ schlechten Bildern, können durch Kameras auch bei schlechten Wetter-/Kamerabedingungen Personen darauf erkannt werden. Dabei ist es sehr wichtig, die richtigen Personen zu identifizieren, da es sonst zu großen Problemen kommen kann und womöglich die falschen Personen identifiziert werden.

Auch im Bereich des Restaurierens kann ein solches Vorgehen hilfreich sein. Da sehr alte Bilder, welche z. B. eingescannt werden, nicht über eine gute Auflösung verfügen, können diese noch schärfer werden. Dabei sollte darauf geachtet werden, ob die Bilder nicht schon Schäden aufweisen, denn diese würden für noch mehr Artefakte auf den Bildern sorgen. Sollte das Upscaling zu oft auf einem Bild angewendet werden, kann es auch sein, dass sich der Inhalt auf dem Bild etwas ändert.

9 Fazit

Das Ergebnis hat gezeigt, dass ungefähr bei der Hälfte aller Bilder das Upscaling erfolgreich war.

Abbildung 9: Vergleich Original und Upscaling

Um ein solches Verfahren in einem realistischen Szenario nutzen zu können, müssten Tests zur Echtzeitfähigkeit gemacht werden, damit diese auch erfolgreich bei einem Video genutzt werden können. Bei einzelnen Bildern, wie es auch in diesem Versuch der Fall war, funktioniert das auch ohne Echtzeitfähigkeit. Ob auch Fahrzeugkennzeichen ermittelt werden können, wenn diese sich bei der Aufnahme bewegt haben ist fraglich, da verschwommene Bilder gerade bei diesen extrem niedrigen Auflösungen nur sehr schwer erkannt werden können. Oftmals sind verschwommene Bilder allein schon eine

Herausforderung für solche Algorithmen, diese Bilder dann auch noch mit einer sehr niedrigen Auflösung vorliegen zu haben macht es vermutlich nahezu unmöglich die Kennzeichen zu erkennen. Das müsste jedoch erst durch einen eigenen Test ermittelt werden. Zusammenfassend ist zu sagen, dass der Test für dieses Einsatzgebiet erfolgreich war und damit einen großen Vorteil für die automatisierte Erkennung von Kennzeichen bietet.

Literaturverzeichnis

[1] cnet. How China uses facial recognition to control human behavior. Retrieved October 12, 2020 from https://www.cnet.com/news/in-china-facial-recognition-public-shaming-and-control-go-hand-in-hand/.

[2] Zheng Hui, Jie Li, Xinbo Gao, and Xiumei Wang. PPON-Github. Retrieved October 12, 2020 from https://github.com/Zheng222/PPON.

[3] Zheng Hui, Jie Li, Xinbo Gao, and Xiumei Wang. 2020. Progressive Perception-Oriented Network for Single Image Super-Resolution.

[4] Zoran Kalafatić. 2003. Bilderdatenbank (2003). Retrieved from http://www.zemris.fer.hr/projects/LicensePlates/hrvatski/rezultati.shtml.

[5] Mario König, Martin Mišiak, and Arnulph Fuhrmann. 2019. Perceptual Comparison of Four Upscaling Algorithms for Low-Resolution Rendering for Head mounted VR Displays.

[6] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe S. Twitter. 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network.

[7] Frank Lin, Clinton Fookes, Vinod Chandran, and Sridha Sridharan. 2007. Super-Resolved Faces for Improved Face Recognition from Surveillance Video.

[8] Papers with Code. Benchmark. Retrieved October 12, 2020 from https://paperswithcode.com/sota/image-super-resolution-on-set5-4x-upscaling.

[9] Sighthound. Automatic License Plate Recognition. Retrieved October 11, 2020 from https://www.sighthound.com/products/alpr.

[10] tomorrow1210 and detectRecog. Chinese City Parking Dataset. Retrieved October 7, 2020 from https://github.com/detectRecog/CCPD.

[11] Vašek Vojtech, Franc Vojtech, and Urban Martin. 2018. License Plate Recognition and Super-resolution from Low-Resolution Videos by Convolutional Neural Networks.

[12] Xu Zhenbo, Yang Wei, Meng Ajin, Lu Nanxue, Huang Huan, Ying Changchu, and Huang Liusheng. 2018. Towards End-to-End License Plate Detection and Recognition: A Large Dataset and Baseline.

Virtual Reality als Tool für Kinder mit Autismus-Spektrum-Störung: Eine systematische Literaturrecherche

Elif Hizli
Hochschule Reutlingen
Elif.Hizli@Student.Reutlingen-University.de

Abstract

Kinder mit einer Autismus-Spektrum-Störung neigen dazu, schnell die Konzentration zu verlieren. Die Möglichkeit der Darstellung von veränderbaren, drei-dimensionalen Umgebungen in der virtuellen Realität, kann die Konzentrationszeit der Nutzer, durch eingesetzte Reize verlängern. Durch den Einsatz der virtuellen Realität kann ein individueller Assistent für die tägliche Betreuung und die Förderung der Lernbereitschaft für Kinder mit Autismus-Spektrum-Störung bereitgestellt werden. Um die Machbarkeit eines virtuellen Assistenten und die Akzeptanz der virtuellen Realität bei autistischen Kindern zu untersuchen, wird eine Forschung der Literatur und durchgeführten Studien vorgenommen. Die Auswahl, Identifizierung und Bewertung der relevanten Forschungsergebnisse wird nach den bevorzugten Reporting Items der systemischen Literaturrecherche durchgeführt.

Betreuer Hochschule: Prof. Dr. -Ing. Natividad Martinez
Hochschule Reutlingen
Natividad.Martinez@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

CCS Concepts

•Applied computing
~Education
~Interactive learning environments
•Human-centered computing
~ Human computer interaction (HCI)
~HCI design and evaluation methods
~User studies

Keywords

Autismus-Spektrum-Störung, Virtual Reality, Virtual Assistant, Systematic Review

1 Einleitung

Autismus-Spektrum-Störung ist eine neurologische Entwicklungsstörung, die im frühkindlichen Alter, mit achtzehn Monaten, bereits diagnostiziert werden kann. Jedoch kann die Diagnose aufgrund der Heterogenität der Symptome in verschiedenem Alter gestellt werden. Weltweit wird eine Prävalenz von 0.6%-1% angenommen [Umweltbundesamt, 2020]. Im Vergleich zu den letzten Jahren ist die Anzahl der Autisten gestiegen. Als Ursache für diesen Anstieg wird die aktuelle Entwicklung der Diagnosewerkzeuge gesehen [Umweltbundesamt, 2020]. Kerndefizite für diese Störung sind soziale Kommunikation, Interaktion und sich wiederholende Verhaltensmuster. Betroffene benötigen eine Unterstützung für Verhalten, Bildung Gesundheit und Freizeit [1]. Außerdem können Defizite bei Informationsverarbeitung über sensorische Modali-

täten, zum Beispiel Sehen (Geste) und Hören (Sprache), auftreten. Für die Diagnose gibt es Screening-Tools, womit das Verhalten der Kinder beobachtet und bewertet werden können. Jedoch reichen diese Tools für eine endgültige Diagnose nicht aus. Hierfür werden, bei auffälligen Verhaltensmustern, sowohl kognitive Tests als auch Sprachtest durchgeführt. Dies kann von einem Kinderarzt, Psychologen oder Neurologen durchgeführt und diagnostiziert werden. Voraussetzung hierfür sind ausreichende Kenntnisse über DSM-5 (Diagnostic and Statistical manual of mental disorders) [1]. Studien, welche in dieses Literaturrecherche präsentiert werden, berichten, dass die Therapie, dieser Störungen, mithilfe von Virtual Reality (VR) unterstützt werden kann. Die virtuelle Welt ist eine 3D Umgebung, in der für die Nutzer eine reale Welt erschaffen wird. Der Fokus wird allerdings auf eine Therapie, mithilfe von VR, auf die Emotionserkennung gelegt [14]. Mithilfe dieser Literaturanalyse wird untersucht, ob der Einsatz auch für Probleme im Alltag von autistischen Kindern als Unterstützung dienen kann.

2 Methoden

Die systematische Literaturrecherche wurde in drei Phasen nach dem Verfahren von Kitchenham [2] durchgeführt. Um die Untersuchung gezielt durchzuführen, wurden zunächst die Forschungsfragen und der Aufbau der Analyse geplant. Als nächstes wurden die Studien auf ihre Qualität überprüft und ausgewählt. Dazu gehören ebenfalls die Datenextraktion und die Datensynthese. Als letztes wurden die Ergebnisse wiedergegeben. In folgendem Kapitel werden diese Schritte detailliert beschrieben.

2.1 Forschungsfrage

Für die systematische Literaturrecherche wurden zunächst die Forschungsfragen formuliert. Da bereits die Erkenntnis besteht, dass VR als Therapie für autistische Kinder in der Emotionserkennung eingesetzt werden kann, war es notwendig zu erfahren, ob der Einsatz von VR auch bei der Unterstützung im Alltag möglich wird. Somit ergaben sich folgende Forschungsfragen: „Wie ist die Akzeptanz der Kinder bei der Nutzung der VR-Brille?“, „Kann die VR-Brille eine Unterstützung im Alltag leisten?“ und „Kann die Lernfähigkeit der Kinder mithilfe einer VR-Brille spielerisch gefördert werden?“

2.2 Quellen und Suchstrategie

Um relevante wissenschaftliche Arbeiten zu finden, wurden die Datenbanken von PubMed, ScienceDirect, IEEE und Web of Science durchsucht. Es wurden nur Quellen berücksichtigt, welche einen Bezug auf die Technologie und die damit verbundene Beziehung zur Medizin haben. Außerdem wurde als Filter für das Jahr der Publikation der Arbeiten 2019 bis 2020 verwendet. Somit wurden nur die aktuellen Studien angezeigt.

2.2.1 Auswahl der Artikel

Für die Suche nach relevanten Arbeiten, welche zur Beantwortung der Forschungsfragen dienen sollen, wurden die Suchbegriffe festgelegt und gesucht. Um die Bereiche der Medizin und der Technologie abzudecken, wurden die Suchbegriffe „Autismus-Spektrum-Störung“ und „Virtual Reality“ festgelegt. Da diese Begriffe im Zusammenhang gesucht werden sollen, werden die Operatoren AND und OR verwendet. Somit ergab sich die Aussage **„(virtual reality OR virtual environment) AND (autism OR autism spectrum disorder) AND (child OR children)”** als Suchbegriff.

2.2.2 Auswahlkriterien für Studien

Als Auswahlkriterien wurden mehrere Punkte berücksichtigt. Diese Kriterien beruhen darauf, dass aufgelistete Artikel inhaltlich nicht nur die medizinische Behandlung untersuchen, sondern einen Bezug auf VR beinhalten. Berücksichtigte Einschlusskriterien werden in der Tabelle 1 dargestellt. Auch für ausgeschlossene Studien wurden Kriterien aufgestellt, welche in Tabelle 2 zu sehen sind.

Tabelle 1. Einschlusskriterien der Studien

Einschlusskriterien	
E1	Studien, welche im Jahr 2019 und 2020 veröffentlicht wurden
E2	Journals und Konferenzbeiträge
E3	Studien mit dem Fokus auf Autismus-Spektrum Störung
E4	Studien im Zusammenhang mit der Nutzung von Technologie (VR)
E5	Studien, welche im Kontext des Lernens geführt werden

Tabelle 2. Ausschlusskriterien der Studien

Ausschlusskriterien	
A1	Studien, welche reinen Fokus auf die medizinische Forschung haben
A2	Studien, die den Fokus auf Betreuer oder Eltern legen
A3	Studien mit dem Fokus auf Augmented Reality

Die Einschluss- und Ausschlussverfahren werden als Diagramm in Abbildung 2 veranschaulicht, um den Prozess der Suche und die Auswahl der Studien deutlichen zu machen.

2.3 Datensynthese

Insgesamt wurden mit dieser Suche 292 Artikel gefunden. Mehrere Datenbanken wurden untersucht und dabei entstandene Duplikate entfernt. Anhand der Titel-Überprüfung wurden ebenfalls 125 Artikel entfernt. Dies wurde anhand der Ausschlusskriterien durchgeführt. Für die Untersuchung der übrigen 125 Artikel wurde zunächst der Abstract gelesen. Auch hier wurden die Einschluss- und Ausschlusskriterien beachtet und Studien selektiert. Schließlich wurden insgesamt 42 Artikel als relevant eingestuft und für die Literaturanalyse beibehalten. Diese Artikel wurden ausführlicher untersucht, um wichtige Informationen zu extrahieren. Nach dieser Extraktion wurden die restlichen Arbeiten komplett gelesen. Eine erneute Selektion wurde in Bezug auf die relevant der Forschungsfragen durchgeführt und schließlich insgesamt 24 Artikel für die Analyse ausgesucht. Um den Rahmen dieser wissenschaftlichen Arbeit jedoch einzuhalten, wurden hierfür 16 Artikel in die Analyse einbezogen, in der die Nutzung und Akzeptanz einer VR-Brille detailliert beschrieben wurde.

3 Erkenntnisse der Studien

3.1 Erscheinungsjahr und Art der Artikel

Um die aktuellen Forschungen zu berücksichtigen, wurde während der Suche ein Filter für das Erscheinungsjahr der veröffentlichten Arbeiten gesetzt. Dazu wurden die Arbeiten berücksichtigt, welche im Jahr 2019 und 2020 publiziert wurden. Eine Verteilung wird in Abbildung 1 dargestellt. Für die Nutzung der Daten und Ergebnisse der Artikel wird die Art berücksichtigt. Hier war es wichtig, ebenfalls empirische Daten zu erhalten. Abbildung 3 zeigt die Verteilung der Artikelarten.

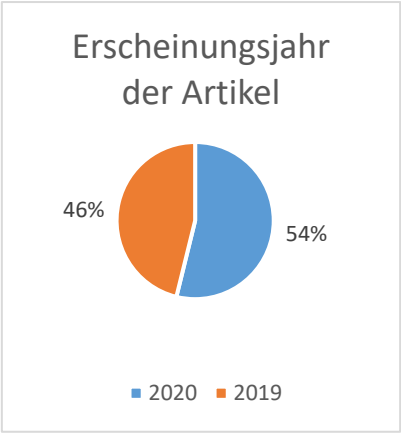

Abbildung 1. Anteil der Publikationsjahre

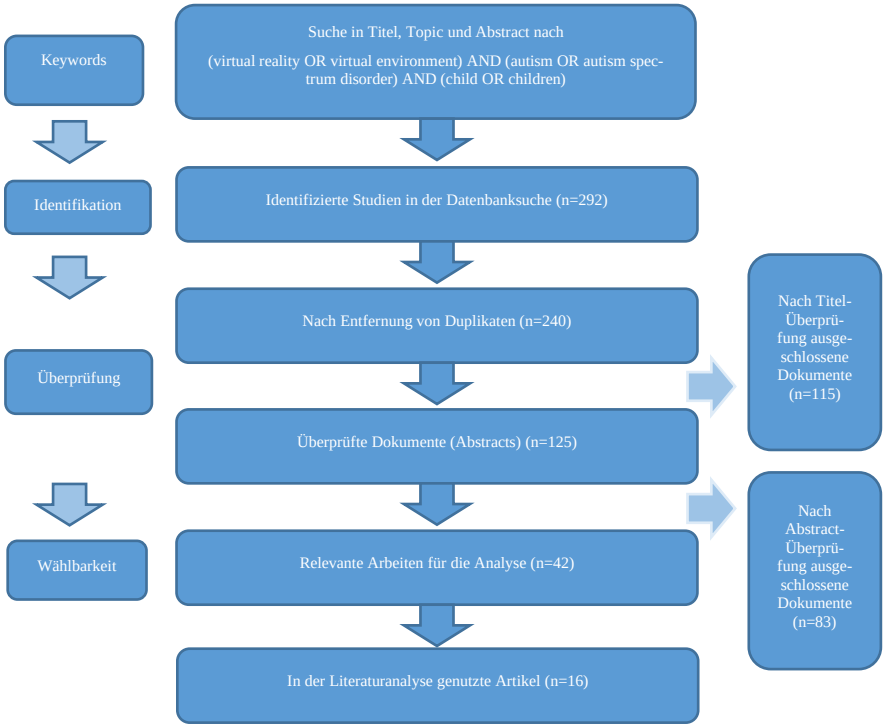

Abbildung 2. Flow Chart: Prozess bei Artikelauswahl

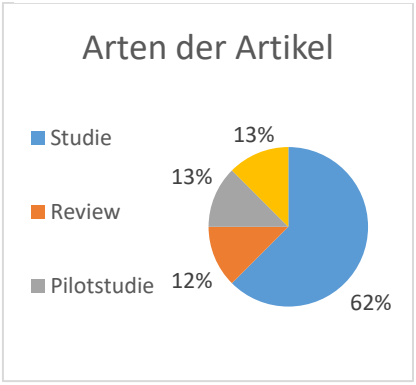

Abbildung 3. Art der Artikel

3.2 Design der Studien und Stichprobengröße

Die Studien werden größtenteils für das Testen der Nutzbarkeit und der Akzeptanz der Probanden durchgeführt. Gefundene Artikel beziehen sich auf autistische Kinder, wobei die Altersgruppe variieren kann. Studien aus Katar [3], Kanada [4] und den Niederlanden [5] haben die breiteste Stichprobengröße und erzielen damit eine höhere Informationsrate (Katar = 45 Kinder, Niederlande =22 Kanada = 35 Kinder). Weitere Studien werden entweder nur mit drei, fünf oder maximal 12 Kindern durchgeführt. Das vorgesehene Alter für die Studien beträgt im Durchschnitt 10 Jahre. Jedoch kann dieser durchschnittlicher Wert nicht verallgemeinert werden, da es auch Studien gibt, welche nur Kinder im Alter von 4-8 Jahren untersuchen. Das Design der Studien lässt sich mit einem Test zur Nutzbarkeit erklären. Die durchgeführten Experimente mit den Kindern liefern Informationen darüber, ob die Kinder mit der virtuellen Welt zurechtkommen und ob sie die daraus erlernten Fähigkeiten nach den Sitzungen in der realen Welt umsetzten können.

Nach den Sitzungen wird jeweils ein Feedback von den Probanden und von den Eltern oder Betreuern eingeholt. Auch nach Beenden der Studie werden Rückschlüsse über das Verhalten der Kinder gezogen. Als Einschlusskriterien für Probanden wird vor dem Experiment eine ärztliche Diagnose der Störung und falls vorhanden, Nachweis für zusätzliche Störungen wie Phobien, Hyperaktivität und Fähigkeit zur verbalen oder nonverbalen Kommunikation der Kinder eingeholt.

3.3 Verwendete Technologien

In der Arbeit von Bilikis et al. [3] wird ein virtuelles Klassenzimmer dargestellt, um die Aufmerksamkeit der Kinder zu messen. Hierfür wird ein Eye-Tracker eingesetzt, welches die Blicke der Kinder verfolgen soll. Mit dem Eye-Tracker kann gemessen werden, wo sich die Interessengebiete (Area of Interest = AoI) der Kinder befinden und wie lange dieses Interesse bestehen bleibt. Malihi et al. [4] stellen ebenfalls eine virtuelle Umgebung vor, in der die Probanden mit einem Head-Mounted-Display (HMD) ein Szenario im Bus zu sehen bekommen. Eine weitere Arbeit aus den USA von Dixon et al. [6] berichtet ein Szenario mit einem HMD für die Lernfähigkeit zur Überquerung der Straße mit Autos und Ampeln. Die Untersuchung von Johnston et al. [7] verwendet zusätzlich zu einem HMD, ein räumliches Audio und ein Szenario im dunklen Wald, um die visuellen und auditorischen Sinneserfahrungen der realen Welt abzustimmen. In dem Pilotbericht von Miller et al. [8] wird ein IphoneX zusammen mit einer Google Cardboard genutzt, um die Nützlichkeit einer kostengünstigen VR-Erfahrung zu demonstrieren. Um einen Vergleich zwischen den virtuellen Umgebungen in Bezug auf das Design der VR-Lernszenarien zu berichten, wird für die Arbeit von Shing et al. [9] eine CAVE (cave automatic virtual environment) und ein HDM verwendet. CAVE ist ein dreidimensionaler Raum, in der die Illusion der virtuellen Realität projiziert werden kann. Ravindran et al. [10] untersuchen eine selbstentwickelte Software, namens „Floreo“, welche Szenen für Google Carboard kompatible Smartphones bereitstellt, wie die Aufmerksamkeit der Kinder mit verschiedenen Spielen gefördert werden kann. Eine weitere Arbeit von Rahmadiva et al. [11] nutzt neben einem HMD, noch das Leap-Motion Gerät. Damit werden die Hände der Probanden durch Infrarot gelesen und in manipulierte Hände in der virtuellen Welt übertragen. Leap-Motion kann mithilfe von Infrarot Objekte erfassen und diese in die virtuelle Welt übertragen. Damit können die Hände der Probanden in der virtuellen Welt als Interaktionsmittel genutzt werden. Die Auswahl für einen HMD fiel bei den Studien auf die Oculus Rift. Mit dieser Brille, welche an einem Computer angeschlossene ist, ist das Erlebnis einer immersiven Realität und anzeigen von dreidimensionalen Bildern möglich.

3.4 Ziele der Studien

Das Ziel dieser Studien besteht darin, die Akzeptanz der virtuellen Umgebung bei autistischen Kindern zu testen und die Lernbereitschaft hiermit zu stärken. Da autistische Kinder häufig an Konzentrationsschwächen leiden und ihre Aufmerksamkeit schnell verlieren, wird mit dem Einsatz von VR versucht, das Interesse der Kinder aufrechtzuerhalten. Sobald die Aufmerksamkeit verschwindet, verlieren sie ihr Interesse an der Aufgabe und den damit verbundenen Lerneffekt. Hierfür wird in der Arbeit von Bilikis et al. [3] untersucht, wie Kinder auf äußerliche Reize reagieren und wie stark sie davon abgelenkt werden. Mit diesen Reizen in der virtuellen Umgebung kann experimentiert werden, indem die Kinder mehr oder weniger Reizen ausgesetzt werden. Diese Untersuchung wird mit der Berechnung der Zeiten wie „Zeit zum Fixieren auf AoI“, „Dauer der ersten Fixierung“, „Durchschnittliche Fixierungsdauer“ und „Summe der Fixationszahl“ erweitert. Diese Zeiten geben einen Aufschluss darüber, wie die virtuelle Umgebung gestaltet werden kann, oder welche Elemente zur Erhaltung der Aufmerksamkeit zusätzlich hinzugefügt werden sollten. Auch der Einsatz der „Floeo“- Software zielt darauf ab, Elemente in die virtuelle Welt zu setzen,

damit die Aufmerksamkeit der Kinder auf gewünschte Elemente oder Bereiche in der virtuellen Welt gezogen wird [10]. Außerdem wird in dieser Arbeit untersucht, ob für Kinder mit eingeschränkten verbalen Fähigkeiten und Bedürfnissen, in Bezug auf sozialer Gegenseitigkeit, ein Lernszenario in VR aufgebaut werden kann [10]. Zusätzlich zu der Aufmerksamkeit, spielt der Fokus auf relevante Informationen eine wichtige Rolle bei der Zielsetzung der Studien. Rahmadiva et al. [11] entwickeln hierfür ein Spiel, in der die Kinder die Verlagerung der Aufmerksamkeit lernen sollen. Denn häufig haben autistische Kinder das Problem, nicht zu wissen, welche Aufgabe wichtiger ist. Deshalb ist es zunächst wichtig das Interesse der Kinder zu wecken, um dann die Konzentration zu steigern. Anschließend kann das Training für den Fokus angegangen werden.

Neben der Aufmerksamkeit ist auch die Emotionserkennung ein Defizit, das bei Autismus auftreten kann. Um hierfür eine Hilfestellung zu leisten, wird eine Studie durchgeführt, die mithilfe von Avataren die Emotionserkennung erleichtern soll [5]. Da durch die Avatare die Emotionen stärker zum Vorschein gebracht werden können, besteht das Ziel dieser Studie darin, ein Lernszenario für Kinder zu entwickeln, in der sie bei der Überquerung einer Einkaufsstraße die Emotionen der entgegenkommenden Avatare erkennen sollen. Die Auswahl, aus Multiple-Choice-Antworten gelingt durch ein Klicken auf die vorgegebenen Antwortmöglichkeiten. Außerdem wird mit dieser Studie versucht die Fähigkeiten der Kinder in Bezug auf das Verständnis von Interaktionen der Avatare zu stärken und durch ein Rollenspiel eine sozialkognitive Problemlösungstechnik zu entwickeln [5].

Auch Fallbeispiele wurden in den Studien untersucht und nach Lösungsansätzen gesucht. Aufgrund der schnellen Verwirrung und Konzentrationsschwäche der Kinder kann das Lernen im Alltag zu Schwierigkeiten führen. Deshalb wird in der virtuellen Welt experimentiert, das Lernen der Straßenregeln spielerisch zu ermöglichen [6],[12]. Dazu wird ein Szenario erstellt, in der die Kinder eine Straße mit Autos, Stoppschilden und Ampeln überqueren sollen. Ziel dabei ist, die Grundregeln einer Straßenüberquerung zunächst in der virtuellen Welt zu lernen und diese Erkenntnisse danach in die reale Welt umzusetzen. Aufgrund der regulierbaren Optionen der virtuellen Welt, mit leerer Straße oder stark befahrene Straßen, wird gezielt die Angst der Kinder minimiert und eine sichere Umgebung vermittelt [6],[12].

Ein weiteres Fallbeispiel wird für die Förderung der Flugreisefähigkeit durchgeführt [8]. Hier wird der Fokus auf die soziale Kommunikation und Interaktion (social communication and Interaction, SCI) gelegt und versucht, mithilfe von Simulationen, die Angstzustände der Kinder in einer neuen Umgebung oder einem Ort mit vielen Menschen zu mildern. Das Ziel ist es, dass Kinder nach diesen Sitzungen einen Aufenthalt am Flughafen oder eine Reise mit dem Flugzeug problemlos meistern können [8]. Zusätzlich zu der Milderung der Angstzustände wird eine weitere Studie durchgeführt, die gezielt Betroffene auf einen Alarm und Sicherheitszustände trainieren möchte. Shree et al. [13] entwickelt hierfür ein Spiel in der virtuellen Umgebung, dass die Aufmerksamkeit der Kinder wecken soll, wenn eine Gefahr besteht. In diesem Szenario wird ein Haus dargestellt, welches sich in einer Waldumgebung befindet, in der nach wenigen Sekunden ein Brand ausbricht. Mit einem Hinweis als ausbrechendem Alarmsignal oder den Hilferufen der Avatare, wird das Kind auf den Brand aufmerksam gemacht. Nach diesem Hinweis sollen sich die Kinder in der virtuellen Welt in das Haus begeben und das Objekt, welches Feuer gefangen hat, identifizieren. Nach der Identifizierung muss sich das Kind in Sicherheit bringen, aus dem Haus laufen und im Wald in Sicherheit warten. Es wird ausdrücklich betont, dass das Ziel dieser Untersuchung die Entwicklung des Bewusstseins für Alarmsituationen ist [13]. Somit entwickelt sich bei dem Kind die Improvisationsfähigkeit zur Sensibilisierung

für Feuer und es kann die Gefahr, ohne in Panik auszubrechen, erkennen.

Um die reale Welt wahrzunehmen, muss das Gehirn einen konstanten Strom von multimodalen Informationen aus verschiedenen Sinneskanälen entschlüsseln. Diese Informationen können beispielsweise visuell, auditiv oder textlich sein. Dafür wurde in der Arbeit von Johnston et al. [7] dargestellt, dass 65% der Autisten eine Komplikation in der auditiven Wahrnehmung besitzen. Deshalb haben sie die räumliche Aufmerksamkeit und die Fähigkeit zur Schalllokalisierung in einer multimodalen VR-Umgebung untersucht. Ziel ist es, die Kopfbewegungen und Rotationsgenauigkeit auf auditive Reize zu bestimmen. Dafür wurde ein Szenario im dunklen Wald vorbereitet, in der die Kinder mithilfe von Sounds Objekte finden sollten. Da in dieser Studie die auditiven Ansätze untersucht werden, wurde gezielt eine dunkle Umgebung ausgewählt, damit die Aufmerksamkeit durch das helle Licht nicht verstreut wird.

In dem Artikel von Malihi et al. [4] wird nicht als Ziel gesetzt, einen Lerneffekt zu erschaffen, sondern die Sicherheit und Usability von VR-Umgebungen bei autistischen Kindern zu testen. Dafür wurde ebenfalls ein Szenario vorbereitet, in der die Kinder in einen stehenden Bus einsteigen und sowohl sensorische als auch soziale Auslöser angezeigt werden. Für die Usability wird hier die Effektivität, Effizienz und die Zufriedenheit getestet, welches durch Fragebögen und Selbstberichten untersucht wird. Auch die Sicherheit während der Nutzung des HDM wird anhand dieser Rückmeldungen und Beobachtungen während der Sitzung untersucht [4]. Li et al. [9] untersuchen, wie sich verschiedene immersive VR-Umgebungen auf das Design der Lernszenarien auswirkt. Es wird untersucht, wie die Lerninhalte gestaltet werden können, damit das erfahrungsbasierte Lernen für autistische Kinder in einer virtuellen Umgebung erleichtert wird. Hier wird von dem Konzept für das erfahrungsbasierte Lernen von Kolb [9] ausgegangen. Dieses Konzept beinhaltet einen iterativen Lernzyklus, der aus vier Schritten besteht:

1. Konkrete Erfahrung
2. Reflektierende Beobachtung
3. Abstrakte Konzeptualisierung
4. Aktives Experimentieren

Auf dieser Grundlage wurden „interaktive soziale Geschichten“ und „Erleben sozialer Vorfälle“ vorbereitet und damit bezweckt, die Lerneffektivität der Kinder zu messen.

4 Diskussion

Nach der Darstellung der Vorgehensweise mehrerer Arbeiten, werden in diesem Abschnitt die gesammelten Ergebnisse diskutiert. Die Beantwortung der Forschungsfragen ist mit den Erkenntnissen der Studien und Informationen aus verschiedenen Literaturrecherchen möglich.

4.1 Ergebnisse

Die Ergebnisse der Arbeiten wurden zunächst durch Teilnehmer- oder Therapeutenbefragungen gesammelt [5]. Während der Testphase wurde festgehalten, wie viele Kinder, beispielsweise den Testvorgang mit einem Bus, abgebrochen haben [4]. Als Grund für diesen Abbruch wird eine negative Erfahrung mit einem Bus genannt. Der Lärm und das Zappeln lösten Angstzustände bei den Kindern, weshalb sie den Vorgang nicht weiterführen konnten [4]. Allerdings wurden bei 30% der Nutzer weniger Nervosität beim Einsteigen in einen echten Bus notiert [4]. Die Eltern der Kinder berichten ebenfalls eine positive Änderung beim Einsteigen in einen realen Bus [4]. Malihi et al. stellen einen vorläufigen Beweis dafür auf, dass Head-Mounted Displays das Potenzial dazu haben, die User Experience, gegenüber auf einem Monitor angezeigten Videos, zu verbessern [4]. Dixon et al. [6] berichten, dass kurze Videos keine Nachwirkung auf die reale Umgebung haben. Auf der anderen Seite lenken lange Videos die Aufmerksamkeit auf unrelevante Bereiche. Deshalb empfeh-

len sie eine Interaktion, während dem Vorgang, für eine nachhaltige Wirkung der Nutzung. Laut Nijman et al. [5] sind Multiple-Choice-Antworten keine hilfreiche Methode. Stattdessen werden offene Antworten eingesetzt, um das Verständnis zu „wie“ und „warum“ zu verstärken. Es hat sich ebenfalls rausgestellt, dass kombinierte Aufgaben Probleme bereiten können [3]. Gleichzeitig auf Buchstaben zu schauen, diese laut auszusprechen und auf die richtige Antwort zu klicken führen bei den Kindern zu einer Irritation [3].

Es muss betont werden, dass die Problematik von autistischen Kindern sehr unterschiedlich ausfallen kann. Die grundlegenden Probleme liegen jedoch in der Wahrnehmung, Aufmerksamkeit, Emotionserkennung und Bewusstsein. Aufgrund der variierenden Reaktionen der Kinder auf unterschiedliche Reize wird eine individuelle oder anpassbare Benutzeroberfläche vorgeschlagen [14]. Ein Kind mit Autismus-Spektrum-Störung braucht eine Ansprechperson, welches die Interessen teilt und konsequent bleibt auch wenn sich das Kind falsch verhält. Dies ist aus therapeutischer Sicht langfristig leider nicht möglich. Dafür wird ein individueller, virtueller Freund vorgeschlagen, der nach Vorlieben anpassbar ist und welcher auf Grundlage der gesammelten Daten intelligent über nächste geeignete Schritte entscheiden kann und eine Wissensbasis in Bezug auf die Interessen aufbaut. Dies kann mithilfe autonomer Systeme bereitgestellt werden [14]. Da eine virtuelle Umgebung vorhersehbar und strukturiert ist, können autistische Kinder ihre Routine und ihr ständig wiederholendes Verhalten beibehalten. Es ist wichtig, die Bequemlichkeit bei Kindern mit dieser Störung soweit es geht nicht zu beeinträchtigen. Für die Aufrechterhaltung der Aufmerksamkeit wird dringend geraten, notwendig Elemente, wie Punkte-Sammeln, verschiedene Levels und Belohnungen, zu integrieren. Es hat sich gezeigt, dass Feedbacks und Avatare die Aufmerksamkeit der Betroffenen fördern [15].

Die virtuelle Welt ist eine sichere und unterstützende Umgebung in der ein Wissenstransfer zwischen der virtuellen und realen Welt stattfinden kann. Da die Manipulation der Einführung oder Entfernung von Aktivitäten oder Aktionen möglich ist, bietet sie ein leistungsstarkes und anpassbares Lernen [16]. Somit werden Hindernisse von autistischen Kindern, die darin bestehen, Lernmaterialen im entsprechenden Format zu finden, aufgehoben. Der Profit von VR besteht darin, Flexibilität, einfache und intuitive Nutzung und eine geringe körperliche Anstrengung zu bieten [16].

4.2 Beantwortung der Forschungsfragen

Nach Jeffs et al. [16] kann der Einsatz der Virtual Reality für die Förderung der Lese- und Schreibfähigkeiten genutzt werden. Dadurch können die Kinder beispielsweise das Lesen und die Unterscheidung gedruckter Markennamen auf Produkten oder Etiketten einfacher durchführen. Ebenfalls kann die Kauffähigkeit, beispielsweise beim Einkaufen von Lebensmitteln, gestärkt werden [16]. Betroffene Kinder können die sozialen Fähigkeiten, bei der Bitte um Hilfe oder die Sicherheitsfähigkeiten, beim Erlernen von Zeichen, für das Verringern einer Gefahr, erlernen [16]. Somit wird klar, dass der Einsatzbereich der virtuellen Realität je nach Bedarf variiert werden kann.

Ziel dieser systematischen Literaturrecherche war es, die folgenden Forschungsfragen zu beantworten:

- Wie ist die Akzeptanz der Kinder bei der Nutzung einer VR-Brille?
- Kann die Brille eine Unterstützung im Alltag leisten?
- Kann die Lernfähigkeit mithilfe einer VR-Brille spielerisch unterstützt werden?

Mit den gesammelten Erkenntnissen können diese Fragen sehr zufriedenstellend beantwortet werden. Untersuchte Artikel haben

gezeigt, dass die Kinder die VR-Brille sehr schnell akzeptieren und auch Freude daran haben diese zu Nutzen. Bei entstehenden Schwierigkeiten bei der ersten Nutzung haben Schulungen geholfen, diese Probleme zu beseitigen. Fallbeispiele haben gezeigt, dass die Nutzung einer VR-Umgebung auch eine Unterstützung für den Alltag leisten kann, wie zum Beispiel die Überquerung einer Straße oder die Vorbereitung auf eine Reise mit dem Flugzeug. Nach dem Ansatz von Zhao et al. [14] wird verstärkt, dass der Einsatz eines individuellen Assistenten hilfreich wird. Mit diesem Anhaltspunkt können weitere Schwierigkeiten im Alltag überwunden werden. Da die Kinder die VR-Brille flexibel nutzen können, ist das Training individuell gestaltbar. Erhaltene Feedbacks nach jeder Sitzung und Nachforschungen nach Abschluss der Experimente haben gezeigt, dass die Lerneffekte nach den Testvorgängen beibehalten werden. Außerdem wird bestätigt, dass humanoide oder nicht-humanoide Avatare den Bildungsprozess voranbringen und die sozialen Fähigkeiten verbessern [17].

Abschließend kann in Kenntnis gesetzt werden, dass die virtuelle Realität ein wichtiger Bestandteil der Lernförderung für autistische Kinder ist und zukünftig auch als Therapietool im Alltag zu sehen sein wird. Da jedes Kind individuelle Schwierigkeiten hat, können diese Probleme durch die Nutzung eines persönlichen VR-Assistenten im Alltag beseitigt werden. Weiterführend wäre es interessant zu untersuchen, ob ein autonomes, anpassbares System für jedes Kind, die erwarteten Erfolge zur Überwindung der Probleme im Alltag machbar macht.

Literaturverzeichnis

[1] S. L. Hyman, S. E. Levy, S. M. Myers. Identification, Evaluation, and Management of Children With Autism Spectrum Disorder. American Academy of Pediatrics.2020.

[2] B. Kitchenham, S. Charters. Guidelines for performing Systematic Literature Reviews in Software Engineering. 2007. Online verfügbar unter https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf.

[3] B. Bilikis, D. Al-Thani, M. Qaraqe. et al.. Impact of mainstream classroom setting on attention of children with autism spectrum disorder: an eye-tracking study. Univ Access Inf Soc. 2020. Online verfügbar unter https://doi.org/10.1007/s10209-020-00749-0.

[4] M. Malihi, J. Nguyen, R. E Cardy. et al. Short report: Evaluating the safety and usability of head-mounted virtual reality compared to monitor-displayed video for children with autism spectrum disorder. 2020. Online verfügbar unter https://doi.org/10.1177/1362361320934214.

[5] S. A. Nijman, W. Veling, K. Greaves-Lord. et al. Dynamic Interactive Social Cognition Cognition Training in Virtual Reality (DiSCoVR) for People With a Psychotic Disorder: Single-Group Feasibility and Acceptability Study. JMIR Ment Health. 2020. Online verfügbar unter https://doi.org/10.2196/17808. PMID: 32763880; PMCID: PMC7442939.

[6] D. R. Dixon, C. J. Miyake, K. Nohelty. et al. Evaluation of an Immersive Virtual Reality Safety Training Used to Teach Pedestrian Skills to Children With Autism Spectrum Disordere. Behav Analysis Practice 13, 631-640. 2020. Online verfügbar unter https://doi.org/10.1007/s40617-019-00401-1.

[7] D. Johnston, H. Egermann, G. Kearney. et al. Measuring the Behavioral Response to Spatial Audio within a Multi-Modal Virtual Reality Environment in Children with Autism Spectrum Disorder. Applied Sciences 9, 3152. 2019. Online verfügbar unter https://www.mdpi.com/2076-3417/9/15/3152/htm.

[8] I. T. Miller, B. K. Wiederhold, C. S. Miller. et al. Virtual Reality Air Travel Training with Children on the Autism Spectrum: A Preliminary Report. Cyberpsychol Behav Soc Net. 2020. Online verfügbar unter https://doi.org/10.1089/cyber.2019.0093.

[9] C. Li, H. H. Shing, P. K. Ma. A Design Framework of Virtual Reality Enabled Experiential Learning for Children with Autism Spectrum Disorder. ICBL. 2019. Lecture Notes in Computer Science, vol 11546. Springer. Online verfügbar unter https://doi.org/10.1007/978-3-030-21562-0_8.

[10] V. Ravindran, M. Osgood, V. Sazawal. et al. Virtual Reality Support for Jpint Attention Using the Floreo Jpint Attention Module: Usability and Feasibility Pilot Study. JMIR Pediatr Parent. 2019. Online verfügbar unter https://doi.org/10.2196/14429. PMID: 31573921.

[11] M. Rahmadiva , A. Arifin, M. H. Fatoni. et al. A Design of Multipurpose Virtual Reality Game for Children with Autism Spectrum Disorder. International Biomedical Instrumentation and Technology Conference. Indonesia. 2019. Online verfügbar unter https://doi.org/10.1109/IBITeC46597.2019.9091713.

[12] Y. Peng, W. Zhu, F. Shi. et al. Virtual Reality Based Road Crossing Training for Autistic Children with Behavioral Analysis. IFTC 2018. Communications in Computer and Information Science, vol 1009. Springer, Singapore. 2019. Online verfügbar unter https://doi.org/10.1007/978-981-13-8138-6_39.

[13] N. Shree, A. G. Selvarani. Virtual Reality based System for Training and Monitoring Fire Safety Awareness for Children with Autism Spectrum Disorder. 2020 5th International Conference on Devices, Circuits and Systems. India. Pp. 26-29. 2020. Online verfügbar unter https://doi.org/10.1109/ICDCS48716.2020.243541.

[14] W. Zhao, X. Liu, X. Luo. Virtual Avatar-Based Life Coaching for Children with Autism Spectrum Disorder. In Computer, vol. 53, no.2, pp. 26-34. 2020. Online verfügbar unter https://doi.org/10.1109/MC.2019.2915979.

[15] K. Valencia, C. Rusu, D. Quinones, E. Jamet. The Impact of Technology on People with Autism Spectrum Disorder: A Systematic Literature Review. 2019. Online verfügbar unter https://doi.org/10.3390/s19204485.

[16] T. L. Jeffs. Virtual Reality and Special Needs. Themes in Science and Technology Education, v2 n1-2, pp. 253-268. 2009. Online verfügbar unter https://eric.ed.gov/?id=EJ1131319.

[17] S. Boucenna, A. Narzisi, E. Tilmont. et al. Interactive Technologies for Autistic Children: A Review. Cogn Comput 6, pp. 722-740. 2014. Online verfügbar unter https://doi.org/10.1007/s12559-014-9276-x.

[18] Umweltbudnesamt. Autismus/Autismus-Spektrum-Störung. 2020. Online verfügbar unter https://www.umweltbundesamt.de/themen/gesundheit/umweltmedizin/autismusautismus-spektrum-stoerungen#undefined

Design Thinking in Unternehmen - Dokumentation von Ideen innerhalb der Ideation-Phase

Claushinrich Himstedt
Hochschule Reutlingen
Claushinrich.Himstedt@Student.Reutlingen-University.de

Abstract

Design Thinking wird neben anderen agilen Methoden zur Innovation und Lösung komplexer Probleme eingesetzt. Die Anwendung findet auch in Unternehmen innerhalb von Teams branchenübergreifend statt. Dabei kann der Ansatz in verschiedensten Bereichen angewandt werden mit dem Ziel nutzerzentrierte Lösungen zu entwickeln. Bei der Entwicklung neuer Ideen in Teams werden häufig die Dokumentation und Rückverfolgbarkeit vernachlässigt, da es keinen transparenten Prozess gibt wie Ideen entstehen und wie die Lösung eines Problems erreicht wurde. Dadurch wird vor allem im Handover-Prozess zwischen Designern und Ingenieuren die interdisziplinäre Arbeit in Teams erschwert. Eine Rückverfolgung und Transparenz der Ideen ist durch fehlende Dokumentation, ob digital und analog oft nicht gegeben. Durch eine Literaturrecherche wurden Probleme innerhalb verschiedener Anwendungsbereiche von Design Thinking analysiert und erläutert. In einer anschließende Umfrage wurden die Probleme in verschiedenen Unternehmen genauer betrachtet, um Lösungsansätze für eine besser Dokumentation von Ideen abzuleiten. In der Umfrage bestätigte sich die fehlende Dokumentation, die unter anderem im Bildungsbereich gegeben ist, vor allem im IT-Bereich nicht.

CCS Concepts

• **Human-centered computing** ~ Collaborative and social computing ~ Collaborative and social computing theory, concepts and paradigms ~ Collaborative content creation • **General and reference** ~ Cross-computing tools and techniques ~ Empirical studies

Keywords

Design Thinking, Ideengenerierung, Dokumentation

1 Einleitung

Das Vorantreiben neuer Innovationen ist für Unternehmen mit Blick auf die Zukunft von sehr großer Relevanz. Um zukunftsorientiert in komplexen und dynamischen Umgebungen innovativ zu bleiben, müssen grundlegende Methoden eingesetzt werden. Bevor die Innovation beginnen kann, wird bei der Generierung von Ideen begonnen. Ein Ansatz, der die Entwicklung von Ideen und neuen Innovationen unterstützt, ist Design Thinking. Dabei legt Design Thinking einen besonderen Fokus auf die Kreativität und

---

Betreuerin Hochschule: Prof. Dr. Gabriela Tullius
Hochschule Reutlingen
Gabriela.Tullius@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

die Interdisziplinarität. Neben anderen agilen Methoden, wie Design Sprints, Scrum oder agiler Softwareentwicklung, bietet Design Thinking die Möglichkeit nutzerzentrierte Lösungen zu entwickeln. Die Anwendung dieser Methodologie bietet die Entwicklung einer Lösung zu einem definierten Problem in beliebigen Kontexten an.

1.1 Motivation und Ziel

Die Motivation der Arbeit leitet sich aus der Forschung zur Traceability (*deutsch: Rückverfolgbarkeit*) und Dokumentation bei der Ideengenerierung unter der Verwendung von Design Thinking ab. In dem wissenschaftlichen Beitrag von Behl et al[1] zur verbesserten Dokumentation von Design Thinking im Bildungsbereich und zu Traceability bei Innovationen [2] geht es um Verbesserungsmöglichkeiten. Dabei geht es darum, dass die Ideen innerhalb der Ideengenerierung oft nicht richtig dokumentiert werden oder nicht richtig rückverfolgbar sind Das hat zur Folge, dass Ideen verloren gehen oder der Ursprung einer Idee zur Umsetzung nicht richtig rückverfolgt werden kann. Die Arbeit geht dabei auch auf die Selbstwirksamkeit und die Gleichberechtigung bei der Ideengenerierung innerhalb von Teams ein. Da diese beiden Faktoren maßgeblich die Ideation-Phase mit beeinflussen, sollte ein Fokus auch auf weichen Teamfaktoren, wie Konstruktivität und Vertrauen, liegen. [17] Das Ziel ist es die Dokumentation und Selbstwirksamkeit bei der Ideengenerierung von Teams in Unternehmen zu untersuchen und Verbesserungsmöglichkeiten zu identifizieren.

1.2 Forschungsfragen

RQ1: Welche Möglichkeiten zur Dokumentation, Transparenz, Rückverfolgbarkeit und Selbstwirksamkeit gibt es bei der Ideengenerierung mit Design Thinking innerhalb von Teams? Diese Forschungsfrage soll mit Hilfe einer Literaturrecherche beantwortet werden. Dafür soll die Funktionalität zu den Möglichkeiten und ein aktueller Stand der Forschung als Überblick aufbereitet werden.

RQ2: Wie wird die Dokumentation, Transparenz, Rückverfolgbarkeit und Selbstwirksamkeit bei der Ideengenerierung in Unternehmen innerhalb von Teams umgesetzt? Um die Forschungsfrage beantworten zu können wird eine Umfrage auf der Basis der Erkenntnisse der Literaturrecherche entworfen und durchgeführt. Mit der Umfrage soll der Stand innerhalb von Unternehmen aufgezeigt und Lösungsmöglichkeiten für Verbesserungen abgeleitet werden.

1.3 Struktur der Arbeit

In Kapitel 2 wird ein Überblick über den Forschungsstand zum Thema mittels einer Literaturrecherche wiedergegeben. Das Kapitel 3 zeigt die Methodologie auf, die das Vorgehen beschreibt, um die Forschungsfrage RQ2 zu beantworten. In Kapitel 4 werden die Ergebnisse der Umfrage präsentiert. Kapitel 5 fasst die Ergebnisse der Arbeit zusammen. In Kapitel 6 folgt das Fazit und der Ausblick auf weitere Erkenntnisse.

2 Stand der Forschung

Im folgenden Kapitel werden die Grundlagen und der Stand der Forschung auf Basis einer Literaturrecherche aufgezeigt. Dabei wird im grundlegenden Design Thinking und die Phase der Ideengenerierung genauer betrachtet. Im Besonderen werden die Möglichkeiten und Herausforderungen bezüglich der Dokumentation, Transparenz, Traceability, sowie der Selbstwirksamkeit und Gleichberechtigung bei der Ideengenerierung genauer betrachtet.

2.1 Design Thinking

Design Thinking beschreibt einen Ansatz, der mit Hilfe von Kreativitätstechniken und Methoden versucht für komplexe Probleme Lösungen zu finden. Der Ansatz zur Problemlösung erfolgt dabei stets nutzerorientiert [20]. Ursprünglich wurde Design Thinking von den beiden Standford University Professoren Terry Winograd und Larry Leifer zusammen mit Gründer der Innovationsagentur David Kelly 1991 entwickelt [8].

Nach Plattner et al. [20] besteht der Erfolg von Design Thinking aus drei Faktoren. Zum einem aus dem *Design Thinking Prozess* selbst, einem *variablen Raum* und einem *multidisziplinären Team* [12]. Der Design Thinking Prozess lässt sich dabei nach dem Hasso Plattner Institute (HPI) und Plattner et. al [20] in 6 nutzerzentrierte iterative Phasen gliedern, siehe Abbildung 1. Die Phasen sind innerhalb des Design Thinking Ansatzes nicht voneinander getrennt. In der Literatur und Forschung zu Design Thinking gibt es unterschiedliche Strukturen des Ansatzes. Somit gibt es Modelle in denen 5-Phasen, wie vom Institute of Design der Stanford University [14] oder das Modell nach Plattner et al. [20], welches in der Arbeit genauer betrachtet wird, da es auch in Zusammenhang mit der Anwendung in Unternehmen untersucht wird [19].

In der 1. Phase *Verstehen* geht es darum eine Problemstellung zu verstehen und alle beteiligten auf den gleichen Wissensstand zu bringen, unter Berücksichtigung der Nutzerzentrierung [7]. In der folgenden 2. Phase *Beobachten* wird das Problem von außen genauer betrachtet und sich in den Nutzer hinein zu versetzen. Dafür werden sowohl auf Methoden der quantitativen wie der qualitativen Forschung, zurückgegriffen. Die Phase *Definieren* beschreibt das Zusammentragen der Informationen aus der vorhergegangen Phase, um eine einheitliche Wissensbasis im Team herzustellen [15]. Die Entstehung und Entwicklung neuer Ideen erfolgt in der 4. Phase *Ideen entwickeln [21]*. Innerhalb der Phase werden zu den vordefinierten und identifizierten Problemen mögliche Lösungen für den Nutzer entwickelt. Dabei werden möglichst viele Ideen generiert, um eine große Auswahlmöglichkeit zu erhalten. Eine Methode, die zur schnellen Ideengenerierung eingesetzt wird, ist das Brainstorming. Erst mit dem Abschluss der Ideengenerierung werden die Ideen bewertet und in Verbindung mit einer Realisierbarkeit gebracht. Daraus wird dann eine finale Idee an Hand der Priorisierung ausgewählt. [20] In der 5. Phase *Prototyp* werden erste einfache Prototypen auf Basis der Ideen für den Nutzer greifbar gemacht. Dieser kann in Form eines Objekts, Interface oder eines Szenarios angelegt sein. Es sollte eine zum entwickelten Produkt, System oder Service passende Form ausgewählt werden [15]. Den Abschluss des Design Thinking Ansatzes bildet die 6. Phase *Testen.* Hier werden die auf Basis der Ideen entwickelten Prototypen mit dem Nutzer getestet. Hier ist es das Ziel die Problemlösungen zu optimieren und weiterzuentwickeln. Dabei kann iterativ in die vorhergehende Phasen zurückgegangen werden, um Anpassungen für die Lösung zu machen [21].

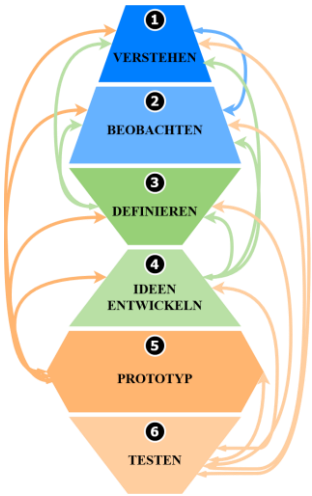

Abbildung 1: Modell zum Design Thinking Prozess nach Plattner et al. [20]

Um die Lösungsmöglichkeiten aus einem großen Spektrum zu wählen, wird in *multidisziplinären Teams* gearbeitet. Dabei ist es besonders wichtig Personen aus unterschiedlichen fachlichen Hintergründen und Funktionen mit einer Offenheit für andere Perspektiven zusammenzubringen. Dies geschieht am besten in heterogenen Teams aus fünf bis 6 Personen. [13]

Eine gute Zusammenarbeit funktioniert vor allem dann, wenn Hierarchien für die Anwendung abgelegt werden. Daher kommt es auch auf das richtige Mindset an, Konflikt- und Kommunikationsfähigkeit, sowie Engagement und die Bereitschaft zur Kollaboration [15]. Der dritte wichtige Faktor für die Anwendung von Design Thinking ist ein *variabler Raum*. Dabei geht es, dynamische Räume zu schaffen, in denen mit analogen Hilfsmitteln, wie Whiteboards oder digitalen Tools zur prototypischen Gestaltung, Ideen entwickelt werden können [15].

2.2 Design Thinking in Teams und Unternehmen

Design Thinking wird in übergreifenden Fachgebieten in Unternehmen und Teams angewandt. Dabei findet es zum Beispiel Anwendung in der Automobilindustrie, Dienstleitungsbranche, Marketing, und Unternehmensentwicklung [13].

Bei einer Anwendung von Design Thinking in Unternehmen und Teams gehen Herausforderungen einher. Besonders schwierig ist nach Dunne et al. [10] die Unterstützung durch die Führungsebene, da das volle Verständnis dafür fehlt. Ein weiterer Punkt ist die Isolierung von Design Thinking von anderen Prozessen oder Projekten im Unternehmen. Nur durch die Kombination können effiziente Ergebnisse erzielt werden.

Wenn Unternehmen zu der Entscheidung kommen Design Thinking anzuwenden, kann das zu messbaren Erfolgen und Verbesserungen bei der Innovation und Ideengenerierung für neue Produkte und Services [9]. Zudem kann eine erfolgreiche Anwendung zu intelligenten Investitionen bei Innovationen und einer erhöhten Rentabilität von Unternehmen führen [15].

2.3 Dokumentation

Ein Problem bei der Ideengenerierung innerhalb von Design Thinking stellt die Dokumentation von Ideen dar. Oft werden nur das Endergebnis einer Idee dokumentiert und nicht der Weg, sowie die Entwicklung der Idee. Dieses Problem hat Beyhl et al. [1] in der Anwendung von Design Thinking im Bildungsbereich analysiert. Dabei geht er auf die Problematik der Dokumentation von Studenten bei Projekten unter Verwendung von Design Thinking ein.

Hier ergibt sich das Problem, dass die untersuchten Studenten bei Projekten die Präsentationen mit Bildern und Dokumenten zwar dokumentieren, aber die Schritte, wie sie zu den Ideen gekommen sind vernachlässigen. Ein Problem in diesem Zusammenhang stellt auch oft das Verwenden von externen Applikationen, wie Filesharing-Diensten dar, da so Kommentare oder Anmerkungen zu Ideen nicht mit in den Prozess der Ideengenerierung mit aufgenommen werden [2]. Um das Problem zu lösen schlagen Beyhl et al. [1, 18] (siehe Abbildung 2), eine Plattform vor welche die verschiedenen Dokumentationskanäle bündeln soll. Dabei können Studenten ihre externen Dienste nutzen und diese zur Dokumentation in eine zentrale Dokumentationsplattform weiterleiten und damit interagieren [2].

Auch Hofer et al. [16] beschreibt das Problem, dass zwar eine zentrale Dokumentation von Arbeitsschritten zur Ideengenerierung stattfindet, die Dokumentation sich aber durch externe Kommunikation verlieren kann. Dies geschieht hauptsächlich durch den Einsatz von Smartphones, die einer zentralen Dokumentation Informationen vorenthalten.

Bitzer et al. [6] empfiehlt in Unternehmen einen Ansatz zur Kombination im Bereich der Dokumentation und des Wissensmanagement. Somit kann eine Kombination aus einem sozialen Betriebsnetzwerk für vergängliche Informationen und einem Wiki für nachhaltiges Wissen zu einer besseren Dokumentation führen, die auch auf Design Thinking angewendet werden kann.

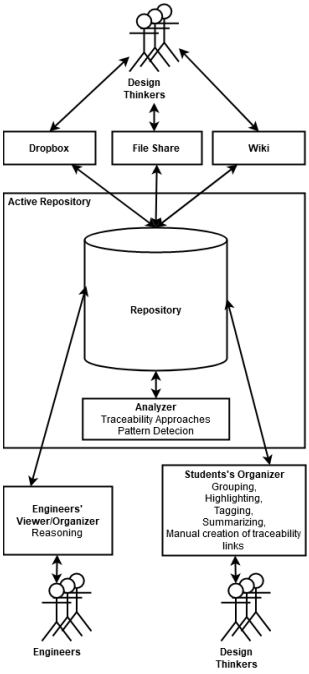

Abbildung 2: Architektur einer Dokumentations-Plattform für Ideen im Bildungsbereich nach Beyhl et al. [1]

2.4 *Traceability und Transparenz*

Damit ein Produkt oder Services erfolgreich umgesetzt werden können richten diese sich nach der Realisierbarkeit, Wirtschaftlichkeit und Nutzbarkeit. Damit Ingenieure zuvor entwickelte Ideen nach diesen drei Faktoren umsetzten können muss ein Verständnis für die Umsetzung vorhanden sein [3, 5]. Daher fehlt es für die Umsetzung an der *Traceability*(deutsch: Rückverfolgbarkeit), um den Prozess der Ideenentwicklung der Design Thinking Anwender zu den Ingenieuren verstehen zu können [4, 5]. Nach Beyhl et al. [2] wird oft nur der Endschritt einer Idee dokumentiert, daher ist die *Transparenz*, wie eine Idee zustande gekommen ist, oft nicht nachvollziehbar dokumentiert. Daher empfiehlt Beyhl nach drei Konzepten bei der *Traceability* vorzugehen [3]. Ein Teil davon sind die *Zeichen*, welche ein analoges oder digitales Artefakt beschreiben. Diese sollten mit einem Zusatz dokumentiert werden, ob es sich zum Beispiel um eine Persona oder Prototyp handelt. Als nächster Punkt sind die *Spuren*, welche den Zusammenhang zwischen verschiedenen Artefakten beschreiben. Die kann zum Beispiel auch die Dokumentation eines Whiteboards durch ein Foto darstellen. Die *Pfade* innerhalb der *Traceability* beschreiben den Pfad der Artefakte. Das heißt sie werden für die Umsetzung durch Ingenieure rückverfolgbar gemacht [2].

2.5 *Selbstverwirklichung und Gleichberechtigung*

Selbstwirksamkeit und die Gleichberechtigung von Ideen innerhalb eines Teams bei der Ideengenerierung sind innerhalb von Design Thinking von Bedeutung. Damit innerhalb des Teams eine Selbstverwirklichung möglich ist, sollte ein geteiltes Vokabular bei der Lösung und Entwicklung von Ideen präsent sein. Dieser Faktor ist nach Gibbons besonders wichtig um über Ideen klar kommunizieren zu können.[11]

Nach Lembke et al. [17] gibt es verschiedene Herausforderung innerhalb von Teams, wie die Arbeitsformen, Zusammenarbeit und die eigenen Vorstellungen betrachtet werden sollten. Daher empfiehlt Lembke von leitenden Personen innerhalb eines Teams, auch weiche Team-Faktoren bei der Ideengenerierung mit Design Thinking betrachtet werden. Somit müssen bei Kollaborationen in Teams stärker auf die Kommunikation und Wissensverteilung geachtet werden [17].

Um eine Gleichberechtigung und demokratische Bewertungsgrundlage zu gewährleisten sollte es eine vertrauensbasierte Team-Kultur geben. Dabei sollten Ideen bei der Ideengenerierung gleichwertig und unabhängig von Hierarchien betrachtet werden [11].

3 Methodologie

Nach den Ergebnissen der Literaturrecherche aus Kapitel 2.7 wird im Folgenden die Methodologie zur Beantwortung der For-

schungsfrage RQ2 aus Kapitel 1.2 vorgestellt. Die Forschungsfrage RQ2 stellt die Frage, wie erfolgreich die Dokumentation, Transparenz, Rückverfolgbarkeit und Selbstwirksamkeit bei der Ideengenerierung in Unternehmen innerhalb von Teams umgesetzt wird. Um die Forschungsfrage beantworten zu können wurde eine Umfrage auf der Basis der Erkenntnisse der Literaturrecherche entworfen und durchgeführt. Mit der Umfrage wurde der Stand innerhalb von Unternehmen aufgezeigt.

3.1 Hypothesen

Um die Forschungsfrage RQ2 aus Kapitel 1.2 beantworten zu können, wurden die Erkenntnisse aus Kapitel 2 zur Aufstellung der Hypothesen verwendet. Den Hypothesen sind, wie in Tabelle 1 zu sehen ist, jeweils die Fragen aus der Umfrage zu Design Thinking in Unternehmen zugeordnet. Die Hypothesen (siehe Tabelle 1) sind jeweils den verschiedenen Teilbereichen, wie die Dokumentation, Transparenz, Rückverfolgbarkeit, Selbstwirksamkeit von Ideen zuzuordnen.

Tabelle 1. Aufgestellt Hypothesen zur Beantwortung der Forschungsfrage mit dem zugehörigen Frageindex

Hypothese	iF
H_{01}: Die Mehrzahl der Teams in Unternehmen arbeiten interdisziplinär.	F04, F09
H_{02}: Die Ideengenerierung innerhalb der Ideation-Phase wird nicht ausreichend dokumentiert.	F09, F10
H_{03}: Eine Rückverfolgbarkeit des Weges zu einer Idee in der Ideation-Phase ist oft nicht möglich.	F09, F10
H_{04}: Ein Großteil der Unternehmen setzt keine Tools zur Dokumentation der Ideation-Phase ein.	F09, F10
H_{05}: Der Erfolg einer Idee wird vom Großteil der Unternehmen nicht gemessen.	F09, F10
H_{06}: Nicht alle Ideen werden von allen Mitgliedern eines Teams gleich gewürdigt.	F11
H_{07}: In Teams werden die Ideen, die zu einer Umsetzung führen nicht demokratisch ausgewählt.	F11
H_{08}: Design Thinking wird selten mit anderen agilen Prozessen innerhalb von Teams kombiniert.	F08

3.2 Durchführung Analyse der Datenerhebung

Für die Methode der Datenerhebung wurde eine Umfrage ausgewählt. Die Fragen wurden dabei auf der Basis der Hypothesen aus Kapitel 3.3 aufgestellt. Um ein differenziertes Bild zur Validierung oder Falsifizierung der Hypothesen zu erhalten werden qualitative und quantitative Daten erhoben. Die quantitativen Daten wurden dabei mit einer 7-skaligen Likert-Skala von 1(trifft nicht zu) bis 7(trifft voll zu) zu Erhebung dargestellt. Für die Erhebung der qualitativen Daten wurde auf die Methode der offenen Fragen zurückgegriffen. Diese können zu den quantitativen Daten ein zusätzliches Bild an praxisnahen Erfahrungen der TeilnehmerInnen der Umfrage abbilden.

Die Umfrage wurde mit Hilfe des Umfragetools LimeSurvey online durchgeführt. TeilnehmerInnen wurden über Aluminikontakte der Hochschule Reutlingen und entsprechenden Unternehmen, die Design Thinking anwenden, akquiriert. Die Umfrage wurde in 3 Teile gegliedert, mit einer Einführung (8 allgemeine Fragen), spezifischen Fragen (22) und offenen Fragen (5), sowie eines Schlussteils (2 Fragen).

Um eine deskriptive Analyse der der quantitativen Daten vorzunehmen, wurde eine Auswertung der Daten in Excel vorgenommen. Die quantitativen Daten wurde dabei zum Teil in Balken Diagramme für die einführenden und abschließenden Fragen ausgewertet. Für die spezifischen Fragen zur Dokumentation und Selbstwirksamkeit wurden die TeilnehmerInnen in Prozentwerte konvertiert und in Form von gestapeltem Balkendiagrammen ausgewertet. Anschließend an die deskriptive Analyse wurden die Ergebnisse mit den Hypothesen verglichen.

4 Ergebnisse

4.1 *Ergebnisse aus der quantitativen Erhebung*

Insgesamt haben n= 25 TeilnehmerInnen an der Umfrage teilgenommen. Davon haben 6 die Umfrage abgebrochen und somit keine vorhandenen Daten hinterlassen. Von den insgesamt n=19 verbliebenen TeilnehmerInnen haben 3 kein Design Thinking eingesetzt. Innerhalb der Umfrage kam heraus, dass ein Großteil der TeilnehmerInnen aus Großunternehmen stammen und im IT-Bereich tätig sind, siehe Abbildung 3. Im Gegensatz dazu sind n=2 jeweils in Start-Ups oder Kleinunternehmen tätig. Bei der Ergebniserreichung und der Größe der Teams verteilen sich dabei die TeilnehmerInnen, wie in Abbildung 3, bei F03 und F04 zu sehen ist gleichermaßen. Bei dem Großteil der TeilnehmerInnen (n=16) wurde Design Thinking eingesetzt. Aufgefallen ist, dass bei einem Großteil der TeilnehmerInnen Design Thinking erst seit 1 Jahr oder mehr als 3 Jahren eingesetzt. Nur n=3 TeilnehmerInnen wenden Design Thinking mehr als 5 Jahre innerhalb des Teams an. Die Ergebniserreichung erfolgt überwiegend innerhalb der Teams und interdisziplinär (siehe Abbildung 3). In Bezug auf die interdisziplinäre Rollenverteilung bei Design Thinking, lässt sich, dass innerhalb der Praxis in Unternehmen bestätigen. Vorzugsweise wird der Design Thinking Ansatz bei der Anwendung mit Scrum oder Agiler Softwareentwicklung verknüpft. Ein weiterer agile Prozess der häufig Anwendung findet sind Design Sprints Die Kombination von Design Thinking, mit Scrum oder agiler Softwareentwicklung, lässt sich auf Grund der Daten, vor allem wegen der hohen TeilnehmerInnen aus dem IT-Bereich rückschließen. Bei der Rollenverteilung der TeilnehmerInnen, siehe Abbildung

Abbildung 3: Übersicht über die allgemeinen Fragen der Einführungs- und des Schlussteils der Umfrage

3, ist eine ausgeglichene Verteilung zu erkennen. Sonstige Rollen sind an Hand der Umfragedaten die Rolle des Business Designer oder übergeordnet ein Chief Technology Officer.

Bei der Auswertung der Dokumentation der von Ideen, siehe Abbildung 4, konnte festgestellt werden, dass eine Dokumentation von Beiträgen innerhalb von Teams stattfindet. Auch bezüglich der Transparenz und Rückverfolgbarkeit sind diese vorwiegend gegeben. Lediglich bei der Mitnahme dokumentierter Ideen in andere Prozesse oder Teams, ist keine klare Mehrheitsverteilung erkennbar. Eine ähnliche Situation bildet sich bei der Honorierung von Ideen und der Messung von Ideen durch Software-Tools, siehe Abbildung 4. Betrachtet man die Selbstwirksamkeit und Gleichberechtigung von Ideen, wurden die Aussagen aus Abbildung 5 zum Großteil nicht bestätigt. Bei einer deutlichen Mehrheit werden die Ideen innerhalb von Teams gleichbehandelt. Bei der Bewertung der Qualität der Beiträge gibt es keine eindeutige Mehrheitsverteilung.

Vergleicht man die Ergebnisse mit den aufgestellten Hypothesen aus Kapitel 3.1 lässt sich feststellen, dass H_{02} bis H_{08} falsifiziert wurden und H_{01} validiert wurde.

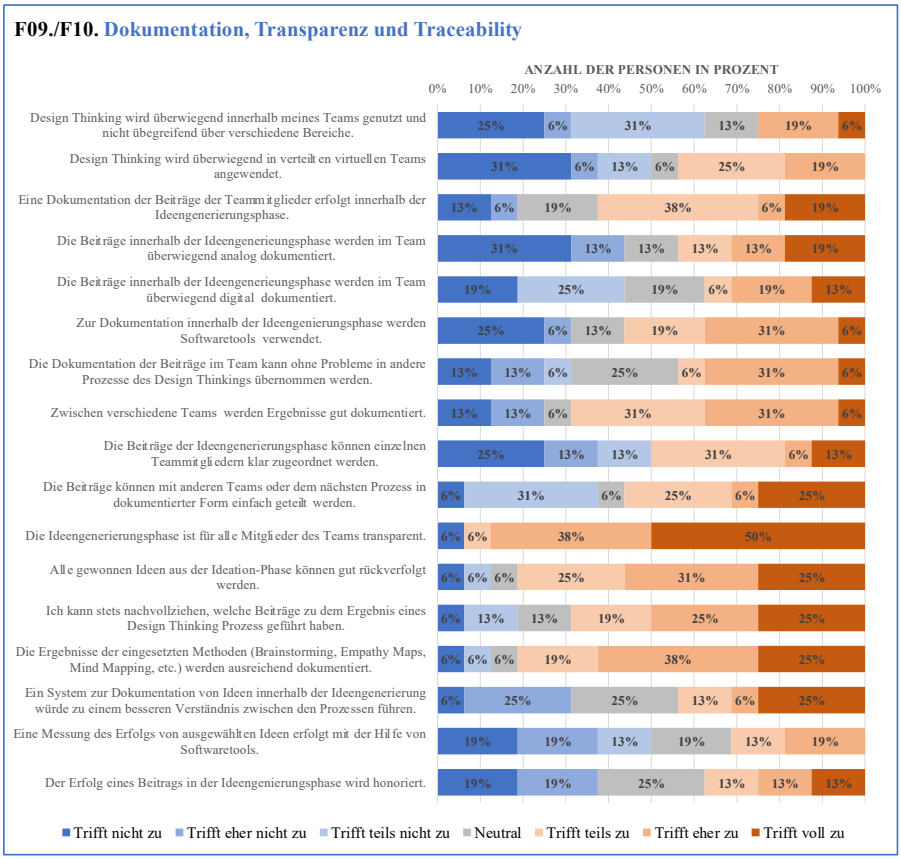

Abbildung 4: Dokumentation, Transparenz und Traceability bei der Ideengenerierung

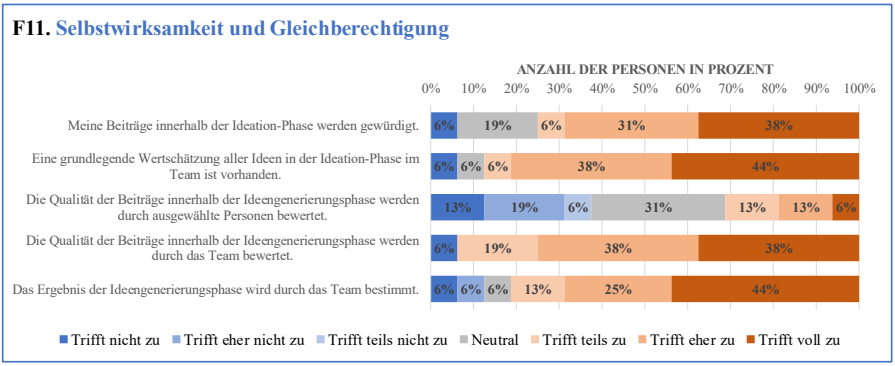

Abbildung 5: Selbstwirksamkeit und Gleichberechtigung bei der Ideengenerierung

Insgesamt lässt sich bei den falsifizierten Hypothesen in Bezug auf die IT-Unternehmen bei der Dokumentation, Transparenz und Rückverfolgbarkeit von Ideen schließen, dass diese bei der Verwendung von Design Thinking gegeben sind. Diese Tendenz bezüglich der Umfrage in Unternehmen, zeigt einen Unterschied zu den aus der Literatur gefunden Herausforderungen bezüglich Design Thinking im Bildungsbereich auf.

4.2 Ergebnisse aus der qualitativen Erhebung

Innerhalb der offenen Fragen bezüglich der Selbstwirksamkeit und Gleichberechtigung von Ideen in Teams konnten unterschiedliche Meinungen aufgefasst werden. Ein Teil der TeilnehmerInnen schlägt zum Beispiel die Anonymisierung von Ideen vor, um eine demokratischere Bewertung der Ideen zu gewährleisten. Zudem führen flache Hierarchien zu einer besseren Gleichbehandlung der Ideen. Auf der anderen Seite wird eine Moderation der Ideen empfohlen, um für eine Gleichbehandlung der Ideen zu sorgen. Im Vergleich dazu kam heraus, dass die Gleichbehandlung der Ideen nicht unbedingt als wichtig aufgefasst wird, da das Team entscheidet, welche Idee gut oder schlecht ist.

Nach den offenen Antworten sind eine bessere Wirksamkeit von Ideen durch Dokumentieren und Messen für eine bessere Motivation und Messung des Erfolgs hilfreich. Dagegen sprechen sich verschiedene Beiträge der Umfrage aus, da die Ursprungsidee nicht so wichtig ist, wie das Gesamtergebnis.

Bei der Dokumentation schlagen die TeilnehmerInnen den Einsatz von digitalen Tools vor, um die Ideen zu dokumentieren. Dies sollte allerdings unter dem Hintergrund geschehen, dass die Dokumentation der Ideengenerierung nicht im Weg steht und den Prozess behindert. Der Fokus sollte dabei aber stets auf dem wesentlich und wichtigen liegen: der Problemlösung. Allgemein lässt sich zusammenfassen, dass auch analoge Ergebnisse dokumentiert werden sollten, die Dokumentation aber nur so groß wie möglich sein sollte.

Für eine bessere Transparenz von Ideen sprechen laut den Antworten eine Kennzeichnung von Ideen durch Tags oder Farbelemente, sowie Abstimmungen der Ideen an Whiteboards, durch Gallery Walks.

5 Zusammenfassung

In der Arbeit zu Design Thinking in Unternehmen wurde die Dokumentation, Transparenz, Traceability und Selbstwirksamkeit im Zusammenhang mit Teams untersucht. Die durchgeführte Literaturrecherche hat Erkenntnisse über den Stand in dem Kontext von Design Thinking hervorgebracht. Im weiteren Verlauf konnten mit einer Umfrage erste Untersuchungen des Themas bezüglich der Situation in Unternehmen gefasst wer-

den. Durch die Literaturrecherche und Umfrage konnten Defizite bei der Dokumentation, Transparenz und Traceability, sowie bei der Selbstwirksamkeit und Gleichberechtigung unter anderem im Bildungsbereich mit Design Thinking festgestellt werden. Die Forschungsfrage RQ2 konnte mit Hilfe der Umfrage größtenteils beantwortet werden. Dabei konnte festgestellt werden, dass von den TeilnehmerInnen die Mehrzahl Design Thinking anwenden und mit anderen agilen Methoden verknüpfen. Des Weiteren wurde festgestellt, dass die Dokumentation, sowie Transparenz und Traceability zusammen mit der Selbstwirksamkeit in Teams im IT-Bereich gegeben ist und es wenige Defizite diesbezüglich gibt.

6 Fazit

Vergleicht man die Erkenntnisse aus der Literaturrecherche mit den Ergebnissen der Umfrage, wurden die Hypothesen weitgehend falsifiziert. Vor allem in Unternehmen im IT-Bereich. Findet die Dokumentation, Transparenz, Rückverfolgbarkeit und Selbstwirksamkeit von Ideen statt. Damit wurden auch die Kernaussagen der Dokumentation und Demokratie bei der Ideengenerierung, nach Design Thinking bestätigt. Da es eine Diskrepanz zur Dokumentation und Transparenz im Bildungsbereich gibt, wäre es empfehlenswert in Richtung alternativer Unternehmensarten, ab vom IT-Bereich, eine konkrete Fragestellung bezüglich der Dokumentation zu untersuchen. Um ein vergleichbares Ergebnis zu erhalten, könnte die Umfrage auf soziale Netzwerke erweitert werden. Alternativ könnten auch direkte Experteninterviews innerhalb von Teams in Unternehmen zu solchen Erkenntnissen führen. Anschließend and die Dokumentation, Transparenz und Rückverfolgbarkeit ergeben sich zwei weitere interessante Fragestellungen. Zum einem, wie die Transparenz und Traceability in Bezug auf den Handover-Prozess von Ideen weiter verbessern könnte. Ein zweiter Punkt, der sich ergibt, ist die Mobilität der Ideengenerierung in Bezug auf die Dokumentation. Wie können beispielweise Ideen analog und digital weiterverwendet werden, um diese zwischen Teams und variablem Raum zu transportieren? Diese Fragen sollen in einer anschließenden Arbeit genauer untersucht werden.

Literaturverzeichnis

[1] T. Beyhl, G. Berg und H. Giese, Eds. 2013. *Towards documentation support for educational design thinking projects. Proceedings of the 15th International Conference on E&PDE, Dublin, Dublin Institute of Technology, Bolton Street, Dublin, Ireland.*

[2] T. Beyhl, G. Berg und H. Giese. 2013. Why innovation processes need to support traceability. In *International Workshop on TEFSE, 2013. 19 May 2013, San Francisco, CA, USA*

[3] T. Beyhl, G. Berg und H. Giese. 2014. Connecting Designing and Engineering Activities. In *Design Thinking Research. Building Innovation Eco-Systems*, Springer International Publishing, Cham, s.l., 153–182.

[4] T. Beyhl und H. Giese. 2015. Traceability Recovery for Innovation Processes. In *2015 IEEE/ACM 8th International Symposium on SST 2015. Florence, Italy, 17 May 2015*. IEEE, Piscataway, NJ, 22–28.

[5] T. Beyhl und H. Giese. 2016. Connecting Designing and Engineering Activities III. In *Design thinking research. Making design thinking foundational*. Springer, Cham, Heidelberg, New York, Dordrecht, London, 265–290.

[6] S. Bitzer und B. Werther. 2019. Herausforderungen und Lösungsansätze durch den Einsatz von digitalen Zusammenarbeitssystemen im Wissensmanagement in einem globalen Mehrmarken-Konzern. *HMD* 56, 1, 109–120.

[7] W. Brenner und F. Uebernickel, Eds. 2016. *Design Thinking for Innovation. Research and Practice* (1st ed. 2016). Springer International Publishing, Cham.

[8] A. Diehl. 2018. Design Thinking – Mit Methode komplexe Aufgaben lösen und neue Ideen entwickeln. *Andreas Diehl*.

[9] F. Dobrigkeit und D. de Paula. 2019. Design thinking in practice. In *ESEC/FSE '19*. The Association for Computing Machinery, Inc, New York, NY, 1059–1069.

[10] D. Dunne. 2018. Implementing design thinking in organizations: an exploratory study. *J Org Design* 7, 1.

[11] S. Gibbons. 2016. *Design Thinking Builds Strong Teams* (2016). Retrieved July 10, 2020 from https://www.nngroup.com/articles/design-thinking-team-building/.

[12] Hasso Plattner Institut. 2020. *Was ist Design Thinking? | HPI-Academy* (July 2020). Retrieved July 15, 2020 from https://hpi-academy.de/design-thinking/was-ist-design-thinking.html.

[13] Hasso Plattner Institut. 2020. *What is Design Thinking? - Design Thinking - Hasso Plattner Institute* (July 2020). Retrieved July 15, 2020 from https://hpi.de/en/school-of-design-thinking/design-thinking/what-is-design-thinking.html.

[14] Hasso Plattner Institute of Design. *An Introduction to Design Thinking PROCESS GUIDE*. Retrieved July 13, 2020 from http://web.stanford.edu/~mshanks/MichaelShanks/files/509554.pdf.

[15] H. Hilbrecht und O. Kempkens. 2013. Design Thinking im Unternehmen – Herausforderung mit Mehrwert. In *Digitalisierung und Innovation*. Springer-Gabler, Wiesbaden, 347–364.

[16] J. Hofer, T. Schoormann, J. Kortum, und R. Knackstedt. 2019. „Ich weiß was ihr letzte Sitzung getan habt" – Entwicklung und Anwendung eines Softwarewerkzeuges zur Dokumentation von Design Thinking-Projekten. *HMD* 56, 1, 160–171.

[17] T.-B. Lembcke, A. B. Brendel und L. M. Kolbe. 2019. Make Design Thinking Teams Work: Einblicke in die Herausforderungen von innovativen Team-Kollaborationen. *HMD* 56, 1, 135–146.

[18] A. Menning, T. Beyhl, H. Giese, U. Weinberg und C. Nicolai. 2014. Introducing the LogCal: Template-Based Documentation Support for Educational Design Thinking Projects. In *Design education & human technology relations. Proceedings of the 16th International Conference on E&PDE, University of Twente, Enschede, The Netherlands, 4th-5th September 2014*.

[19] S. Ney und C. Meinel. 2019. *Putting Design Thinking to Work*.(1st ed. 2019). Understanding Innovation. Springer International Publishing; Imprint: Springer, Cham.

[20] H. Plattner, C. Meinel und U. Weinberg. 2011. *Design Thinking. Innovation lernen - Ideenwelten öffnen* (Nachdr). mi-Wirtschaftsbuch, München.

[21] D. R.A. Schallmo und K. Lang. 2020. *Design Thinking erfolgreich anwenden. So entwickeln Sie in 7 Phasen kundenorientierte Produkte und Dienstleistungen* (2nd ed. 2020).

Accessibility Evaluation Tools for Android Mobile Applications

Jessica Giebel
Hochschule Reutlingen
Jessica.Giebel@Student.Reutlingen-University.de

Abstract

Accessibility is an important topic in everyday life, for example elevators in buildings for people with physical disabilities or traffic lights with acoustic signals for visually impaired people. Since smartphones are also becoming a more important part of daily life, mobile applications have to be accessible for everyone. Evaluation tools can be used to evaluate the accessibility. These should help developers in the different stages of their development process to design accessible applications. In this work, five different accessibility evaluation tools for Android mobile applications are examined. Based on the findings, recommendations for developers are given regarding the accessibility evaluation tools.

CCS Concepts

• **Human-centered computing → Accessibility design and evaluation methods**; **Accessibility systems and tools**;

Keywords

Mobile Accessibility, accessibility evaluation, accessibility evaluation tools

1 Introduction

In Germany, the number of smartphone users has increased from 51 million to 57.7 million between 2016 and 2019 [1]. This demonstrates that an increasing number of people are using a smartphone. Most people use mobile applications (apps) for entertainment, communication with other people and for obtaining information on the Internet [2]. Apps must therefore be designed so that all people can operate them. In this context, *Mobile Accessibility* and its evaluation is becoming more important. Thus, *accessibility evaluation tools* are developed to support developers creating accessible apps. The goal of this work is to help developers to find the right tool for their needs in order to design accessible apps and improve existing apps. A selection of accessibility evaluation tools are analysed using various criteria that have emerged from the analysis.

First, the term accessibility is introduced in chapter 2. Guidelines for accessibility are presented in chapter 3 and chapter 4 summarizes accessibility evaluation. In chapter 5, the method used in this work is presented followed by the introduction of the examined evaluation tools in chapter 6. The result of the analysis is discussed in chapter 7. Chapter 8 contains a conclusion of the work.

2 Accessibility

Accessibility is becoming a more important topic in society. However, there are many different definitions for this term. According to the German Behindertengleichstellungsgesetz (BGG, engl. Equal Opportunities for

Academic supervisor: Prof. Dr. rer. nat. Gabriela Tullius
Hochschule Reutlingen
Gabriela.Tullius@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

People with Disabilities Act) something is accessible, if it can be found, accessed and used by people with different disabilities (section 4 BGG). Necessary aids regarding the impairment have to be supported. The intended areas of accessibility are specified as all areas of life created by humans such as buildings, objects, services or information. This also involves the Internet and everything related to it. An important term in this context is Web accessibility. Websites have to be designed in a manner that all people can access and interact with them, as well as with necessary assistive technologies [3]. Mobile accessibility extends Web accessibility. It relates to the use of websites and applications on mobile devices [4]. Mobile accessibility adopts many aspects from Web accessibility, but also adds other issues. For example, mobile devices are provided with a touchscreen, which is much smaller compared to a computer screen. The aim of Web and mobile accessibility is to design websites and apps in a way that all people can use them. The Web Accessibility Initiative (WAI) defines five categories for the different types of impairments, which should be considered during web development and, consequently, mobile applications development: *auditory*, *cognitive*, *physical*, *speech* and *visual* [5]. These categories are also important for developing mobile apps.

According to the Federal Statistical Office, 7.9 million severely disabled people were living in Germany at the end of the year 2019, which equates to a proportion of 9.5 % of the population [6]. In addition, many people have slight impairments like a mild vision loss. There can be various reasons for an impairment. Since society is getting older, it is important to consider people with afflictions of old age as well [5]. Assistive technologies are used to enable people with disabilities to take part in everyday life, despite their limitations [7]. They include solutions to enable people with different impairment types to access websites and mobile apps and thus should be supported by various devices. For example, a technology, that is often used by people with visual impairments, is the screen reader. It captures the visual information on the screen and reads it out to the user. Moreover, keyboard control is often used instead of mouse control [8].

3 Guidelines

Over the years, guidelines have been created so that developers can use these for orientation to design more accessible content. In the following, the most popular guidelines for websites and mobile applications are introduced.

3.1 Websites

Guidelines for mobile applications are often based on the guidelines for websites. For that reason, the Web Content Accessibility Guidelines (WCAG), the most common guidelines, are explained in this chapter. The WCAG are published by the WAI of the World Wide Web Consortium (W3C). The first version was published in 1999 [9] and the latest version WCAG 2.1 in 2018 [10]. The version 2.2 is currently in process. The WCAG 2.0 is an international standard for accessible web development: ISO/IEC 40500:2012 [11]. The WAI established four principles for web accessibility: *perceivable*, *operable*, *understandable* and *robust* [12]. Each principle includes specific guidelines, for example text alternatives for non-text content to create perceivable information [12]. In addition, each guideline has success criteria, which are considered requirements. These success criteria are testable and are divided into three levels of conformance. Success criteria on Level A are basic criteria, which have to be fulfilled. Advanced criteria are classified into Level AA. Level AAA is the highest level of conformance, but this level is not recommended as standard by the WAI [13]. Corresponding techniques exist to each success criteria, divided into sufficient and advisory [12].

Web accessibility is also an issue in legislation. The Barrierefreie-Informationstechnik-Verordnung (BITV, engl. barrier-free information technology ordinance) is an ordinance in Germany. The last version (BITV 2.0) was published in 2011 and last edited in 2019. The aim is to create accessible information and communication technology especially for public departments. These include, for example, websites and mobile applications. In section 3 (1) BITV, the four principles of WCAG are mentioned as requirements for accessible technology.

3.2 Mobile applications

Special guidelines are also needed for mobile devices. In contrast to a desktop computer or laptop, a mobile device has a much smaller screen. Nevertheless, many accessibility guidelines related to web can be transferred to mobile. The WAI published a document in 2015, which explains how WCAG can be applied to mobile devices and applications. In particular, contrast is an important characteristic for accessible design of mobile applications. The reason for this is that mobile devices are used in various environments with different lighting conditions. Due to the small screen size, the contrast ratio has to be adapted to the font size. Further guidelines are defined that apply specifically to mobile devices, for example touchscreen gestures. They range from a simple tap on a button up to complex movements with multiple fingers. These can be an obstacle for physically impaired people. So, touchscreen gestures should be as simple as possible [14].

In this work especially apps for the operating system Android are considered. Google, to whom Android belongs, published own guidelines for development of accessible apps. They defined three basic guidelines for accessibility. The first guideline recommends that text visibility should be enhanced by paying attention to contrast ratio. As in WCAG, the guiding value for contrast ratio depends on text formatting. The second guideline is to use large and simple controls. The width and height of a touchable target should be at least 48dp x 48dp. The third guideline describes that each user interface (UI) element should contain a description. Screen readers read out these descriptions to the user. In the description, the purpose of the element is outlined, which should be simple and unique for each element [15]. An example for a screen reader for Android devices is TalkBack [8]. Based on these guidelines, Google presents best practices for the development of accessible apps. For example, according to the third guideline, headings should be marked as headings. This simplifies the navigation for users of assistive services [16].

4 Accessibility evaluation

The guidelines show how to design accessible apps, but evaluating the app is also important in order to investigate if the app is actually accessible for all people. During development, issues like missing alternative texts for images are not immediately obvious in hundreds of lines of code. To test the accessibility of a mobile app, manual evaluation methods can be used. One possibility is to let selected users explore the app. The inspections are guided and the accessibility problems are recorded. It is important that users with assistive technologies are also considered [17], [18]. Moreover, an expert can evaluate the app on his own and inspect each site and element using accessibility guidelines [17]. Conducting these experiments is very resource-intensive and difficult to reproduce [18].

For this reason, more attention is being paid to automated evaluation methods. These are already used, for example, to test software. Time exposure and cost are reduced and more code can be covered. The test cases are reusable and more reliable [19]. Automated tools are also increasingly used to evaluate accessibility. Therefore, the tools are oriented

to accessibility guidelines. For example, success criteria of the WCAG can be tested by such a tool.

5 Method

In this work, accessibility evaluation tools for Android mobile apps are analysed. This decision was made to ensure that there are no problems due to different operating systems when comparing the tools. This review focuses on native Android mobile apps on smartphones to create a consistent framework condition. Native means that the mobile app is specialized on a particular operating system [20]. For research, the digital library *Web of Science*, *ACM Digital Library* and *IEEE Xplore digital library* were used. The following search keywords were utilised and combined with each other in various ways: mobile accessibility, accessibility evaluation, accessibility evaluation tools, Android apps, Android applications, automatic evaluation and mobile application. Through use of the introduced libraries and search keywords, the most important works were identified. Further relevant sources were found by cross-referencing. A list of eleven possible tools for the comparison was created and finally five tools were selected, which are presented in chapter 6. These tools were selected in order to be able to compare static as well as dynamic analysis tools with different degrees of automation and to identify their similarities and differences.

5.1 Test environment

An *Android Emulator* in Android Studio was used to run the evaluation tools, because some tools analyse the source code of an app and this is not feasible on a real device. The Android Virtual Device is a Pixel 3 smartphone with a screen size of 5,46 inches and Android 10.0 as operating system. Open source apps are installed on the virtual device and used to test the tools. The first app is called *Counter App*. The code is originally used for a Google Codelab which deals with the development of accessible apps and is available on GitHub[1]. This app purposefully contains accessibility flaws. The second app is the open source app *Simple Calendar Pro*, which is available on F-Droid[2]. This is a calendar, where various appointments can be entered and organized.

5.2 Analysis criteria

The accessibility evaluation tools are examined using various criteria. First, the minimum *Android version* required is identified. In addition, the tool is examined for the *guidelines* that are considered. The next category *disabilities* describes, which type of disabilities are included or excluded by the tool. Besides, the *approach* for the analysis is presented. The tools are classified in one of the two categories: static analysis, which investigates the source code without execution of the app, and dynamic analysis, which examines the app during runtime. This classification was already done in the work of Silva et al. [18], but was not adopted in detail for this work.

Moreover, the *degree of automation* describes in which extent the tool is automated. For that reason, a scale which ranges from one to five is used in the comparison which describes the degree of automation. Value one is a manual evaluation, which is not usually done with certain evaluation tools. Value three describes semi-automatic tools, where the user has to perform an action such as clicking a button. Value five is the highest degree of automation and the user only needs to activate the tool without performing further actions. Two and four are intermediate values. Finally, the *result display* is examined. It describes how the results of the analysis are communicated to the user. Each criterion obtains a *weighting factor* according to its importance for the evaluation of accessibility.

[1] https://github.com/googlecodelabs/android-accessibility (Retrieved 20.08.2020)

[2] https://f-droid.org/de/packages/com.simplemobiletools.calendar.pro/ (Retrieved 27.08.2020)

Guidelines and considered disabilities as well as degree of automation and result display of the tool are more important than Android version and the used approach for the analysis. So they are weighted twice.

6 Evaluation tools

In the following, five evaluation tools are introduced: Accessibility Scanner, Android Lint, Espresso, Mobile Accessibility Testing and Accessibility Engine for Android.

6.1 Accessibility Scanner

Accessibility Scanner is a mobile app by Google LLC and is available on Google Play[3]. It is based on the Accessibility Test Framework for Android (ATF)[4], also created by Google. Once Accessibility Scanner has been activated, the desired app can be checked for accessibility. A single screen is scanned by taking a snapshot and multiple screens can be scanned by recording the screen while interacting with the app [21].

Android version: API 23[3].
Guidelines: Scans are performed for the following categories: Content labelling, implementation, touch target size and contrast. Assistive technologies like screen readers need content labels to work correctly. Furthermore, barriers in implementation are identified such as the correct marking of links. In addition, touch target size should be at least 48 x 48 dp and contrast ratio of text and images should not be lower than 3.0. The settings for touch target size and contrast ratio can be modified [21]. As mentioned, the scans are based on the Accessibility Test Framework. On GitHub, Google has linked the WAI website for information about mobile accessibility. It can therefore be assumed that these guidelines are based on the WCAG.
Disabilities: Concerning the guidelines, barriers for visually and physically impaired people are identified by the tool.
Approach: Dynamic analysis is used. The tool scans the app during runtime, recording the screen.
Degree of automation: The degree of automation is three, because the tool automatically scans the different screens while manually interacting with the app.
Result display: If problems were found, rectangles are drawn around this item within the recorded snapshots. Every rectangle includes advices concerning the accessibility guidelines. These snapshots are stored in the app. Furthermore, the report can be shared as a ZIP file.

6.2 Android Lint

Android Lint is a code scanning tool that is integrated in Android Studio. It is available as a standalone tool too. The project files are analysed for different categories such as security or usability. Accessibility is also a category and therefore can be used for accessibility evaluation. In Android Studio, Lint automatically checks the code for all categories [22].

Android version: No specific Android version required.
Guidelines: Lint scans the code for five different criteria, which are described in Android Studio. Clickable Views are checked for their correct implementation of click actions. In addition, images are checked for missing content descriptions. Furthermore, clickable widgets should always be focusable as well, so that the user can interact with them using a keyboard. It is important that an empty and editable text field has a hint, so that the user understands the intention of this field. The method getContentDescription() should not be overridden, because this method is used by some other technologies.

[3] https://play.google.com/store/apps/details?id=com.google.android.apps.accessibility.auditor&hl=en (Retrieved 22.08.2020)
[4] https://github.com/google/Accessibility-Test-Framework-for-Android (Retrieved 22.08.2020)

Disabilities: The guidelines focus on people with visual and physical disabilities, which use screen readers and keyboards.
Approach: Static analysis is used. The tool scans the source code.
Degree of automation: The user only has to activate the Lint tool. In this case, the degree of automation is five.
Result display: The result is presented in a separate window in Android Studio. The different categories and their issues can be viewed individually. To every criterion, the concerned file and the line in the code is showed as well as a description.

6.3 Espresso

Espresso is a testing framework for Android, developed by Google. With Espresso, UI tests can be provided [23]. These tests are written in separate scripts, which can be written manually or using the Espresso Test Recorder [24]. With the Espresso Test Recorder, the user can click through the app and set assertions on View elements. Based on this, a UI test is automatically generated [24]. Espresso also supports testing for accessibility by integrating ATF in the existing test [25].

Android version: API 10 [23].
Guidelines: Since Espresso uses ATF, the guidelines of the Accessibility Scanner also apply here: Content labelling, implementation, touch target size and contrast.
Disabilities: According to the guidelines, the needs of visually and physically impaired people are considered.
Approach: Dynamic analysis is used. The tool scans the app during runtime by test scripts.
Degree of automation: The user has to either write the test script on his own or generate the test scripts by using the Espresso Test Recorder and clicking through the app. So, Espresso is almost a semi-automated tool with value two.
Result display: The result is presented in a separate window in Android Studio. The test will fail if there are any accessibility errors and the type of error is displayed in the result.

6.4 Mobile Accessibility Testing

The Mobile Accessibility Testing (MATE) tool is based on ATF and UI Automator framework[5] to create UI tests. This project is still under development and a prototype is used for this work (M. M. Eler, personal communication, 29 September 2020). After MATE is installed, the app to be examined have to be executed and MATE can be started. The tool uses randomly generated user interactions to find accessibility flaws in the app [26].

Android version: API 28 (M. M. Eler, personal communication, 08 October 2020)
Guidelines: MATE uses various guidelines of WCAG like contrast minimum, target size or non-text content and guidelines of the British Broadcasting Corporation (BBC) [27] for mobile accessibility.
Disabilities: With this tool, barriers for people with visual and physical impairments can be detected [26].
Approach: Dynamic analysis is used. The tool scans the app during runtime using a random strategy.
Degree of automation: The tool only has to be started and the user interactions are automatically generated for the accessibility test. Thus, it is an automated tool and the degree of automation is five.
Result display: MATE generates a CSV file with all accessibility flaws found. The CSV contains further information for every flaw, for example the size of an element if it is too small. Furthermore, snapshots are taken during the scanning process. On the

[5] https://developer.android.com/training/testing/ui-automator (Retrieved 07.10.2020)

snapshot, the concerned element is marked and the type of error is displayed.

6.5 Accessibility Engine for Android

Accessibility Engine (axe) for Android is a mobile app by Deque Systems Inc. and is available on Google Play[6]. It is based on the inhouse library Axe Android[7]. As soon as the app is installed, the scan can be executed while the app, that should be examined, is running. Then the screen is scanned by axe.

Android version: API 21[6].
Guidelines: The guidelines of axe are mostly based on the WCAG 2.0 and WCAG 2.1 as well as other best practices. For example, contrast ratio, touch target size or the name of checkboxes[7]. Overall, there are 10 different guidelines.
Disabilities: The used guidelines in this tool mainly focus on people with visual and physical disabilities.
Approach: Dynamic analysis is used. The tool scans the app during runtime capturing a single screen.
Degree of automation: The user has to activate the tool to start the scan, but it happens only for one screen of the app. This means that the scanning process has to be restarted for each screen. So, the tool is almost semi-automated and gets a value of two on the scale.
Result display: The result is presented on a separate window at the bottom of the scanned app. In this window, the number of accessibility violations and adherences are shown. The user can get more information of what guidline has been violated by clicking on the arrows. The concerned item is marked and the results can be shared as a CSV file.

[6] https://play.google.com/store/apps/details?id=com.deque.axe.android (Retrieved 27.08.2020)
[7] https://github.com/dequelabs/axe-android (Retrieved 27.08.2020)

7 Discussion

The presented tools have differences and similarities. All tools except Android Lint are based on WCAG for accessible mobile devices and this shows that the standard is applied almost everywhere. The aim is to cover all needs of different types of disabilities. However, in relation to the tools analysed, the focus is set strongly on visually and physically impaired people. It can be assumed that these two groups are focussed since they have problems to read or touch UI elements on a small smartphone screen and are dependent on assistive technologies like screen readers and keyboard control. The tools analyse the UI of an app and not its content. People with cognitive disabilities, which for example need simple language, cannot be considered by these tools. For people with an auditory disease, evaluation tools have to look for captions for audio content used in an app. Probably people with speech disabilities are not considered by such tools, because it is not common to create an app with only voice interaction. This is usually an additional function that supports people who have difficulties to type or clicking on an UI element.
The majority of examined tools uses a dynamic analysis to detect accessibility flaws. Accessibility Scanner, Espresso and axe are semi-automated tools. Android Lint and MATE are automated tools, but Android Lint does not examine the apps from the perspective of the user, just from the source code. MATE generates the user interaction on its own to identify all possible accessibility flaws.
Moreover, Android Lint and Espresso are integrated in Android Studio and show their result in a separate window. Espresso presents the accessibility flaws while Android Lint additionally displays the affected file and line of code. It must be noted that Android Lint uses an analysis of the source code and Espresso utilizes UI tests for accessibility evaluation.

Accessibility Scanner shows further information directly on the device and the prototype of MATE as well as axe display the detailed result in a CSV file.

8 Conclusion

In this work, five different accessibility evaluation tools for Android mobile apps are analysed. They have different degrees of automation and methods to display the result of the evaluation. Android Lint and Espresso can be used during the development process, because they are integrated in Android Studio, which is a development environment for mobile apps. CSV files with further information about the flaws can be also helpful for developing accessible apps. Accessibility Scanner, axe and MATE can be applied to the finished product or any existing app. They can be used by users which are not developers and want to check an app for its accessibility as well as developers who are made aware of existing accessibility flaws by the tool in order to improve it.

Automated accessibility evaluation testing can reduce the amount of time and cost, but currently many tools only focus on visually and physically impaired people. During manual evaluation, barriers for other types of disabilities can be found like complex sentences, which are difficult to detect with automated evaluation. In future work, the opportunities to test accessibility criteria for people with other disabilities than visual and physical impairments should be investigated. Moreover, evaluation tools for other operating systems should be compared with tools for Android mobile apps.

References

[1] VuMa. (Nov. 13, 2019). Number of smartphone users in germany from january 2009 to 2019, Statista, [Online]. Available: https://www.statista.com/statistics/461801/number-of-smartphone-users-in-germany/ (visited on 09/29/2020).

[2] Statista. (May 1, 2020). Which kind of smartphone apps do you use regulary?, Statista, [Online]. Available: https://www.statista.com/forecasts/998679/smartphone-app-usage-by-type-in-germany (visited on 09/20/2020).

[3] H. Petrie, A. Savva, and C. Power, "Towards a unified definition of web accessibility", in *Proceedings of the 12th Web for All Conference*, ser. W4A '15, event-place: Florence, Italy, New York, NY, USA: Association for Computing Machinery, 2015, ISBN: 978-1-4503-3342-9. DOI: 10.1145/2745555.2746653. [Online]. Available: https://doi.org/10.1145/2745555.2746653.

[4] S. L. Henry and J. Brewer. (Mar. 1, 2019). Mobile accessibility at w3c, W3C Web Accessibility Initiative (WAI), [Online]. Available: https://www.w3.org/WAI/standards-guidelines/mobile/ (visited on 08/18/2020).

[5] S. Abou-Zahra. (May 15, 2017). Diverse abilities and barriers, W3C Web Accessibility Initiative (WAI), [Online]. Available: https://www.w3.org/WAI/people-use-web/abilities-barriers/ (visited on 08/03/2020).

[6] Statistisches Bundesamt. (Jun. 24, 2020). Schwerbehinderte menschen am jahresende, Destatis, [Online]. Available: https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Gesundheit/Behinderte-Menschen/Tabellen/geschlecht-behinderung.html (visited on 09/04/2020).

[7] S. Shahrestani, "Aging, disability, and assistive internet of things", in *Internet of Things and Smart Environments*. Cham: Springer International Publishing, 2017, pp. 1–9, ISBN: 978-3-319-60164-9. DOI: 10.1007/978-3-319-60164-9_1. [Online]. Available: http://link.springer.com/10.1007/978-3-319-60164-9_1 (visited on 09/05/2020).

[8] H. Nicolau and K. Montague, "Assistive technologies", in *Web Accessibility*, Y. Yesilada and S. Harper, Eds., Series Title: Human–Computer Interaction Series, London: Springer London, 2019, pp. 317–335, ISBN: 978-1-4471-7440-0. DOI: 10.1007/978-1-4471-7440-0_18. [Online]. Available: http://link.springer.com/10.1007/978-1-4471-7440-0_18 (visited on 09/04/2020).

[9] W. Chisholm, G. Vanderheiden, and I. Jacobs. (May 5, 1999). Web content accessibility guidelines 1.0, W3C, [Online]. Available: https://www.w3.org/TR/WAI-WEBCONTENT/ (visited on 08/09/2020).

[10] A. Kirkpatrick, J. O Connor, A. Campbell, and M. Cooper. (Jun. 5, 2018). Web content accessibility guidelines (WCAG) 2.1, W3C, [Online]. Available: https://www.w3.org/TR/WCAG21/ (visited on 08/12/2020).

[11] S. L. Henry. (Sep. 10, 2020). Web content accessibility guidelines (WCAG) overview, W3C Web Accessibility Initiative (WAI), [Online]. Available: https://www.w3.org/WAI/standards-guidelines/wcag/ (visited on 09/06/2020).

[12] B. Caldwell, M. Cooper, L. Guarino Reid, and G. Vanderheiden. (Dec. 11, 2008). Web content accessibility guidelines (WCAG) 2.0, W3C, [Online]. Available: https://www.w3.org/TR/WCAG20/ (visited on 08/07/2020).

[13] Understanding WCAG 2.0: Understanding Conformance. (2016). W3C, [Online]. Available: https://www.w3.org/TR/UNDERSTANDING-WCAG20/conformance.html (visited on 09/13/2020).

[14] K. Patch, J. Spellman, and K. Wahlbin. (Feb. 26, 2015). Mobile accessibility: How WCAG 2.0 and other w3c/WAI guidelines apply to mobile, W3C, [Online]. Available: https://www.w3.org/TR/mobile-accessibility-mapping/ (visited on 08/09/2020).

[15] Make apps more accessible. (Dec. 27, 2019). Android Developers, [Online]. Available: https://developer.android.com/guide/topics/ui/accessibility/apps (visited on 09/04/2020).

[16] Principles for improving app accessibility. (May 21, 2020). Android Developers, [Online]. Available: https://developer.android.com/guide/topics/ui/accessibility/principles (visited on 09/04/2020).

[17] M. Billi, L. Burzagli, T. Catarci, G. Santucci, E. Bertini, F. Gabbanini, and E. Palchetti, "A unified methodology for the evaluation of accessibility and usability of mobile applications", *Universal Access in the Information Society*, vol. 9, no. 4, pp. 337–356, Nov. 2010, ISSN: 1615-5289, 1615-5297. DOI: 10.1007/s10209-009-0180-1. [Online]. Available: http://link.springer.com/10.1007/s10209-009-0180-1 (visited on 08/14/2020).

[18] C. Silva, M. M. Eler, and G. Fraser, "A survey on the tool support for the automatic evaluation of mobile accessibility", in *Proceedings of the 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Infoexclusion - DSAI 2018*, Thessaloniki, Greece: ACM Press, 2018, pp. 286–293, ISBN: 978-1-4503-6467-6. DOI: 10.1145/3218585.3218673. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3218585.3218673 (visited on 07/02/2020).

[19] D. M. Rafi, K. R. K. Moses, K. Petersen, and M. V. Mantyla, "Benefits and limitations of automated software testing: Systematic literature review and practitioner survey", in *2012 7th International Workshop on Automation of Software Test (AST)*, Zurich, Switzerland: IEEE, Jun. 2012, pp. 36–42, ISBN: 978-1-4673-1822-8. DOI: 10.1109/IWAST.2012.6228988. [Online]. Available: http://ieeexplore.ieee.org/document/6228988/ (visited on 09/09/2020).

[20] T. Vilcek and T. Jakopec, "Comparative analysis of tools for development of native and hybrid mobile applications", in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia: IEEE, May 2017, pp. 1516–1521, ISBN: 978-953-233-090-8. DOI: 10.23919/MIPRO.2017.7973662. [Online]. Available: http://ieeexplore.ieee.org/document/7973662/ (visited on 09/21/2020).

[21] Accessibility Scanner. (2020). Android Accessibility Help, [Online]. Available: https://support.google.com/accessibility/android/faq/6376582?hl=en (visited on 09/04/2020).

[22] Improve your code with lint checks. (Aug. 26, 2020). Android Developers, [Online]. Available: https://developer.android.com/studio/write/lint (visited on 09/05/2020).

[23] D. Zelenchuk, *Android Espresso Revealed: Writing Automated UI Tests*. Berkeley, CA: Apress, 2019, ISBN: 978-1-4842-4315-2. DOI: 10.1007/978-1-4842-4315-2. [Online]. Available: http://link.springer.com/10.1007/978-1-4842-4315-2 (visited on 09/13/2020).

[24] Create UI test with Espresso Test Recorder. (Aug. 25, 2020). Android Developers, [Online]. Available: https://developer.android.com/studio/test/espresso-test-recorder (visited on 09/06/2020).

[25] Accessibility checking. (Dec. 27, 2019). Android Developers, [Online]. Available: https://developer.android.com/training/testing/espresso/accessibility-checking (visited on 09/04/2020).

[26] M. M. Eler, J. M. Rojas, Y. Ge, and G. Fraser, "Automated accessibility testing of mobile apps", in *2018 IEEE 11th International Conference on Software Testing, Verification and Validation (ICST)*, Vasteras: IEEE, Apr. 2018, pp. 116–126, ISBN: 978-1-5386-5012-7. DOI: 10.1109/ICST.2018.00021. [Online]. Available: https://ieeexplore.ieee.org/document/8367041/ (visited on 07/02/2020).

[27] E. P. Richens, G. F. Williams, J. Knight, M. Mathews, and R. Nancarrow. (2019). Mobile accessibility guidelines, Accessibility for Products, [Online]. Available: https://www.bbc.co.uk/accessibility/forproducts/guides/mobile/ (visited on 10/07/2020).

Intrusion Detection Systeme
Eine Einführung

Johannes Timotheus Zillig
Hochschule Reutlingen
Johannes_Timotheus.Zillig@Student.Reutlingen-University.de

Abstract
Dieser wissenschaftliche Artikel beschäftigt sich mit dem aktuellen Stand der IT-Sicherheit in Deutschland und eine der zahlreichen Möglichkeiten, sich dieser anzunähern. Dabei liegt hier das Augenmerk auf den sogenannten Intrusion Detection Systemen. Sie ermöglichen es den Anwendern, im täglichen IT-Betrieb, verdächtiges Verhalten und Angriffe zu erkennen, indem Daten, Ressourcen und Netzwerkströme analysiert werden. Durch vorangegangene Recherchen werden die verschiedenen Varianten, verfügbaren Detektionsarten und deren Arbeitsweisen kurz erläutert und vorgestellt. Dabei sollen vor allem die Funktionsweise der Systeme, deren Einsatzort und deren Grenzen im Fokus stehen. Ziel der Arbeit ist die Auswahl eines passendend Intrusion Detection Systems für die Hochschule Reutlingen, deren Rechenzentrum und der Computerlabore. Dazu wird, im Anschluss an die Vermittlung der Grundlagen, eine verkürzte Anforderungsanalyse präsentiert und das gewählte Intrusion Detection System, welches den Anforderungen am ehesten entspricht, vorgestellt. Im letzten Schritt folgt ein Fazit, zu den erarbeiteten Erkenntnissen.

CCS Concepts
• Security and privacy → Intrusion/anomaly detection and malware mitigation → Intrusion detection systems.

Keywords
IT-Sicherheit, Intrusion Detection System, Host-Based, Network-Based, Hybride-Systeme, Signatur-Basiert, Anomalie-Basiert

1 Einleitung
Dieses Kapitel bietet eine kurze Hinführung zum Thema der Intrusion Detection Systeme. Es wird die eigene Motivation und die aktuelle Lage der IT-Sicherheit in Deutschland herausgearbeitet. Im letzten Unterkapitel wird der weitere Aufbau der wissenschaftlichen Arbeit beschrieben.

1.1 Aktuelle Lage
Die aktuelle Lage der IT-Sicherheit in Deutschland, wie auch Weltweit, nimmt dramatische Formen an. Die Angreifer werden immer erfinderischer und bringen fast minütlich neue Angriffstechniken und -werkzeuge hervor [1]. Aus diesem Grund rückt die IT-Sicherheit immer mehr in den Fokus des Staates, der Wirtschaft und der Öffentlichkeit. Dabei ist das oberste Ziel die Information, Daten und Dienstleistungen aller Teilnehmer zu schützen und vor Manipulation zu bewahren. Dem entgegen steht, wie in Abbildung 1 zu sehen, der starke Anstieg neuer Schadprogramme bzw. Malware über die letzten Jahre.

Betreuer Hochschule: Prof. Dr.-Ing. Michael Tangemann
Hochschule Reutlingen
Michael.Tangemann@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

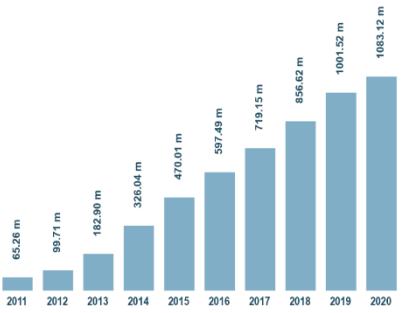

Abbildung 1: Entwicklung der registrierten Schadprogramme in den Jahren 2011 bis Aug. 2020 [2]

In dieser Auflistung wird jedoch nicht zwischen Trojanern, Würmern, Vieren, Multi-Plattform-Skripten etc. unterschieden. In den Jahren 2019 bzw. 2020 wurden im Durchschnitt 320.000 neue Schadprogramme pro Tag registriert [1]. Diese Zahl ist jedoch sogar noch nach oben zu korrigieren, da nicht immer eine korrekte Einordnung der Software möglich ist. So können zum Beispiel, Potentially Unwanted Applications (kurz: PUA), nicht immer eindeutig als Schadprogramm definiert werden [1]. Um diesem Trend der letzten Jahre zu begegnen, müssen auch die Sicherheitsexperten ihre Abwehr- und Analysemaßnahmen stetig verbessern und an die Vielzahl von Bedrohungen anpassen. Genau zu diesem Zweck können Intrusion Detection Systeme eingesetzt werden. Sie bieten, sobald diese korrekt installiert und konfiguriert wurden, ein mächtiges Werkzeug zum Aufspüren ungewollter Aktivitäten und Zugriffe.

1.2 Motivation

Die Motivation dieser wissenschaftlichen Arbeit liegt darin, ein umfassendes Grundverständnis für Intrusion Detection Systeme, deren Arbeitsweisen und Schutzpotential zu erarbeiten. Dadurch sollen die Fragen beantwortet werden, welche verschiedenen Arten es auf dem Markt gibt, wie diese genau Arbeiten, für welche Zwecke die einzelnen Systeme eingesetzt werden können und wo deren Grenzen liegen. Diese Wissensbasis bildet die Grundlage, um der Hochschule Reutlingen ein weiteres Werkzeug an die Hand zu geben, um ihre IT-Infrastruktur zu schützen. Bis jetzt werden dazu diverse Werkzeuge, wie z. B. Firewalls und Anti-Viren Software eingesetzt [Experteninterview]. Dadurch wird schon jetzt der Netzwerkverkehr von außen und ein Großteil der IT-Geräte im eigenen Netzwerk abgesichert. Nach dieser Arbeit sollte es dem Leser möglich sein, anhand der Anforderungsanalyse, den Entscheidungsprozess des ausgewählten Intrusion Detection Systems nachzuvollziehen.

1.3 Aufbau der Arbeit

Die weiteren Kapitel der wissenschaftlichen Arbeit beschäftigt sich im nächsten Schritt mit den Intrusion Detection Systemen und ihrer diversen Variationen. Dabei werden in den einzelnen Unterkapiteln noch die verschiedenen Platzierungsmöglichkeiten bzw. ihre Funktionsweise herausgearbeitet. Das dritte Kapitel ist den Erkennungs- und Analysearten gewidmet. Hier steht jedes Unterkapitel für eine eigene Vorgehensweise bei der Konfiguration und der Erkennung von Gefahren. Im Anschluss wird die komplette Thematik, im letzten Hauptkapitel, zusammengefasst und ein abschließendes Fazit gezogen.

2 Intrusion Detection Systeme

Da die Wirtschaft und Gesellschaft ihre täglichen Angelegenheiten immer mehr auf komplexe IT-Systeme verlagert, machen sie sich unweigerlich davon abhängig. Es wurde über die letzten Jahrzehnte immer deutlicher, dass IT-Sicherheit, nicht nur durch ein reaktives Konzept erreicht werden kann, sondern eine Präventive Herangehensweise benötigt wird. Dies bedeutet Gefahren zu erkennen, bevor daraus ein Risiko für die eigenen Systeme entsteht.

2.1 Grundlagenwissen

Aus diesem Grund und um auf Risiken und Sicherheitsverletzungen reagieren zu können, wurde seit Mitte der 1980er Jahre die

Forschung, im Bereich der Intrusion Detection verfolgt. Resultat daraus sind die heute üblichen Intrusion Detection Systeme (kurz: IDS), wie sie vielerorts täglich im Einsatz sind. Diese Systeme unterscheiden sich geringfügig in ihrem internen Aufbau, jedoch sind die, in Tabelle 1 selbst definierten Hauptkomponenten in nahezu jedem IDS zu finden.

Tabelle 1: Definition der Module eines IDS

Bez.	Beschreibung
Sensor	Kleinste Einheit, die z. B. die Aufgabe hat, die CPU-Auslastung zu überwachen
Agent	Ein Agent ist auf einem IT-System aufgespielt und koordiniert die einzelnen Sensoren, teilweise findet hier eine Vorverarbeitung der Werte statt
Analyse-Einheit	Dieses Modul ist für die Auswertung der gesammelten Daten zuständig, der genaue Sitz kann variieren. Teilweise im Agenten, eigenes Modul oder im Server integriert
Server	Dieses Modul sammelt und verwaltet alle Informationen zentral. Teilweise erfolgt hier die Auswertung aller Informationen und die Entscheidung, was daraus resultiert

Ein IDS verfolgt die Aufgaben, jedes ungewöhnliche Verhalten zu dokumentieren und gegebenenfalls Alarm auszulösen. Diese arbeiten im Normallfall komplett autonom und bedürfen nur weniger aktiver Eingriffe durch die Administratoren. Dabei steht die möglichst frühe Erkennung der Angriffe im Mittelpunkt, um den Schaden der Zwischenfälle zu minimieren. Gleichzeitig sammeln Sie im normalen Betrieb auffällige Verhaltensmuster, um auf neue mögliche Angriffsmethoden aufmerksam machen zu können [3]. Das Ziel bei der Auswertung der gesamten Daten ist es, eine hohe Erkennungsrate zu erreichen und dabei so wenige Fehlalarme wie möglich zu erhalten [4]. Die genaue Definition einer Intrusion bzw. eines Zwischenfalls, für einen Host oder das Netzwerk, verletzt die Sicherheitsrichtlinien und bringt das IT-System in einen unzulässigen Zustand [5]. In der folgenden Auflistung [3] sind die verschiedenen Systemzustände aufgeführt, die ein IDS erlangen kann, nachdem die Analyse-Einheit die Informationen verarbeitet hat:

- **Wahr-Positiv:** Eine Veränderung wurde korrekt als Risiko erkannt
- **Wahr-Negativ:** Eine Veränderung wurde korrekt als harmlos erkannt
- **Falsch-Positiv:** Eine Veränderung wurde fälschlicherweise als Risiko erkannt
- **Falsch-Negativ:** Eine Veränderung wurde fälschlicherweise als harmlos erkannt

Bei der Optimierung der Analyse-Einheit stellen zwei Klassifikationsergebnisse eine Herausforderung dar. 1. Falsch-Positiv, also ein sicherer Zustand, der als Risiko erkannt wurde und 2. Falsch-Negativ, also ein Risiko, welches nicht als solches erkannt wurde [3]. Die Voraussetzung für eine erfolgreiche automatische Erkennung liegt auf der Qualität der Daten. Eine große Anzahl an Informationen hilft nur, wenn diese auch rechtzeitig ausgewertet werden können. Deshalb ist es durchaus sinnvoll, nur einen speziellen Teil eines IT-Systems periodisch überwachen und nicht den vollen Umfang an Daten händeln zu müssen. Was und wie beobachtet wird, müssen in den meisten Fällen die Administratoren in den Konfigurationsdateien einstellen [3]. Ein IDS enthält dabei immer die folgenden Komponenten: Einen oder mehrere Sensoren, welche die Ereignisse an ihren Agenten weiterleiten. Eine Analyse-Einheit und ein Modul zur Berichterstattung z. B. durch einen Server. Dies können Konsolenausgaben sein, E-Mails o-

der SMS die verschickt werden oder der direkte Report an ein Intrusion Prevention System (kurz: ISP) [5].

Je nach Einsatzort und Installationsart unterscheidet man die folgenden Möglichkeiten, wie sin in der folgenden Auflistung [5] zu sehen sind.

- **Lokale Installation:** Wird genutzt, um einen einzelnes IT-System zu überwachen. Sensoren, Agent und Server befinden sich auf demselben IT-System
- **Agenten Installation:** Wird genutzt, um mehrere IT-Systeme zu überwachen. Sensoren und Agenten arbeiten, mit Analyse Einheit auf dem IT-System und geben Alarmmeldungen an eine zentrale Installation weiter
- **Server Installation:** Erweiterung der Agenten Installation. Daten werden, auch ohne Alarmmeldung weitergeleitet. Die Analyse-Einheit sitzt auf einem extra Server. So können die Informationen kumuliert verarbeitet werden

In den folgenden Unterkapiteln werden die einzelnen Einsatzmöglichkeiten der Intrusion Detection Systemen genauer erläutert.

2.2 *Host-Basiert*

Bei einem Host-basierten IDS (kurz: HISD) handelt es sich, z. B. um eine Software, die sowohl Agent als auch Sensor enthält. Sie wird auf dem zu überwachenden IT-System, wie einem Server oder einer Workstation, installiert und hat direkten Zugriff auf die Ressourcen. So kann z. B. der komplette Kommunikationsstrom überprüft werden, bevor er dem Betriebssystem präsentiert wird. Dadurch kann auch verschlüsselte Kommunikation aus dem Netzwerk noch überprüft werden, welcher einem Netzwerk-basierten Intrusion Detection System entgehen könnte. Ebenfalls können ausgehende Pakete, für das Netzwerk, vor der eigenen Verschlüsselung überprüft werden. Da der Sensor und Agent direkt auf dem Host installiert sind, hat er Zugriff auf wichtige Funktionen zur Überwachung. So verfügt er z. B. über System-Level Checks, kann die Dateiintegrität überprüfen, überwacht Veränderungen in der Registry, analysiert lokale Protokolle und bietet eine Rootkit Erkennung. Darüber hinaus verfügen einige HIDS über eine aktive Response, um größeren Schaden direkt zu verhindern [6]. Ein Beispiel, mit einer Agenten Installation ist in Abbildung 2 zu sehen.

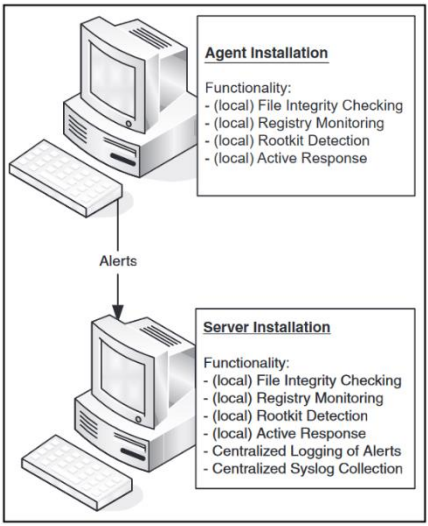

Abbildung 2: Beispiel für eine Host-basierte Agenten Installation [6]

Dabei wird ein Agent auf dem zu überwachenden IT-System installiert und die gewünschte Funktionalität konfiguriert. Werden unzulässige Zustände, in den einzelnen Sensoren erkannt, wird eine Alarmmeldung an den Server gesendet. Diese Infrastruktur lässt sie weiter skalieren, bis der Server seine Maximalanzahl an zu überwachenden Clients erreicht hat und diese nicht mehr verarbeiten kann. Zur Veranschaulichung einer lokalen Installation, steht Abbildung 3 zur Verfügung. Hier befindet sich die komplette

Überwachung des IT-Systems auf einem einzelnen Host.

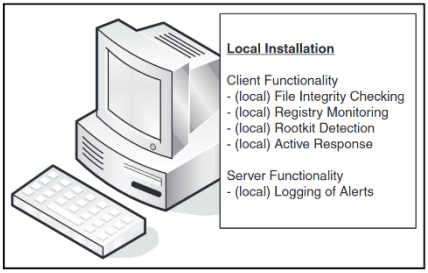

Abbildung 3: Beispiel für eine Host-basierte lokale Installation [6]

2.2.1 Dateiintegritätscheck

Durch eine Hashfunktion bildet jede Datei auf einem IT-System einen eindeutigen Fingerabdruck. Dieser ist abhängig vom Inhalt, der Dateigröße und dem jeweiligen Dateinamen und verändert sich jeder Art von Manipulation. Werden diese Fingerabdrücke periodisch überprüft, fällt eine Veränderung sofort auf und kann näher kontrolliert werden [6]. Grundlage dieser Maßnahme ist wiederum, dass das System, zum Zeitpunkt der Inbetriebnahme des IDS, sich in einem zulässigen Zustand befand.

2.2.2 Registry Überwachung

Bei der Systemregistry handelt es um ein zentrales Verzeichnis auf einem Windows Betriebssystem. Hier werden alle Hard- und Software-Einstellungen, Betriebssystemkonfigurationen, wie auch Benutzer und Gruppenrichtlinien konzentriert verwaltet. Änderungen daran, egal ob durch Benutzer oder Administratoren, werden durch Schlüssel gespeichert. Ein HIDS kann auf Änderungen bei diesen wichtigen Schlüsseln achten, indem es periodisch prüft. Dadurch werden böswillige Absichten, durch Nutzer oder Programme, die z. B. ein neues Programm installieren wollen, sofort erkannt [6]. Vorausgesetzte Annahme: Das System befand sich zur Zeit der Installation in einem zulässigen Zustand.

2.2.3 Rootkit Erkennung

Rootkits existieren schon seit den 1990er Jahren. Sie sind als eine Hintertür zu betrachten, durch die Angreifer sich als Administrator (engl. root) bewegen können [7]. Gut im System verborgen interagiert es mit dem System. Dabei kann das Rootkit Ports, Dateien, Verzeichnisse, Dienste und Registryschlüssel vor dem Nutzer verstecken. Die am häufigsten vorkommende Art des Rootkits, die auf Anwendungsebene, ersetzt originale Anwendungs-Binärdateien durch eigene, manipulierte Versionen. An dieser Stelle funktioniert die Detektionsart ähnlich dem Dateiintegritätscheck. Bei Rootkits auf Kernelebene oder virtualisierten Rootkits, zwischen Hardware und Betriebssystem, die Systemcalls manipulieren wird die Detektion schon schwieriger [6].

2.2.4 Aktive Response

Als direkte Antwort auf Bedrohungen, können HIDS gegebenenfalls in den laufenden Betrieb des IT-Systems eingreifen, indem sie vordefinierte Kommandos ausführen. So können z. B. Ports gesperrt, die Ausführung eines fragwürdigen Programms geblockt oder ein Alarm rausgegeben werden. Von dieser Funktionalität profitieren auch IPS, die direkt angesprochen werden und die umfangreichere Gegenmaßnahmen einleiten können. So groß die Vorteile einer dieser Aktiven Response Einheiten sind, bieten sie ebenfalls großes Risikopotential. Falsch-Positive Systemzustände können z. B. den Betrieb stören oder zum Erliegen bringen. Oftmals ausgelöst durch schlecht definierte Regeln oder ungenaue Beobachtung von Anomalien [6].

2.3 Netzwerk-Basiert

Die Netzwerk-basierten Intrusion Detection Systeme (kurz: NIDS) arbeiten als eigene Netzwerkteilnehmer oder erweitern, z. B. eine Firewall (kurz: FW). Ihre Aufgabe besteht darin, den Datenverkehr im Netzwerksegment zu analysieren und bei auffälligem Verhalten Alarm zu schlagen. Zur Abgrenzung der zwei Systeme: Eine Firewall schützt vor Angriffen von und nach außen, eine NIDS schützt und analysiert den Datenverkehr im eigenen Netzwerk. Eine Kombination ist also durchaus empfehlenswert und in Abbildung 4 zu sehen.

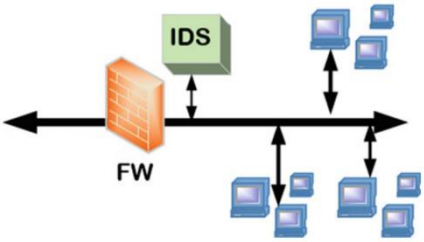

Abbildung 5: Kombination aus NIDS und Firewall in einem Netzwerk [4]

Diese Konfiguration inspiziert alle Pakete, die sich über das Netzwerk bewegen [4]. Um bei einem modernen Netzwerk an alle Pakete zu gelangen, muss der Switch über eine Mirrorport-Funktion verfügen. Damit wird der gesamte Verkehr des Switches ebenfalls an das NIDS weitergeleitet. In früheren Aufbauten mit Netzwerkhubs gab es diese Voraussetzung noch nicht. Ein großer Nachteil des NIDS ist die benötigte Bandbreite und die Rechenintensive Auswertung der Pakte [4].

2.4 Hybride Systeme

Hybride IDS haben den Vorteil, dass sich die HIDS und das NIDS ergänzen. Im besten Fall liefern alle Agenten ihre gewonnenen Informationen an einem zentralen Server ab. So können diese Erkenntnisse kumuliert verarbeitet werden und exaktere Aussagen, zur aktuellen Gefährdungslage, liefern. Ein Beispielaufbau ist in Abbildung 5 zu sehen. Hier

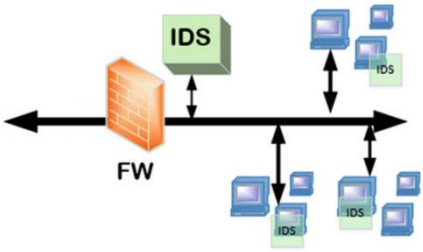

Abbildung 4: Beispielaufbau eines hybriden IDS [4]

wird das gesamte Netzwerk von außen durch eine Firewall geschützt, der interne Netzwerkverkehr durch in NIDS überwacht und wichtige IT-Systeme verfügen zusätzlich noch über ein HIDS. Im optimalen Fall werden alle Erkenntnisse an einem zentralen Punkt analysiert, um die genauesten Vorhersagen treffen zu können.

3 Detektionsmethoden

Grundlegend arbeiten alle IDS, egal ob HIDS oder NIDS, mit drei verschiedenen Detektionsmethoden. Jede davon benötigt Daten zur Analyse und geht mit ihren Stärken und Schwächen einher. Eine Kombination verschiedener Herangehensweisen ist durchaus empfehlenswert bzw. üblich.

3.1 Regel-Basiert

Die wohl einfachste Form der Filterung ist die Regel-basierte Herangehensweise. Sie wird durch einen Satz, fest vorkonfigurierter Regeln und Einschränkungen gespeist und blockt oder ermöglicht genau diese Art von Ereignis. Diese Detektionsmethode ist nur so gut, wie ihre Konfiguration und kann darüber hinaus nichts Weiteres erkennen [4].

3.2 Signatur-Basiert

Signatur-basiert IDS arbeiten mit bekannten Signaturen oder Angriffsmustern, um böswilliges Verhalten zu erkennen. Sie eignen sich, um mit geringem Aufwand, schon bekannte Angriffe zu erkennen. Kommen sogenannte Zero-Day-Exploits oder neuere Methoden zum Einsatz, sind die Signatur-basierten IDS nicht effektiv beim Erkennen ei-

ner Bedrohung. Wichtig ist das die Signaturen regelmäßig aktualisiert werden, damit die Schutzfunktion erhalten bleibt. Dazu gibt es Open Source Ansätze, sowie auch kommerzielle Anbieter [4]. Generell liefert diese Methode robuste und scharfe Ergebnisse, auf deren Basis agiert werden kann [3].

3.3 Heuristische Methoden

Heuristische Methoden, auch Anomalie-basierte Erkennung genannt, ist das Mittel der Wahl, wenn es um die Detektion von bisher unbekannten Angriffsmustern geht. Diese Detektionsmethode definiert einen expliziten Wert für normales Verhalten der IT-Systeme. Je nach eingestelltem Schwellenwert, wird bei einer Anomalie, also einer Abweichung vom Normverhalten, eine Alarmmeldung ausgelöst. Probleme sind, dass nicht jedes unnatürliche Verhalten ein Alarmzustand sein muss. Ein weiterer Nachteil sind die Erkennungsraten solcher Verfahren bei bekannten Angriffen. Diese sind den Signatur-basierten Detektionsmethoden unterlegen und produziert keine so zuverlässigen Ergebnisse [3] [4].

4 Intrusion Detection System für die Hochschule Reutlingen

Zielsetzung dieser wissenschaftlichen Arbeit ist die fundierte Auswahl eines passenden IDS, nach den Ansprüchen des Rechenzentrums (kurz: RZ) der Hochschule Reutlingen (kurz: HSRT).

Im ersten Schritt wurde eine Anforderungsanalyse durchgeführt. Dazu stand ein Mitarbeiter des RZ für ein Experteninterview zur Verfügung. Während des Gesprächs wurde auf die bestehende und zukünftige IT-Infrastruktur eingegangen, die bisherigen Angriffe und Zwischenfälle besprochen und ebenfalls Wünsche und Anregungen für das neue System geäußert. Auf Basis dieses Gesprächs wurden Anforderungen abgeleitet und diese gewichtet. Die wichtigsten Erkenntnisse daraus sind in Tabelle 2 aufgeführt. Bei diesen handelt es sich um die primären Anforderungen, welche erfüllt sein müssen, um eine Softwarelösung zu qualifizieren.

Tabelle 2: Primäre Anforderungen des RZ an ein IDS

Bez.	Beschreibung
A1	Das System dar keine zusätzlichen (Lizenz-) Kosten mit sich bringen
A2	Der Fokus des gewünschten Systems liegt auf den einzelnen Hostsystemen, z. B. in den Laboren
A3	Die Agenten bzw. Sensoren dürfen kein Betriebssystem (Windows, Linux, MacOS) ausschließen
A4	Das System soll die Nutzung der Hosts bzw. den Anwender nicht merklich einschränken
A5	Das System muss, durch eigens geschriebene Module erweiterbar sein, um auf jeden denkbaren Fall reagieren zu können

Ausgehende der primären Anforderung A2, wurde ein HIDS gesucht, während die NIDS beim Auswahlprozess keine Beachtung fanden. Die Anforderungen A1 und A5 führten die Recherche zu diversen Open Source Projekten, mit breiter Community-Unterstützung. Die Anforderung A3 konnte durch Recherchen geklärt werden und bei A4 wurden die Systemanforderungen, der Sensoren und Agenten, betrachtet. Die Auswertung, durch eine Tabelle mit Gewichtung der Eigenschaften der recherchierten HIDS, brachte einen vielversprechenden Kandidaten hervor - die aktuelle Version von Wazuh [8]. Dabei handelt es sich um eine aktiv gepflegte Fork, des bekannten HIDS OSSEC [12]. „Wazuh bietet eine Sicherheitslösung, die in der Lage ist, Ihre Infrastruktur zu überwachen, Bedrohungen, Eindringversuche, Systemanomalien, schlecht konfigurierte Anwendungen und nicht autorisierte Benutzeraktionen zu erkennen. Sie bietet auch einen Rahmen für

die Reaktion auf Vorfälle und die Einhaltung gesetzlicher Vorschriften.“ [8]. Die verteilte Installation, auch als Server Installation aus den vorhergegangenen Kapiteln bekannt, verfügt über drei Hauptkomponenten. Diese sind in Abbildung 6 zu sehen. Auf die drei Säulen wird im Folgenden genauer eingegangen.

Abbildung 6: Server Installation von Wazuh [10]

Der Wazuh Agent, in Abbildung 6 die rechte Säule, stellt die Sensoren für verschiedene IT-Systeme und Aufgaben bereit. Darunter auch Dateiintegritätschecks, Sammlung von Protokoll- und Registry Daten, Listung von laufenden Prozessen und installierten Programmen, Überwachung von offenen Ports und der Netzwerkkonfiguration und weitere Funktionen. Die Verwaltung und Konfiguration läuft dabei über den Wazuh Server [9].

Der Elastic Stack, in der mittleren Säule in Abbildung 6, bietet die Möglichkeit, die gesammelten Daten anschaulich zu Visualisieren. Dabei gibt es für die Weboberfläche vorgefertigte Dashboards, die nach Belieben abgeändert werden können. Dazu wurde Kibana [11] in den Elastic Stack integriert [9].

Die letzte Säule, ganz links in Abbildung 6, stellt den Wazuh Server dar. Er hat die Aufgabe, die von den Agenten erhaltenen Daten zu analysieren und die Ergebnisse durch Regeln und weitere Methoden auszuwerten. Im täglichen Betrieb kann ein einzelner Wazuh Server hunderte von Agenten bedienen und ist skalierbar, wenn dieser im Cluster Modus installiert wurde. Ein weiterer großer Vorteil des ausgewählten HIDS ist, dass die Konfiguration und Aktualisierung der Agenten per Fernzugriff möglich ist. Ebenfalls kann der Server den Agenten Kommandos schicken, um sich ggf. vor einem erkannten Angriff zu schützen [9].

5 Fazit

Das Resultat dieser Arbeit ist die Erkenntnis, dass es einige sehr potente Ansätze im Open Source Segment der IDS gibt. Unterschiede sind oftmals der Funktionsumfang, die Leistungsfähigkeit und der Einsatzzweck der Anwendungen. Dabei bauen einige Systeme aufeinander auf oder nutzen gleiche Codebausteine. In Bezug auf diese Arbeit, ist es als Erfolg zu werten, dass alle primären Anforderungen, an das IDS, erfüllt und eine entsprechende Softwarelösung gefunden werden konnte. Darüber hinaus bietet Wazuh noch zusätzliche Funktionalität, wie z. B. die Konfiguration der Agenten über das Netzwerk, die nicht Teil der primären Anforderungen waren.

Literaturverzeichnis

[1] Bundesamt für Sicherheit in der Informationstechnik (BSI): Die Lage der IT-Sicherheit in Deutschland 2019; Bundesamt für Sicherheit in der Informationstechnik (BSI); 2019; Bonn; Art.Nr.: BSI-LB19/508; https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Lageberichte/Lagebericht2019.pdf;jsessionid=8CCE2790C8E9DBB3DEDFED40D66394F1.2_cid502?__blob=publicationFile&v=7; Besucht am 17.06.2020

[2] AV Test GmbH: Total Malware; 2020; https://www.av-test.org/typo3temp/avtestreports/print_total_ distribution_10-years_en.png?1597595107; Besucht am 17.08.2020

[3] M. Meier: Intrusion Detection effektiv! Modellierung und Analyse von Angriffsmustern; Springer; Berlin Heidelberg; 2007; ISBN 978-3-540-48251-2

[4] K. Kim, M. Aminanto: Network Intrusion Detection using Deep Learning –

A Feature Learning Approach; Springer; Singapore; 2018; DOI https://doi.org/10.1007/978-981-13-1444-5

[5] P. Oorschot: Computer Security and the Internet – Tools and Jewles; Springer; Cham; 2020; DOI https://doi.org/10.1007/978-3-030-33649-3

[6] A. Hay, D. Cid: OSSEC Host-Based Intrusion Detection Guide; Elsevier; 2008; ISBN 978-1-59749-240-9

[7] P. Kraft, A. Weyert: Network Hacking – Professionelle Angriffs- und Verteidigungstechniken gegen Hacker und Datendiebe; Franzis; 2010; ISBN 978-3-645-60030-9

[8] Wazuh – Open Source Network Intrusion Detection System Webiste; https://wazuh.com/; Besucht am 02.10.2020

[9] Wazuh -Produktwebsite; https://wazuh.com/product/; Besucht am 04.10.2020

[10] Wazuh – Docs Installation Guide; https://documentation.wazuh.com/3.13/installation-guide/index.html; Besucht am 03.10.2020

[11] Kibana Website; https://www.elastic.co/kibana; Besucht am 05.10.2020

[12] OSSEC Website; https://www.ossec.net/; Besucht am 10.10.2020

Intrusion Detection System mit maschinell lernenden Algorithmen

Kevin Jucknischke
Hochschule Reutlingen
Kevin.Jucknischke@Student.Reutlingen-University.de

Abstract
Maschinelles Lernen hat in den letzten Jahren zunehmend an Relevanz gewonnen, so auch in der IT-Sicherheit. Mithilfe von Algorithmen werden Intrusion Detecion Systeme trainiert, um auf neue Angriffsvektoren reagieren zu können. In dieser Arbeit werden die Grundlagen des maschinellen Lernens erläutert sowie die Ergebnisse von zwei Forschungen aufgezeigt, um der Forschungsfrage nachzugehen, welche Algorithmen sich eignen, um ein maschinell lernendes Intrusion Detection System zu trainieren. Außerdem wird die Software-Bibliothek Scikit-Learn und die Software Weka vorgestellt mit der die Implementierungen stattfinden.

Keywords
Intrusion Detection System; Algorithmen; maschinelles Lernen; KDD; NSL-KDD;

1 Einleitung
Die IT-Sicherheit ist ein immer weiter wachsendes Thema der Informatik. Durch die Digitalisierung vieler Prozesse und steigenden Zahlen der Internet of Things wie Smartphones, Tablets und Wearables, wird die Sicherheit immer relevanter. Neben der wachsenden Zahl der mit dem Internet verbundenen Geräte, steigt ebenfalls die Notwendigkeit Sicherheitsmechanismen zu etablieren, die diesem Anstieg gewachsen sind. Da dies allerdings nicht mehr händisch zu bewältigen ist, ist es nötig, dies automatisiert zu überwachen. Für IT-Infrastrukturen bieten sich Systeme an, die den Netzwerkverkehr automatisiert kontrollieren. Vor allem ist es wichtig, dass diese Systeme sich an ihre Umgebung anpassen und von ihr lernen können.

Diese Arbeit befasst sich mit dieser Anforderung und geht der Frage nach, wie maschinelles Lernen eingesetzt werden kann und welcher Nutzen sich daraus ziehen lässt. Im Folgenden werden die Forschungsfragen aufgeführt, die diese Arbeit behandelt und beantworten will. F1: Welche Algorithmen eignen sich für ein Intrusion Detection System? F2: Wo können die Algorithmen gefunden werden und wie werden sie angewendet?

Um diese Fragen beantworten zu können wird zunächst in den Grundlagen das benötigte Wissen vermittelt. Dazu werden zunächst einige Schutzziele und deren Bedeutung erläutert. Die Schutzziele stellen Anforderungen an die IT-Sicherheit dar, welche gewahrt werden müssen. Weiterhin wird erklärt was Angriffsvektoren sind, da diese genutzt werden, um eben jene Schutzziele zu verletzen. Daraufhin folgt ein kurzer Einblick in die Erkennungssoftware, die solche Angriffsvektoren automatisiert erkennen und melden kann. Weiterhin werden Datensätze aufgezeigt, die genutzt werden, um der Erkennungssoftware gewisse Verhaltensmuster anzueignen. Um eine Aussage zu treffen, ob die angelernten Verhaltensmuster die gewünschten Ergebnisse liefern, ist

Betreuer Hochschule: Prof. Dr. Michael Tangemann
Hochschule Reutlingen
Michael.Tangemann@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

es notwendig sie anhand definierter Metriken zu messen und vergleichen zu können.

In Kapitel 3 wird der Begriff maschinelles Lernen erläutert, gefolgt von der Implementierung der Algorithmen. Im Anschluss folgen zwei Forschungen und deren erreichten Ergebnisse, die ein solches System umgesetzt haben. Zuletzt finden eine Diskussion und ein Fazit über die gefundenen Erkenntnisse sowie die Beantwortung der Forschungsfragen statt.

2 Grundlagen

In einer früheren Arbeit wurden bereits die Schutzziele und Angriffsvektoren im Hinblick auf ZigBee untersucht [6]. Das dort gesammelte Wissen wird aufgearbeitet und im Bezug zu Intrusion Detection Systeme wiederverwendet.

2.1 Schutzziele

Im Bereich der IT-Sicherheit gibt es Schutzziele, die es zu wahren gilt. Mithilfe dieser Schutzziele sollen vertrauliche Informationen vor einem unautorisierten Zugriff oder Diebstahl geschützt werden. Die drei wichtigsten sind die CIA Schutzziele (engl. CIA Triad) [3]. Im Folgenden werden sie aufgeführt und ihre Bedeutung erläutert:

Confidentiality (Vertraulichkeit): Für eine Informationsvertraulichkeit ist es Voraussetzung, dass das System keine unautorisierte Informationsgewinnung zulässt. Dies bedeutet konkret, dass auf die Informationen nur mit einer Befugnis zugegriffen werden darf. Diese Befugnis kann beispielsweise durch eine Verschlüsselung eingerichtet werden. Durch diese Verschlüsselung ist es zwar dennoch theoretisch möglich, dass eine dritte Person an die Daten gelangen, diese aber ohne den ausgehandelten Schlüssel nicht lesen kann. Die dritte Person kann daher keine Informationen aus den Daten „gewinnen“ [2].

Integrity (Integrität): Bei der Integrität ist es das Ziel, die Korrektheit der Daten bzw. die korrekte Funktionsweise des Systems zu wahren. Dies bedeutet, dass es nicht möglich sein darf die Daten unautorisiert oder unbemerkt zu manipulieren, wobei hier die Integrität in unterschiedliche Kategorien gegliedert wird. Dies ist davon abhängig, inwiefern die Zugriffsberechtigung auf die Daten geregelt ist. Hat der Nutzer beispielsweise eine Schreibberechtigung, kann die Integrität nur beschränkt gewährleistet werden. Von einer Integrität „im Sinne einer authentischen, korrekten Funktionalität [2]“ wird gesprochen, wenn die Daten nur durch wohl definierte Methoden verwendet werden können. Dadurch werden die Manipulationsmöglichkeiten deutlich eingeschränkt [2]. Bei der Verwendung verschiedener Methoden kann eine Manipulation der Daten festgestellt werden. Realisiert wird dies u.a. durch den Einsatz einer Prüfsumme. Der Sender errechnet aus den Ausgangsdaten eine Prüfsumme, die zusammen mit den Daten übertragen wird. Der Empfänger errechnet durch dasselbe Verfahren und den empfangenen Daten ebenfalls eine Prüfsumme und vergleicht diese mit der Empfangenen. Variieren diese Summen, ist davon auszugehen, dass die Integrität der Daten nicht mehr gegeben ist.

Availability (Verfügbarkeit): Dieses Ziel wird erreicht, wenn die Funktionalität und Erreichbarkeit des Systems gewährleistet ist. Hier muss allerdings unterschieden werden zwischen einer größeren Wartezeit und einem Systemausfall. Im normalen Betrieb kann es passieren, dass ein angeforderter Dienst bzw. die Antwort eines Systems verzögert ankommt. Dies ist von verschiedenen Faktoren abhängig und stellt noch keine Verletzung der Verfügbarkeit dar. Verletzt wird dieses Ziel meist durch Denial of Service Angriffe, mit denen versucht wird die Rechenkapazität eines Systems zu überlasten, um das System zum Ausfall zu bringen [2].

Authentizität: Ein weiteres wichtiges Schutzziel ist die Authentizität. Dieses Ziel dient dazu, die Identität eines Nutzers zu authentifizieren, um sicherstellen zu können, dass die Kommunikation tatsächlich zwischen den gewünschten Partnern stattfindet und nicht über einen unbekannten Dritten.

Sichergestellt werden kann dies durch den Einsatz einer eindeutigen Benutzererkennung (z.B. Benutzerkonto). Der Nachweis der Identität eines Benutzers erfolgt meist durch den Einsatz eines Passwortes. Durch die Eingabe des Passwortes weist der Nutzer dem System gegenüber seine Identität nach [2].

2.2 Angriffsvektor

Ein Angriffsvektor beschreibt eine Methode bzw. einen Weg, den der Hacker geht, um in das ausgewählte System unautorisiert einzudringen. Dabei kann der Hacker das gesamte System übernehmen, Schadsoftware installieren oder Ressourcen des angegriffenen Systems für eigene Zwecke missbrauchen [1]. Durch den Angriffsvektor ist es dem Hacker möglich, Schwachstellen des Systems auszunutzen und es anzugreifen. Meist ist auch eine Kombination mehrerer Angriffsvektoren möglich, um ein System anzugreifen. Angriffe können in zwei Kategorien unterschieden werden. Passive Angriffe dienen dazu, Daten während der Übertragung abzufangen und mitzulesen. Mit Hilfe eines aktiven Angriffs kann der Angreifer Daten abändern, versenden oder sie komplett entfernen [2].

In Abbildung 1 werden einige Möglichkeiten und Wege aufgezeigt, wie ein Angriffsvektor aussehen kann.

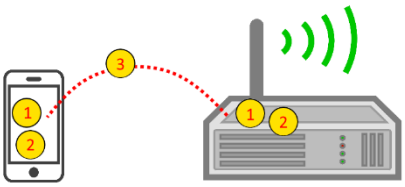

Abbildung 1: Ansicht möglicher Angriffsvektoren

1. Angriff auf das physische Gerät: Hierbei zählt zum Einen der direkte Angriff auf das Gerät, um die Funktionen zu sabotieren [4]. Dabei geht es darum, die Ressourcen des Gerätes auszulasten, sodass es zu einer Fehlfunktion im Betrieb kommen kann (Angriff auf die Verfügbarkeit). Weiterhin ist hier ein physikalischer Angriff auf die Geräte möglich, mit der Absicht diese zu zerstören. Dazu ist es notwendig, dass der Angreifer in direkten Kontakt mit dem Gerät treten kann.

2. Software des Gerätes angreifen: Dabei wird versucht eine Schwachstelle der Software auszunutzen, um beispielsweise root-Rechte des Gerätes zu erhalten [4]. Ebenfalls wird hierbei versucht, Informationen über das Funkmodul in Erfahrung zu bringen, die für weitere Angriffe relevant sein können (Angriff auf die Vertraulichkeit). Es werden beispielsweise die Firmwareversion oder weitere vertrauliche Informationen ausgelesen, die das Gerät zur Kommunikation nutzt. Weiterhin können manipulierte Daten an das Gerät gesendet werden, um eine Zustandsänderung an diesem auszulösen.

3. Kommunikationskanal angreifen: Da die Kommunikation über Funk stattfindet, bieten sich hier viele Möglichkeiten an, die Übertragung anzugreifen. Die Übertragung kann zum einen abgehört werden, um an nützliche Informationen zu gelangen. Zum anderen kann versucht werden das Übertragungsmedium mit Störfrequenzen zu überlagern, sodass die Kommunikation zwischen den Geräten beeinträchtigt wird. Zuletzt kann versucht werden die Kommunikation abzufangen, diese zu verfälschen und anschließend an den eigentlichen Empfänger weiterzuleiten (Angriff auf die Integrität und Authentizität) [4].

2.3 Intrusion Detection System

Bei der Intrusion Detection werden innerhalb eines Netzwerks oder eines Computersystems auftretende Ereignisse überwacht und analysiert. Das Ziel dabei ist es, mögliche Bedrohungen oder eventuelle Verletzungen der Schutzziele ausfindig zu machen und diese Vorfälle zu melden. In der Regel gehen diese Verletzungen von Angreifern aus, die sich unautorisiert, mithilfe von unterschiedlichen Angriffsvektoren, einen Zugang zum jeweiligen System verschaffen wollen. Ebenfalls können diese Vorfälle aber auch von Personen ausgelöst werden, die keine

bösen Absichten verfolgen. Beispielsweise kann durch die falsche Eingabe einer Adresse eines Computers ein solcher Vorfall ausgelöst werden [5].

Mithilfe eines Intrusion Detection Systems (IDS) wird der Prozess zur Erkennung solcher Vorfälle automatisiert. Ein IDS ist eine Software, die innerhalb einer IT-Infrastruktur eingesetzt wird, um selbstständig potenzielle Angriffe auf Computersysteme oder Netzwerke zu erkennen und den Anwender bzw. Administrator zu informieren. Dabei unterstützt ein IDS die üblichen Funktionen einer Firewall und ersetzt diese nicht. Ein Intrusion Prevention System (IPS) besitzt sämtliche Funktionen eines IDS und erweitert dieses um die Funktion, eigenständig Maßnahmen gegen anliegende Vorfälle zu unternehmen. In der Literatur wird oftmals von Intrusion Detection und Prevention Systemen (IDPS) gesprochen, da sich bei IPS-Produkten die Präventionsfunktionen deaktivieren lassen, sodass sie als IDS funktionieren [5]. Das IDS lässt sich zunächst in die zwei Kategorien Host-Based und Network-Based unterteilen.

Ein **Host-Based IDS** ist auf einem System direkt implementiert und überwacht dessen Ereignisse auf verdächtige Aktivitäten. Darunter zählen Systemprotokolle, laufende Prozesse, Dateizugriffe und der Netzwerkverkehr des Systems. Solche Erkennungssysteme werden meist auf systemkritischen Hosts installiert, wie z.B. öffentlich zugänglichen Server [5]. Allerdings kann das System beispielsweise durch einen Distributed-Denial-of-Service (DDoS)-Angriff außer Gefecht gesetzt werden, wodurch der Schutz des IDS unwirksam wird.

Ein **Network-Based IDS** wird innerhalb eines Bereichs im Netzwerk implementiert und überwacht dessen Netzwerkverkehr. Dabei werden alle Datenpakete mitgelesen und auf verdächtige Muster überprüft. Jedoch muss hier eine entsprechende Bandbreite und hohe Performance des IDS sichergestellt werden, da sonst eine vollständige Überwachung nicht gewährleistet werden kann. Das Network-Based IDS lässt sich weiter unterteilen in Bereiche, die speziell den drahtlosen Netzwerkverkehr überwachen oder gezielt Bedrohungen identifizieren soll. Beispielsweise kann hier nach DDoS-Angriffen oder bestimmte Formen von Malware gesucht werden [5].

Ein s.g. **hybrides IDS** vereint die beiden Technologien und nutzt sowohl Host-basierte als auch Netzwerk-basierte Sensoren und führt diese in einem Managementsystem zusammen.

2.4 Trainingsdatensatz

Der KDD Cup 1999 Datensatz [14] basiert auf den gesammelten Daten des Darpa´89 Datensatzes und wird seit 1999 für die Evaluierung von Methoden zur Erkennung von Anomalien genutzt [10]. Der KDD Datensatz beinhaltet etwa 4,9 Million Einträge und besteht aus Datenpaketen, die innerhalb eines Netzwerks versendet werden. Weiterhin werden die Einträge als *normal* oder *Angriff* bezeichnet und einem Angriffsvektor zugeordnet. Diese Angriffe lassen sich in folgende vier Kategorien unterteilen [12]:

1. User to Root (U2R): Bei diesem Angriff versucht der Angreifer die administrativen Rechte des Systems zu erhalten oder Zugriff auf Dateien zu erlangen, die eine entsprechende Rolle benötigen.

2. Root to Local Attack (R2L): Hierbei versucht der Angreifer per remote-Verbindung auf ein Computersystem innerhalb eines Netzwerks zuzugreifen und Datenpakete an das System zu senden, um lokale Zugriffsberechtigungen zu erhalten.

3. Denial of Service (DoS): Mithilfe dieses Angriffes nutzt der Angreifer die eigenen Ressourcen von Systemkomponenten, um das Ziel zu überlasten, so dass es für eine gewisse Zeit nicht mehr erreichbar ist. Hierzu wird das Zielsystem mit Anfragen überschwemmt bis es unter der Last die Anfragen nicht mehr verarbeiten kann und abstürzt.

4. Probing: Bei diesem Angriff wird versucht Informationen über das Zielsystem zu erhalten, so dass dessen Sicherheitskontrollen umgangen werden können.

Tabelle 1 zeigt die Aufteilung aller Angriffsvektoren des KDD Datensatzes und in welche Kategorien sich dieser Angriff einsortieren lässt. Insgesamt sind dort 21 Arten von Angriffen aufgeführt. Hierbei lässt sich erkennen, dass ein Großteil der Angriffe aus DoS-Angriffen besteht und nur ein kleiner Teil die anderen Vektoren bedient.

Tabelle 1: Aufteilung der Angriffsvektoren des KDD Datensatzes. [11]

Categories of Attack	Attack name	Number of instances
DOS	SMURF	2807886
	NEPTUNE	1072017
	Back	2203
	POD	264
	Teardrop	979
U2R	Buffer overflow	30
	Load Module	9
	PERL	3
	Rootkit	10
R2L	FTP Write	8
	Guess Passwd	53
	IMAP	12
	MulitHop	7
	PHF	4
	SPY	2
	Warez client	1020
	Warez Master	20
PROBE	IPSWEEP	12481
	NMAP	2316
	PORTSWEEP	10413
	SATAN	15892
normal		972781

Trotz dessen, dass der Datensatz bereits über 20 Jahre alt ist, geht man davon aus, dass sich solche Datensätze noch immer dazu eignen, um Systeme zu trainieren, da neuartige Angriffe meist nur Varianten von bereits bekannten Angriffen sind [10]. Dennoch wurde dieser Datensatz im Laufe der Zeit überarbeitet, da sich im ursprünglichen Datensatz beispielsweise viele redundante Einträge befinden. Weiterhin steht dieser Datensatz unter der Kritik, da beispielsweise der Angriff *Probing* nur unter gewissen Umständen als Angriff gewertet werden kann. So muss bei diesem Angriff z.B. ein gewisser Schwellenwert der Iterationen überschritten werden, da sonst nicht eindeutig festzustellen ist, ob es sich tatsächlich um einen Angriff handelt. Da ein Großteil der Intrusion Detection Systeme allerdings nur mit binären Werten arbeitet, also *normal* oder *Anomalie*, ist dieser Kritikpunkt kein Ausschlusskriterium für diesen Datensatz [10].

Der NSL-KDD Datensatz [15] wurde mehrmals überarbeitet, wodurch ein Großteil der kritisierten Probleme behoben wurden. So wurden beispielsweise redundante Einträge gelöscht und der Datensatz so angepasst, dass er reale Netzwerke besser repräsentieren kann. Trotzdem ist dieser Datensatz nicht fehlerfrei. Allerdings wird er zur Forschung noch immer genutzt, da ein Mangel an öffentlich zugänglichen Datensätzen besteht. Weiterhin besteht er aus einer angemessenen Anzahl an Trainings- und Testdaten, wodurch Forschungen mit ihm möglich sind und nicht um zufällig ausgewählte Daten erweitert werden müssen. Weiterhin bietet er eine einheitliche Basis durch welche Forschungsarbeiten konsistent und vergleichbar sind.

2.5 Bewertungsmetriken

Um eine Aussage darüber treffen zu können wie gut ein Algorithmus funktioniert, ist es notwendig Metriken [16] zu nutzen, mit denen sich die erzielten Ergebnisse darstellen lassen.

Accuracy (Genauigkeit): Die Genauigkeit ist die einfachste Metrik und gibt die Anzahl an, wie viele richtige Klassifizierungen geteilt durch alle Klassifizierungen gemacht wurden.

$$Accuracy = \frac{\#korrekt\ Klassifiziert}{\#alle\ Klassifizierungen}$$

Berechnung der Genauigkeit [16]

Precision (Präzision): Die Präzision ergibt sich aus allen relevanten Treffern durch die Anzahl aller relevanten und nicht relevanten Treffern. Das Ergebnis besagt also, dass alles was gefunden wurde, auch relevant ist.

$$Precision = \frac{hit}{hit + false\ alarm}$$

Berechnung der Präzision [16]

Recall (Abdeckung): Die Abdeckung beschreibt, ob alle relevanten Treffer gefunden wurden. Sie ergibt sich aus der Anzahl der relevanten Treffer geteilt durch die Summe der relevanten Treffer und irrelevanten Treffer.

$$Recall = \frac{hit}{hit + miss}$$

Berechnung der Abdeckung [16]

F1-Score: Es ist erstrebenswert, sowohl eine hohe Abdeckung als auch eine hohe Präzision zu erreichen. Da dies in der Praxis allerdings nicht immer möglich ist und die Werte nicht gegeneinander abwägen kann, lässt sich der s.g. F1-Score ermitteln. Dieser fasst beide Werte zusammen und gibt ein harmonisches Mittel aus. Dieses ergibt sich aus der Multiplikation von Abdeckung und Präzision geteilt durch die Summe von Abdeckung und Präzision.

$$F1 = 2 * \frac{Recall * Precision}{Recall + Precision}$$

Berechnung des F1-Scores [16]

Confusion Matrix (Wahrheitsmatrix): Mithilfe einer Wahrheitsmatrix können die Ergebnisse von Algorithmen bei Klassifizierungen dargestellt werden. Damit ist es möglich zu sehen, ob gefundene Ergebnisse korrekt sind, Ergebnisse nicht gefunden wurden oder gar falsch klassifiziert wurden. Die folgende Tabelle 2 zeigt die Wahrheitsmatrix mit den möglichen Ergebnissen.

Tabelle 2: Wahrheitsmatrix nach [16]

		Tatsächliche Werte	
		Positiv	Negativ
Vorhergesagte Werte	Positiv	TP	FP
	Negativ	FN	TN

Richtig positiv (TP / True Positive): Es wurde ein positiver Wert vorhergesagt und es ist korrekt.

Falsch positiv (FP / False Positive): Es wurde ein positiver Wert vorhergesagt und es ist falsch.

Richtig negativ (TN / True Negative): Es wurde ein negativer Wert vorhergesagt und es ist korrekt.

Falsch negativ (FN / False Negative): Es wurde ein negativer Wert vorhergesagt und es ist falsch.

3 Maschinelles Lernen

Maschinelles Lernen ist ein Teilbereich der künstlichen Intelligenz. Dabei soll eine Maschine bzw. ein Computerprogramm aus Erfahrungen lernen und eine Verhaltensänderung dessen bewirken. Anstatt etwas statisch zu programmieren, sollen Computer aus Daten, also Erfahrungen, ein Verhalten lernen, es sich aneignen und anwenden. Hierbei ist es allerdings nicht unbedingt das Ziel, dass ein Computer ständig weiterlernt, sondern sich ein gewisses Verhalten mithilfe von Daten einmalig aneignet und es anwendet. Beispielsweise kann ein Rasenmäher Roboter sein Verhalten verbessern, indem er Daten von seiner Umgebung mithilfe von Algorithmen verarbeitet. Das Resultat daraus kann sein, dass der Roboter effizienter den Rasen mäht, er dennoch keinerlei Intelligenz besitzt; „man spricht davon, das Verhalten an die Umgebung zu adaptieren [7]“.

Damit ein Computerprogramm eigenständig lernen und sich ein Verhalten aneignen kann, ist ein vorheriges Handeln eines Menschen notwendig. Beispielweise muss die Software zunächst mit ausgewählten Daten und Algorithmen ausgestattet werden. Weiterhin können Regeln aufgestellt werden, die das System beim Analysieren der Daten befolgen muss, um sich bestimmte Muster anzueignen [7]. Die Software kann dadurch auch unbekannte neue Daten verarbeiten, die relevanten Informationen extrahieren und weiterverarbeiten. Neben dem Begriff des maschi-

nellen Lernens wird oft von Knowledge Discovery in Databases (KDD) und Data-Mining gesprochen.

KDD beschreibt einen Prozess, indem fachliche Zusammenhänge aus großen Datenbeständen erkannt werden. Im Gegensatz zum Data-Mining befasst sich KDD auch um die Vorbereitung der Daten bzw. einer Bewertung der gefundenen Resultate. Abbildung 2 stellt den gesamten Prozess dar [7]. Ausgehend von einer Sammlung an Informationen wird (semi-)automatisch Wissen aus dieser Ansammlung gewonnen. Dabei läuft der Prozess interaktiv und iterativ, da der Anwender nach jedem Schritt Entscheidungen treffen muss oder Schritte u.U. wiederholt werden müssen.

Im ersten Schritt muss eine Auswahl der Daten, im Hinblick auf vorab definierte Ziele, getroffen werden. Diese Auswahl muss vorab getroffen werden, da die Ausgangsdatensammlung Informationen enthalten kann, die irrelevant sind. Im Anschluss müssen die Daten in der Vorverarbeitung bereinigt werden. Mithilfe der Transformation werden die Daten in eine einheitliche Form gebracht, die zur Weiterverarbeitung geeignet ist. Dies bedeutet, dass Daten in numerische Werte umgewandelt bzw. zusammengefasst werden. So könnten beispielsweise drei einzelne Werte zu einem zusammengefasst werden, wenn es die Datenstruktur erlaubt. Mithilfe einer geeigneten Methode aus dem Bereich des maschinellen Lernens können diese Daten verarbeitet werden. Daraus ergeben sich Muster, aus denen sich Wissen extrahieren lassen. Der letzte Schritt lässt sich ebenfalls mit dem Begriff Data-Mining beschreiben. Hierbei geht es, wie bereits erläutert, um die Gewinnung von Wissen aus bereits vorhanden Daten. Nicht jedoch um die Generierung der Daten [7].

4 Implementierung der Algorithmen

4.1 Scikit-Learn

Scikit-Learn ist eine freie Software-Bibliothek für die Programmiersprache Python. Sie wird im Bereich des maschinellen Lernens eingesetzt und baut auf die Python-Bibliotheken NumPy, SciPy und Matplotlib auf. Da sie zur freien Verfügung steht, sie eine gute Dokumentation besitzt und die Möglichkeit besteht sie kommerziell zu nutzen, hat die Bibliothek an großer Beliebtheit gewonnen [9]. Sie bietet eine Vielzahl an Klassifikations-, Regressions- und Clustering-Algorithmen und lässt sich durch einfach zu nutzende Schnittstellen in eigene Projekte integrieren [8].

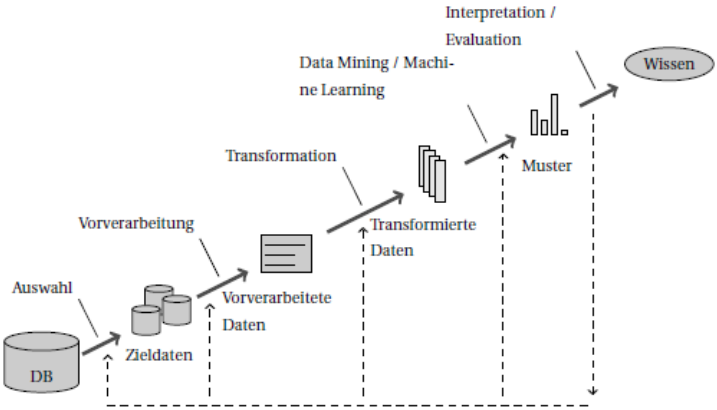

Abbildung 2: Knowledge Discovery in Databases als Prozess [7]

4.2 Weka

Waikato Environment for Knowledge Analysis (Weka) ist eine Software, die verschiedene Algorithmen für das maschinelle Lernen zur Verfügung stellt. Das Tool wurde in der Sprache Java von der University of Waikata entwickelt [13]. Da Weka als Open-Source angeboten wird, ist es möglich das Programm z.B. durch Pakete wie LibSVM zu erweitern, wodurch der Funktionsumfang um Vector Machines erweitert wird. Bei der Installation wird sowohl ein Handbuch als auch ein Ordner mit Beispiel-Datensätzen mitgeliefert. Dies vereinfacht den Einstieg in die Software und bietet ein praktisches Beispiel, wie sie zu nutzen ist.

5 Maschinelles Lernen in der Praxis

In diesem Kapitel werden zwei Forschungen aufgeführt, die unterschiedliche Algorithmen nutzen und deren erzielten Ergebnisse präsentiert. Das Ziel der Arbeiten ist es Anomalien innerhalb der Datensätze zu erkennen und sie richtig zu klassifizieren. Durch den Einsatz von maschinell lernenden Algorithmen können Intrusion Detection Systeme genauere Vorhersagen treffen, wodurch die Schutzziele der IT-Sicherheit besser geschützt sind. Auf eine detaillierte Beschreibung der Algorithmen wird verzichtet. Zur näheren Betrachtung kann unter [7, 11, 17] nachgeschlagen werden.

Die Arbeit von U. Ahmad et al. [12] nutzt die Software-Bibliothek Scikit-Learn, um unterschiedliche Algorithmen miteinander zu vergleichen und eine Aussage über deren erzielten Ergebnisse zu treffen. Insgesamt wurden sechs unterschiedliche Algorithmen auf den NSL-KDD Datensatz angewendet. Die verwendeten Algorithmen sind folgende: *Decision Tree, Naïve Bayes, Ada Boost, Multi-Layer Perceptron (MLP), Random Forest* und *Linear Support Vector Machine (Linear SVM)*. Hierfür wurden die implementierten Algorithmen zunächst mit den NSL-KDD Trainingsdaten angelernt, um ein entsprechendes Verhalten zu erzielen. Im Anschluss wurden die Testdaten genutzt, um die sechs Algorithmen miteinander vergleichen zu können. Die folgende Tabelle 3 zeigt einen Vergleich der eingesetzten Algorithmen. Zu sehen ist, dass alle Algorithmen beim Erkennen der Angriffe gute Ergebnisse erzielt haben. Besonders gut haben die Algorithmen MLP, Decision Tree und Linear SVM abgeschnitten mit einer Genauigkeit von 1,0; 0,986 und 0,974.

Tabelle 3: Vergleich der erzielten Ergebnisse der Algorithmen [12]

Classifier	Accuracy	Precision	Recall	F1-Score
MLP	1.0	1.0	1.0	1.0
Decision Tree	0.986	0.99	0.99	0.99
Naïve Bayes	0.765	0.84	0.77	0.77
Ada Boost	0.743	0.84	0.74	0.75
Random Forest	0.743	0.81	0.74	0.75
Linear SVM	0.974	0.97	0.97	0.97

Ein ähnliches Ergebnis lässt sich aus der Wahrheitsmatrix entnehmen. Dort haben die Algorithmen MLP (Tabelle 4) und SVM (Tabelle 5) gut abgeschnitten bzw. MLP keine falschen Aussagen getroffen. Zum Vergleich zeigt der Algorithmus Random Forest (Tabelle 6) ein schlechteres Ergebnis indem über 6000 Falsch Positive Erkennungen stattgefunden haben.

Tabelle 4: Wahrheitsmatrix des MLP Algorithmus [12]

MLP		Actual Labels		Total
		Anomalous	Normal	
Predicted Labels	Anomalous	TP = 9710	FN = 0	9710
	Normal	FP = 0	TN = 12833	12833
	Total	9710	12833	22543

Tabelle 5: Wahrheitsmatrix des SVM Algorithmus [12]

SVM		Actual Labels		Total
		Anomalous	Normal	
Predicted Labels	Anomalous	TP = 9570	FN = 140	9710
	Normal	FP = 3	TN = 12830	12833
	Total	9573	12970	22543

Tabelle 6: Wahrheitsmatrix des Random Forest Algorithmus [12]

Random Forest		Actual Labels		Total
		Anomalous	Normal	
Predicted Labels	Anomalous	TP = 9101	FN = 609	9710
	Normal	FP = 6285	TN = 6548	12833
	Total	15386	7157	22543

Die Arbeit von M. Almseidin et al [11] nutzt die Software WEKA, um sieben unterschiedliche Algorithmen auf den KDD Datensatz anzuwenden. Die verwendeten Algorithmen sind folgende: *J48, Random Forest, Random tree, Decision table, MLP, Naive Bayes* und *Bayes Network*. Mithilfe von insgesamt 148 753 zufällig ausgewählten Einträgen des Datensatzes, wurden die Algorithmen zunächst trainiert. Im Anschluss wurden weitere 60 000 zufällig ausgewählte Einträge genutzt, um deren Verhalten zu testen. Tabelle 7 zeigt die Ergebnisse der Untersuchung. Darin lässt sich zunächst erkennen, dass alle Algorithmen gute Ergebnisse von

Tabelle 7: Vergleich der erzielten Ergebnisse der Algorithmen [11]

Machine Learning Classifiers	Correctly classified Instances	incorrectly classified Instances	Accuracy Rate
J48	55865	4135	93.10%
Random Forest	56265	3735	93.77%
Random tree	54345	5655	90.57%
Decision table	55464	4536	92.44%
MLP	55141	4859	91.90%
Naive Bayes	54741	5259	91.23%
Bayes Network	54439	5561	90.73%

mindestens 90,57% Genauigkeit erreicht haben. Die höchste Genauigkeit erreicht der Algorithmus *Random Forest* mit 93,77%. Von insgesamt 60 000 Einträgen wurden lediglich 3735 nicht korrekt erkannt. Der Algorithmus *Random tree* hat mit 90,57% die niedrigste Genauigkeit und insgesamt 5655 Einträge nicht korrekt erkannt.

6 Diskussion

Die Suche nach Algorithmen hat ergeben, dass sich prinzipiell alle gezeigten Algorithmen eignen, um ein IDS zu trainieren. Jedoch weisen sie teilweise große Unterschiede in ihrer Genauigkeit auf. Der Algorithmus MLP erreicht in beiden Arbeiten gute Ergebnisse bzw. in der Untersuchung von U. Ahmad et al. [12] sogar eine Genauigkeit von 100%. Auffällig sind vor allem die Ergebnisse der Algorithmen *Random Forest* und *Naive Bayes*. Während sie in der Arbeit von U. Ahmad et al. [12] lediglich eine Genauigkeit von 74,3% und 76,5% erreichen, schneiden sie in der Arbeit von M. Almseidin et al [11] mit einer Genauigkeit von 93,77% und 91,23% ab. Dieses Ergebnis ist vor allem deshalb interessant, da beide Arbeiten prinzipiell mit den gleichen Daten trainiert haben. Die Arbeit von U. Ahmad et al. [12] nutzt im Gegensatz zur Forschung von M. Almseidin et al [11], den NSL KDD Datensatz. Dieser ist eine überarbeitete Version des ursprünglichen KDD Datensatzes wie bereits in Kapitel 2.4 erläutert wurde. Um eine bessere Aussage treffen zu können, würde es sich anbieten die übrigen Algorithmen mit dem jeweils anderen Datensatz zu trainieren und zu testen. Möglicherweise könnte dabei ein gewisser Trend in Bezug zur Genauigkeit erkannt werden. Ebenfalls wäre eine weitere Untersuchung mit einem dritten Datensatz sinnvoll, um dessen Ergebnisse mit den bestehenden zu vergleichen. Aus den Ergebnissen beider Arbeiten geht hervor, dass sich der Algorithmus *MLP* besonders zu eignen scheint, da mit ihm die besten Ergebnisse erzielt wurden.

Mithilfe der Software-Bibliothek Scikit-Learn und der Software Weka können die Algorithmen eingesetzt werden. Die vorgestellten Datensätze unterscheiden sich vor allem in ihrer Beschaffenheit. Das bedeutet, dass beispielsweise der Datensatz KDD Cup 1999 noch immer genutzt werden kann, um ein IDS zu trainieren. Aufgrund der fehlenden Vielfalt an öffentlich zugänglichen Datensätzen, ist die Auswahl beschränkt. Es bietet sich an den überarbeiteten Datensatz NSL-KDD zu nutzen, da er auf dem ursprünglichen KDD Datensatz von 1999 basiert, jedoch viele Kritikpunkte verbessert wurden. Außerdem unterscheiden sich die Angriffsvektoren von Früheren nur bedingt. Sie sind vielmehr eine weitere Variation bereits bestehender, wodurch moderne Angriffe ebenfalls erkannt werden können.

7 Fazit

Es zeigt sich, dass sich durch den Einsatz des maschinellen Lernens gute Ergebnisse erzielen lassen und sich dadurch eine gute Vorhersage treffen lässt, wie das IDS verbessert wird. Der Algorithmus *MLP* kann daher in eine engere Auswahl aufgenommen und in einer späteren Arbeit genauer untersucht

werden. Dennoch sollten die anderen Algorithmen nicht von vornherein ausgeschlossen werden, da sie wie die Untersuchung von M. Almseidin et al [11] zeigt, ebenfalls gute Ergebnisse erzielen können. Vor allem hat diese Arbeit gezeigt, dass es sinnvoll sein kann einen weiteren Datensatz zu nutzen bzw. einen eigenen Datensatz zu produzieren, um die Performance der Algorithmen erneut zu testen. In den Untersuchungen haben zwar fast alle Algorithmen gut abgeschnitten, jedoch bleibt die Frage offen, weshalb sich die Ergebnisse der beiden Algorithmen unterscheiden und welche Ergebnisse mit einem anderen Datensatz erzielt werden können.

Literaturverzeichnis

[1] M. Kofler et al. Hacking & Security. Das umfassende Handbuch. Bonn: Rheinwerk Verlag, 2018.

[2] C. Eckert. IT-Sicherheit. Konzepte – Verfahren – Protokolle, 10. Auflage. Berlin: De Gruyter, 2018.

[3] W. Stallings, L. Brown. Computer Security. Principles and Practice, 4th Edition. London: Pearson, 2018.

[4] C. Liebig. Sicherheitsanalyse von mobilen Geschäftsanwendungen. Bremen, 2014.

[5] K. Scarfone, P. Mell. Guide to Intrusion Detection and Prevention Systems (IDPS). National Institute of Standards and Technology. Gaithersburg, 2007.

[6] K. Jucknischke. IT-Sicherheit – Das ZigBee Smart Home steht offen. Bachelorarbeit. Reutlingen, 2019.

[7] J. Frochte. Maschinelles Lernen. Grundlagen und Algorithmen in Python, 2. Auflage. Carl Hanser Verlag GmbH & Co. KG. München, 2019.

[8] F. Pedregosa et al. Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830, 2011.

[9] D. Paper. Hands-on Scikit-Learn for Machine Learning Applications: Data Science Fundamentals with Python. Apress. Logan, UT, USA, 2020.

[10] M. Tavallaee, E. Bagheri, W. Lu, A. A. Ghorbani. A detailed analysis of the kdd cup 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, pp. 1-6, 2009.

[11] M. Almseidin, M. Alzubi, S. Kovacs, M. Alkasassbeh. Evaluation of Machine Learning Algorithms for Intrusion Detection System. Department of Information Technology, University of Miskolc, H-3515 Miskolc, Hungary, 2017.

[12] U. Ahmad, H. Asim, M. T. Hassan, S. Nasser. Analysis of Classification Techniques for Intrusion Detection. International Conference on Innovative Computing (ICIC), 2019.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten. The weka data mining software: an update. ACM SIGKDD explorations newsletter, vol. 11, pp. 10–18, 2009.

[14] KDD Cup 1999 Data, 2020. Online verfügbar unter: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html; Besucht am 05.10.2020.

[15] NSL-KDD dataset, 2020. Online verfügbar unter: https://www.unb.ca/cic/datasets/nsl.html; Besucht am 05.10.2020.

[16] Machine-Learning-Modelle bewerten – die Crux mit der Metrik, 2020. Online verfügbar unter: https://blog.codecentric.de/2019/07/machine-learning-modelle-bewerten-die-crux-mit-der-metrik/; Besucht am 05.10.2020.

[17] V.K. Ayyadevera. Pro Machine Learning Algorithms. Apress, Berkeley, CA, 2018.

GAN and their Chances and Risks in Face Generation and Manipulation

Anna Taphorn
Hochschule Reutlingen
Anna_Elisabeth.Taphorn@Student.Reutlingen-University.de

Abstract
Generative adversarial networks (GAN) are powerful deep learning architectures that revolutionized machine learning regarding the synthesis of data. The manipulation and generation of realistic face images is an important field in computer vision that can be realized by GAN. This leads to a high scientific interest and to many economic and artistic applications. However, there are some risks and challenges in GAN training that should be regarded. Hence, the question arises if GAN contribute a real value.
In this work the basic functionality of GAN will be outlined. Furthermore, their chances and risks with regard to facial images will be critically examined and evaluated.

CCS Concepts
• Computing methodologies → Machine learning → Learning paradigms → Adversarial Learning

Keywords
Generative Adversarial Networks, Deep Fakes, Deep Generative Models

Academic supervisor: Prof. Dr. Cristóbal Curio
Hochschule Reutlingen
Cristobal.Curio@Reutlingen-University.de
External supervisor: Dr. Johannes Stelzer
Colugo GmbH
Stelzer@Colugo.ai

Informatics Inside Fall 2020
25. November 2020, Hochschule Reutlingen

1 Introduction
Generative adversarial networks (GAN) are machine learning architectures that are designed to learn deep representations of high-dimensional data without the need for labelled datasets. Since they were introduced by Goodfellow et al. [19], GAN massively drive the progress of artificial intelligence and computer vision. Countless variations and improvements of GAN have emerged in the past years and many applications are based on this architecture.
Face processing and synthesis is a wide field in generative deep learning, likely because it is connected with many practical tasks like face recognition and editing. The synthesis of high-quality face images is made possible through GAN, which is not trivial due to the complex and high-dimensional features faces have.
This leads to the question, how GAN work and how relevant they are in the field of face generation and manipulation. Since a comparison of benefits and disadvantages of GAN can barely be found in literature, this work serves the critical examination of this topic. In the following chapters the basic functionality, practical relevance and risks of GAN in face processing will be outlined and finally evaluated.

2 Generative Adversarial Networks
In this chapter the functional background and training challenges of GAN will be outlined. Furthermore, some extracts from the current

state of the art are presented to illuminate the progress and potential of GAN.

2.1 Background

Generative adversarial networks (GAN) were introduced in 2014 by Goodfellow et al. [19]. They described a machine learning framework consisting of two counterpart models: a generative model *G* that simulates data based on a random noise vector in the latent space and a discriminative model *D* that estimates if given samples are created from the generator or originate from the training set. The training is designed as a minimax-game in which the aim of *D* is to maximize the rate of correct predictions while *G* tries minimize the success of *D* by replicating the training data. This results in the following adversarial loss function *V (G, D)* which is derived from binary cross entropy:

$$\min_G \max_D V(D,G) = E_x\left[\log\left(D(x)\right)\right] + E_z\left[\log\left(1 - D\left(G(z)\right)\right)\right],$$

where D(x) represents the probability that data x came from the dataset rather than the generator´s distribution, G(z) is the output of the generator with given noise z, D(G(x)) is the estimated probability of a fake sample being real, E_x is the expected value over all real data instances and E_z is the expected value over all random inputs to the generator [20].

As *D* and *G* are built of multilayer perceptrons, the weights of both models can be updated by backpropagation which leads to mutual improvements during the training procedure. The layers in both models are fully connected. The training objective is achieved when the discriminator cannot distinguish real and fake so that D(x) is 0.5 [19]. Goodfellow et al. [19] were using the algorithm with three different datasets to generate handwritten digits, faces as well as animals and vehicles. GAN can be used in an unsupervised context as well as in semi- and supervised manners [11]. After Goodfellow et al. laid the foundation for generative adversarial training, many GAN variations followed. In 2016 Radford et al. [43] introduced the deep convolutional architecture DCGAN for unsupervised learning that uses a convolutional neural network (CNN) as a discriminator and a transposed CNN for the generator. DCGAN shows improved stability [27]. LAPGAN [14] can increase the resolution of given input images with the use of Laplacian pyramids, and conditional GAN (CGAN) [39] feed a class label as an information vector to generator and discriminator to achieve supervised training.

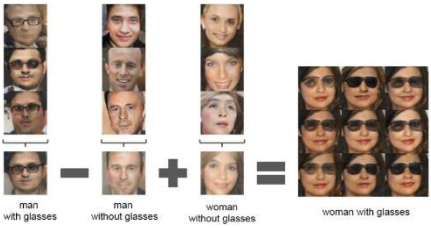

Figure 1: Arithmetic with latent vectors. Three face images are averaged for stable results. Retrieved from [43].

The mentioned variations are only exemplary extracts from the variety of GAN algorithms to illustrate the flexibility and semantic power of GAN. As a matter of fact, Radford et al. [43] found, that it is possible to apply simple arithmetic to the generator´s vectors to create samples with specific semantic features. Figure 1 shows an example in which the averaged *z* vectors of men with and men and women without glasses are combined to generate new vectors of women with glasses. As the *z* vectors represent learned features the results differ significantly from arithmetic in pixel space [43]. This implicates that the network has a kind of graphical understanding of the concept of glasses.

2.2 State of the art

Because of their generative potential GAN have evolved in many directions in the past years. StyleGAN, a recent development by Karras et al. [30, 31], is able to produce photorealistic face images with high resolution

and can even combine content and style of specific images. The style-based generator architecture makes use of an intermediate latent space *W* that controls the generator through adaptive instance normalization instead of directly using the latent code *z* as input of the generator [30]. In StyleGAN the generator learns to separate different aspects of facial images into content and style similar to style transfer [17].

The first attempt to get high resolution images from StyleGAN was a progressive growing of the image resolution, starting with a few pixels and increasing the amount of pixels with each layer in the model [30]. But Karras et al. found that progressive growing is not the ideal approach because it leads to specific artifacts. They therefore use a skip generator and a residual discriminator instead [31].

Similar to StyleGAN Choi et al. [8, 9] proposed StarGAN, an architecture that captures style and content of face images. It learns the mappings between the available domains with a single generator. This enables a transfer from one image to multiple domains like gender and emotion and therefore can change facial attributes like age or gender as well as emotions.

Abdal et al. [1] stated that the manipulation of given images would bring more benefits than generating random faces. Hence, they investigated the implementation of given images into the latent space of StyleGAN with promising results.

They use an extended latent space *W+* on a pre-trained StyleGAN which allows them to successfully embed face images as well as non-facial images using an optimization-based approach. It turns out that the embedding of face images is semantically meaningful concerning morphing, style transfer and expression transfer whereas the structure of embedded non-facial images gets lost [1].

Shen et al. [45] proposed a similar approach to real image processing and were able to apply learned semantic attributes to given images through varying the latent codes of a GAN model. They successfully manipulated attributes like pose, emotion, age, gender and glasses in given face images with the StyleGAN architecture. The architecture can even correct artifacts in GAN generated images.

2.3 Challenges

Although GAN have evolved much during the past years, the training of GAN still comes along with some challenges that make the training difficult. In the following chapters some typical problems are examined.

2.3.1 Vanishing Gradients

The problem of vanishing gradients can occur in any gradient-based and backpropagation training. Arjovsky and Bottou [2] found that confident discriminators in GAN can cause very small gradients that do not provide sufficient information for the generator to improve. This can lead to training stagnation. A common practice is using Wasserstein loss proposed by Arjovsky et al. [3] instead of the minimax loss in order to to prevent the gradients from vanishing even when the discriminator is trained to its optimum. The Wasserstein loss is discussed in section 2.3.2.

2.3.2 Mode Collapse

A problem that often occurs in GAN training is the generator creating output with limited diversity of samples. This is known as mode collapse [38] that happens when the generator fails to generalize because it can fool the discriminator with a single mode that does not need to be diversified. Thereupon the loss function collapses to almost 0 and even training of the discriminator for the corresponding mode would just lead to the generator choosing another one [15].

Although mode collapse is a well-known problem in adversarial training, it has not been entirely solved yet. One way to remedy

mode collapse is to use the loss of Wasserstein GAN (WGAN) presented by Arjovsky et al. [3]. It prepares modified loss functions for both, generator and discriminator, that regard the generator´s convergence and output quality and stabilizes the optimization process [3]. WGAN uses the Wasserstein distance which determines the distance between probability distributions.

The WGAN loss minimization for *D* is $min_D -\left(E_x[f_w(x)] - E_z\left[f_w\left(G(z)\right)\right]\right)$, and the WGAN loss minimization for *G is* $min_G -\left(E_z\left[f_w\left(G(z)\right)\right]\right)$. Here, f_w is *D* with additional constraints so the loss output gets clipped to a minimum and maximum rather than ranging between $[-\infty, \infty]$.

The major difference between minimax loss and WGAN is that with Wasserstein loss *D* serves to fit the Wasserstein distance rather than being a binary classifier [27].

2.3.3 Non-Convergence

In contrast to other deep learning models that usually utilize optimization methods, the non-cooperative minimax game between *G* and *D* is required to result in a Nash equilibrium [44]. In the Nash equilibrium no player has the incentive to change its strategy, regardless of the opponent´s actions. However, this state is not guaranteed to reach and it happens that the parameters oscillate or destabilize so that the training fails to converge [44]. The instability of GAN training is one of the largest problems when using GAN [28].

The problem of non-convergence can be diminished through instance noise like independent Gaussian noise added to the data points [2] or with zero-centered gradient penalties [37]. The detailed explanation of these two procedures is beyond the scope of this paper but can be found in [37].

2.3.4 Other Challenges

Some GAN show deficits in representing rich class structures which results in poor perspective, shape and feature amounts. It occurs, that output images display class-related characteristics, but the visible features are not in the correct position or the amount is unrealistic. Therefore, it produces for example dogs with five eyes and a skewed shape that is flat and prospectively wrong [28].

Besides that, since the generator is always graded against the current discriminator, the loss function becomes uninformative over time [15]. A low generator loss does not indicate a high output quality because the discriminator rates on a viewpoint related basis. This makes it difficult to quantitatively evaluate and compare GAN models. Finding suitable measures for validating GAN is often a subject in research. Furthermore, when *G* is outperforming *D* which then guesses on an almost random level, it can cause a quality drop because *G* gets trained on the basis of poor feedback [15].

3 Other Deep Generative Models

GAN are not the only generative deep learning models that are used to generate high-dimensional data. In the following sections, three other models are presented.

3.1 Boltzmann Machines

A Boltzmann machine is a stochastic network architecture that consists of symmetrically connected neurons with probabilistic weights [24]. The weights are updated through backpropagation and affect the binary values of the neurons.

Because large general Boltzmann machines are hard to train, only restricted Boltzmann machines (RBM) have a practical value in machine learning. RBM have only one hidden layer. Within the layers there are no interconnections, but every visible neuron is connected to every hidden neuron in the adjacent layers [24]. RBM are used in Deep Belief Networks (see section 3.2).

The restriction of Boltzmann machines is necessary for training but also a disadvantage. GAN have fewer restrictions in their structure in contrast to RBM which

makes them more flexible and allows adjustments for many different requirements [28].

3.2 Deep Belief Nets

Deep Belief Networks (DBN) [25, 26] are similar to Boltzmann machines probabilistic deep generative models built of multiple hidden layers without intralayer connections. They usually consist of stacked RBM. Stacking is used so that the fixed output of one RBM serves as input for a higher-level one. The layer of visible neurons therefore represents data, and the hidden layer learns the feature representation. As soon as the training for one RBM is completed, the weights are locked and another RBM can be put on top [25].

Figure 2: Faces generated with DBN. Retrieved from https://cutt.ly/dbn.

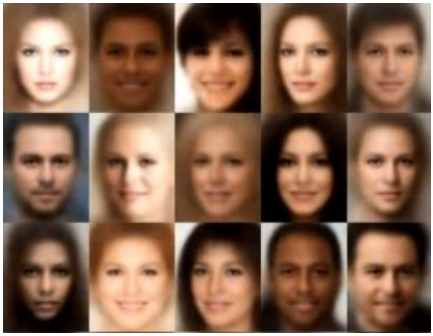

Figure 3: Face images generated with VAE. Retrieved from https://cutt.ly/Vae.

In its beginnings this algorithm was game-changing because the training could have a reasonably high depth [41]. DBN have downward directed connections between all layers except for the top two layers. They can learn complex representations in an unsupervised manner that enable image recognition and generation [25]. Figure 2 shows an example of face images that were generated with a DBN.

DBN are slower than GAN because they need runtime proportionally to the data, while GAN are able to generate samples in parallel [28]. Furthermore, Goodfellow rates the output of GAN better than DBN outputs [28]. Recent literature about DBN concerning face images is often referred to face recognition and classification tasks which leads to the conclusion that they were superseded by GAN.

3.3 Variational Autoencoders

Besides GAN, variational autoencoders (VAE) [32] are an unsupervised learning approach with high potential in realistic data generation. Autoencoders are neural networks that are designed to learn efficient data representations and to reconstruct the input data accurately with reduced dimensionality. They consist of an encoder for compression and a decoder for reconstruction. In the encoding process the model learns a latent space representation of the underlying data characteristics with which the difference between original and reconstructed data can be minimized [15]. VAEs are regularized autoencoders that are able to generalize on the basis of a whole data batch rather than a single data entity.

GAN and VAE are both relevant generative deep learning models. As shown in Figure 3, image generation with VAE often leads to blurriness and a lower subjective perceptual quality than with GAN. On the other hand, VAE training is more stable [18] and VAE are able to represent the entire data distribution while GAN tend to lack full support over the data [21, 33]. Some algorithms utilize a hybrid of VAE and GAN to get the advantages of both models [7, 16, 21].

4 Application fields

GAN have developed a lot since their first introduction and hence the architecture is not only interesting in scientific settings but has long since found application in various fields. In this chapter the economic and artistic value of GAN with respect to facial images will be outlined.

4.1 Commercial perspectives

The generation and manipulation of realistic face images with GAN have opened a wide field for different applications that can have economic value. GAN can be used for an automated enhancement of facial attractiveness in photos [29], expression editing [35], age manipulation [46] or general face editing. For instance, the application PortraitPro[1] is using GAN and other deep learning algorithms for face editing and reenactment.

Furthermore, GAN enable image and face reconstruction when parts are covered or corrupted [6, 57]. This is also known as image inpainting. Some approaches enable image resolution enhancement [6] which is used in some image applications like the online ImageUpscaler[2]. Game and animation design as well as movie post-production can profit from automized facial animation [13, 50] that enables an easy and flexible face reenactment or animation. Animations can even be created from single images [42].

GAN have also facilitated face recognition as it enables low and cross resolution face recognition [47] as well as pose-independence [49].

Besides useful applications for organizations and individuals, GAN have gained relevance since they can be used to support police investigations. The Parliament in Germany has recently approved the use of computer generated pornographic images of non-real children to combat abuser networks [23]. With the help of automatically created fake material investigators are able to infiltrate pedophile platforms as pornographic material is often needed to gain access.

4.2 Art

Artificial intelligence is often used for generating artworks, likely because math-based algorithms in combination with creativity evoke fascination. Especially GAN offer a wide range of abilities to create new designs and unprecedented images.

Several artists are using GAN for their generation of creative works (e.g. [51]). In 2018 a GAN artwork named “Portrait of Edmond Belamy” was sold for more than 430,000$ [10].

The creative value of GAN is not restricted to the use by professional artists. Some websites present GAN-created vessel designs, dresses, song lyrics or even humorous memes and satire[3]. GAN also enable the automation of artistic portrait generation from face images [54]. The open source platform ArtBreeder[4] is based on the GAN architecture BigGAN and enables the creation of unique artworks ranging from abstract art over avatars and landscapes to artistic portraits and realistic faces. The underlying pedigree of each image is saved and can be explored as shown in Figure 4. Users can additionally feed the network with own images.

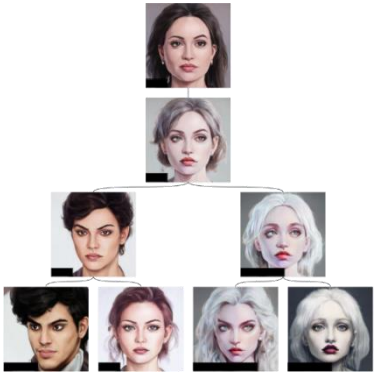

Figure 4: Pedigree of example portraits. Any portrait is created by ArtBreeder. Retrieved from www.artbreeder.com.

[1] www.anthropics.com/portraitpro/
[2] www.imageupscaler.com
[3] See www.thisxdoesnotexist.com
[4] www.artbreeder.com

Furthermore, it is possible to edit the semantic attributes and get crossbreeds and children of the generated image. This makes it a powerful tool for infinite creations.
All in all, the mentioned aspects reveal the value and perspective of GAN models and other kinds of artificial intelligence (AI) in artistic contexts.

5 Risks of GAN

Although GAN have many practical benefits, they have induced some bad aspects besides training challenges (section 2.3) that should be taken into account when discussing the model. A major disadvantage of the powerful AI technology is the misuse for unethical or criminal purposes. DeepFakes are a rising problem that will be discussed in the following section.

5.1 DeepFakes

The well-known term DeepFake represents realistic fake content that was manipulated with the help of deep learning algorithms [48]. The term DeepFake in general is not only negatively afflicted, however as it is deeply connected with harmful applications and misleading content this chapter exclusively refers to the negative connotation of DeepFakes.
Due to the availability of sophisticated algorithms and open source datasets, the creation of imperceptible fake content has become facile. DeepFakes can be used to sabotage people, organizations and information realistically with low effort. Architectures like GAN are predestined for tasks like these, especially those models that are capable of human face processing. Face processing is a major part of DeepFakes, likely because faces appear trustworthy and are hard to fake due to the detailed face perception of humans. [48] make up three categories of DeepFakes in facial images in which GAN are playing an important role:

i) the synthesis of entire non-existent faces. Approaches like StyleGAN [31] can be used for this. This kind of DeepFake could be used for fake avatars in social media.
ii) the manipulation of facial attributes like age, gender, skin color and many others. StarGAN [9] is capable of implementing this. There are also some applications like FaceApp[5] that make use of this kind of face editing.
iii) expression swap or face reenactment to change facial expressions. This is a powerful field as it can be used to change misrepresent people in images or even fake their speech in videos.

DeepFakes can be harmful for individuals, for example when they are misrepresented in media or their statements are forged. A rising problem is people getting transposed into pornographic content without their knowledge or agreement [12, 48].
DeepFakes also pose a threat to personal and organizational security because scam and blackmailing are facilitated. Social engineering is one of the use cases for DeepFakes. Furthermore, the generation of fake news and misleading information can affect society. With false facts and news people can be misled, unsettled, intimidated or persuaded. This can influence behavior and opinion of individuals and groups and even have an impact on the political consensus and elections. During the U.S. elections in 2016, fake news and misinformation have already been ubiquitous and probably have had an impact on votes [4, 34]. Besides that, the possibility to fake any digital information can cause distrust and insecurity towards media and science among the population.
All in all, it cannot be denied that DeepFakes with GAN and other deep learning technologies can pose a threat to individuals, society and democracy and therefore must be prevented. While continuously improving GAN algorithms, it should always be considered what a powerful AI can wreak when it gets into the wrong hands.

[5] www.faceapp.com

5.2 Counteracting DeepFakes

To counteract the increasing amount of artificial data the field of DeepFake detection is gaining growing attention. Determining manipulated face images is challenging due to the realistic, qualitative results that can be generated. The automatic detection of AI generated fakes can be performed with similar deep learning algorithms. GAN images usually have specific artifacts that characterize them like fingerprints [48].

[22], [55] and [56] detect those artifacts with the help of neural networks and are able to differentiate GAN generated and real images. [36] stated that GAN images show typical color schemes and propose a promising forensic based on the frequency of over-exposed pixels.

Wang et al. [52] chose an approach that is based on monitoring of neuron behavior when feeding fake and real face images into a neural network, since neuron activation patterns can capture subtle differences. Thus, they are able to detect synthesized images as well as attribute manipulation and face reenactment. Similarly, [40] propose an architecture to spot different kinds of GAN generated image manipulations through pixel co-occurrence on the image RGB channels that are fed to a convolutional neural network. [5] used a deep learning architecture consisting of a supervised RBM.

It is also possible to prevent image manipulation from the outset. [53] marginally modify images to protect them so any manipulation creates visible artifacts and is easy to detect.

6 Discussion

The previous chapters have outlined the challenges, risks and chances of GAN. Therefore, the question arises if GAN have a legitimate use in computer science and general applications or if they pose a risk and require too much effort.

Regarding other generative deep learning algorithms, it becomes apparent that GAN are exceptional in their flexibility and output quality referring to face images. Moreover, in comparison to Deep Belief Networks, they seem to be faster in training because of parallelizable tasks.

On the other hand, GAN involve some challenges that make them difficult to train. In comparison to Variational Autoencoders, the training is very unstable, and it can happen, that it does not converge. Moreover, GAN are not able to support the entire data distribution. It can furthermore happen, that gradients vanish so that the generator is no longer able to improve on the basis of the discriminator´s feedback. Another well-known problem is mode collapse in which the generator produces samples basing on a single mode. However, approaches like Wasserstein loss, that avoid the mentioned issues, exist and can be implemented with low effort. Therefore, it is just a question of implementing the right strategies.

Another problem of GAN is an ethical one, because they facilitate the creation of misleading DeepFakes. Those pose a threat to society and should be defeated. There are several technologies to detect or avoid DeepFakes and most of them are practical and reliable. But as fake generators will evolve over time, it is an endless race of technology in which detectors must always be able to keep up. In conclusion, the possibility of misusage should always be regarded when developing GAN and evolving detection systems should be an ongoing process. Most importantly, people need to get media competencies and awareness that enable detecting and coping with DeepFakes.

In the end, GAN enable unprecedented extensive applications in economic, artistic and scientific fields, that constitute a high gain. No other technology can replace the capabilities of GAN and as there is ongoing progress and existing solutions for challenges, it can be said that the benefits outweigh the disadvantages.

7 Conclusion

In this work the basic functionality of GAN was outlined as well as their training challenges and risks regarding DeepFakes. A comparison with similar deep generative

models was presented and the practical applications for art and commercial purposes were specified. The final evaluation leads to the assumption, that GAN can be seen as a valuable gain for machine learning and computer science, although some disadvantages should be regarded.

References

[1] Abdal, R., Qin, Y., and Wonka, P. 2019. Image2StyleGAN: How to embed Images into the StyleGAN Latent Space? In *Proceedings of the IEEE International Conference on Computer Vision*, 4432–4441.

[2] Arjovsky, M. and Bottou, L. 2017. Towards Principled Methods For Training Generative Adversarial Networks. *arXiv preprint arXiv:1701.04862*.

[3] Arjovsky, M., Chintala, S., and Bottou, L. 2017. Wasserstein GAN. *arXiv e-prints*, arXiv:1701.07875.

[4] Benjakob, O. 2020. *From Israel to the U.S., Deepfake Videos Are Becoming a Major Threat to Democracy*. https://www.haaretz.com/israel-news/tech-news/.premium-from-israel-to-the-u-s-deepfake-videos-are-becoming-a-major-threat-to-democracy-1.9180187. Accessed 9 October 2020.

[5] Bharati, A., Singh, R., Vatsa, M., and Bowyer, K. W. 2016. Detecting Facial Retouching Using Supervised Deep Learning. *IEEE Transactions on Information Forensics and Security* 11, 9, 1903–1913.

[6] Cai, J., Hu, H., Shan, S., and Chen, X. 2019. FCSR-GAN: End-to-End Learning for Joint Face Completion and Super-Resolution. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, 1–8.

[7] Chen, H.-Y. and Lu, C.-J. 2019. Nested Variance Estimating VAE/GAN for Face Generation. In *2019 International Joint Conference on Neural Networks (IJCNN)*, 1–8.

[8] Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., and Choo, J. 2018. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8789–8797.

[9] Choi, Y., Uh, Y., Yoo, J., and Ha, J.-W. 2020. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8188–8197.

[10] Christie´s Inc. 2018. *Is artificial intelligence set to become art's next medium?*

[11] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. 2018. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine* 35, 1, 53–65.

[12] Croft, A. 2019. *From Porn to Scams, Deepfakes Are Becoming a Big Racket—And That's Unnerving Business Leaders and Lawmakers*. https://fortune.com/2019/10/07/porn-to-scams-deepfakes-big-racket-unnerving-business-leaders-and-lawmakers/.

[13] Das, D., Biswas, S., Sinha, S., and Bhowmick, B. 2020. Speech-Driven Facial Animation Using Cascaded GANs for Learning of Motion and Texture. In *European Conference on Computer Vision*, 408–424.

[14] Denton, E. L., Chintala, S., szlam, a., and Fergus, R. 2015. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama and R. Garnett, Eds. Curran Associates, Inc, 1486–1494.

[15] Foster, D. 2019. *Generative deep learning: teaching machines to paint, write, compose, and play*. O'Reilly Media.

[16] Gao, R., Hou, X., Qin, J., Chen, J., Liu, L., Zhu, F., Zhang, Z., and Shao, L. 2020. Zero-VAE-GAN: Generating Unseen Features for Generalized and Transductive Zero-Shot Learning. *IEEE Transactions on Image Processing* 29, 3665–3680.

[17] Gatys, L. A., Ecker, A. S., and Bethge, M. 2015. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*.

[18] Genevay, A., Peyré, G., and Cuturi, M. 2017. GAN And VAE From An Optimal Transport Point Of View. *arXiv preprint arXiv:1706.01807*.

[19] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680.

[20] Google Developers. *Loss Functions. Minimax Loss*. https://developers.google.com/machine-learning/gan/loss. Accessed 24 September 2020.

[21] Grover, A., Dhar, M., and Ermon, S. 2017. Flow-GAN: Combining Maximum Likelihood And Adversarial Learning In Generative Models. *arXiv preprint arXiv:1705.08868*.

[22] Guarnera, L., Giudice, O., and Battiato, S. 2020. DeepFake Detection by Analyzing Convolutional Traces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 666–667.

[23] Hallam, M. and Pieper, O. 2020. *Germany: Online child abuse investigators to get more powers*. https://www.dw.com/en/germany-online-child-abuse-investigators-to-get-more-powers/a-52037583. Accessed 8 October 2020.

[24] Hinton, G. E. 2007. Boltzmann machine. *Scholarpedia* 2, 5, 1668.

[25] Hinton, G. E. 2009. Deep Belief Networks. *Scholarpedia* 4, 5, 5947.

[26] Hinton, G. E., Osindero, S., and Teh, Y.-W. 2006. A Fast Learning Algorithm For Deep Belief Nets. *Neural computation* 18, 7, 1527–1554.

[27] Hong, Y., Hwang, U., Yoo, J., and Yoon, S. 2019. How Generative Adversarial Networks And Their Variants Work: An Overview. *ACM Computing Surveys (CSUR)* 52, 1, 1–43.

[28] Ian Goodfellow. 2017. *NIPS 2016 Tutorial: Generative Adversarial Networks*.

[29] Jingwu He, Chuan Wang, Yang Zhang, Jie Guo, and Yanwen Guo. 2020. FA-GANs: Facial Attractiveness Enhancement with Generative Adversarial Networks on Frontal Faces. *arXiv preprint arXiv:2005.08168*.

[30] Karras, T., Laine, S., and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4401–4410.

[31] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8110–8119.

[32] Kingma, D. P. and Welling, M. 2013. Auto-Encoding Variational Bayes. *arXiv preprint arXiv:1312.6114*.

[33] Kingma, D. P. and Welling, M. 2019. An Introduction to Variational Autoencoders. *arXiv preprint arXiv:1906.02691*.

[34] Lazer, D. M. J., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild, D., and others. 2018. The Science Of Fake News. *Science* 359, 6380, 1094–1096.

[35] Liu, J., Li, S., Song, W., Liu, L., Qin, H., and Hao, A. 2018. Automatic Beautification for Group-Photo Facial Expressions using Novel Bayesian GANs. In *International Conference on Artificial Neural Networks*, 760–770.

[36] McCloskey, S. and Albright, M. 2018. Detecting GAN-Generated Imagery Using Color Cues. *arXiv preprint arXiv:1812.08247*.

[37] Mescheder, L., Geiger, A., and Nowozin, S. 2018. Which Training Methods for GANs do Actually Converge? *arXiv preprint arXiv:1801.04406*.

[38] Metz, L., Poole, B., Pfau, D., and Sohl-Dickstein, J. 2016. Unrolled generative adversarial networks. *arXiv preprint arXiv:1611.02163*.

[39] Mirza, M. and Osindero, S. 2014. Conditional Generative Adversarial Nets. *arXiv preprint arXiv:1411.1784*.

[40] Nataraj, L., Mohammed, T. M., Manjunath, B. S., Chandrasekaran, S., Flenner, A., Bappy, J. H., and Roy-Chowdhury, A. K. 2019. Detecting GAN Generated Fake Images Using Co-Occurrence Matrices. *Electronic Imaging* 2019, 5, 532-1.

[41] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C., and Iyengar, S. S. 2018. A Survey On Deep Learning: Algorithms, Techniques, And Applications. *ACM Computing Surveys (CSUR)* 51, 5, 1–36.

[42] Pumarola, A., Agudo, A., Martinez, A. M., Sanfeliu, A., and Moreno-Noguer, F. 2018. GANimation: Anatomically-aware Facial Animation from a Single Image.

In *Proceedings of the European conference on computer vision (ECCV)*, 818–833.

[43] Radford, A., Metz, L., and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

[44] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. 2016. Improved Techniques for Training GANs. In *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett, Eds. Curran Associates, Inc, 2234–2242.

[45] Shen, Y., Gu, J., Tang, X., and Zhou, B. 2020. Interpreting the Latent Space of GANs for Semantic Face Editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9243–9252.

[46] Song, J., Zhang, J., Gao, L., Liu, X., and Shen, H. T. 2018. Dual Conditional GANs for Face Aging and Rejuvenation. In *IJCAI*, 899–905.

[47] Talreja, V., Taherkhani, F., Valenti, M. C., and Nasrabadi, N. M. 2019. Attribute-Guided Coupled GAN for Cross-Resolution Face Recognition. *arXiv preprint arXiv:1908.01790*.

[48] Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., and Ortega-Garcia, J. 2020. Deepfakes And Beyond: A Survey Of Face Manipulation And Fake Detection. *arXiv preprint arXiv:2001.00179*.

[49] Tran, L., Yin, X., and Liu, X. 2017. Disentangled Representation Learning GAN for Pose-Invariant Face Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1415–1424.

[50] Tripathy, S., Kannala, J., and Rahtu, E. 2020. ICface: Interpretable and Controllable Face Reenactment Using GANs. In *The IEEE Winter Conference on Applications of Computer Vision*, 3385–3394.

[51] Vincent, J. 2019. *A never-ending stream of AI art goes up for auction*. https://www.theverge.com/2019/3/5/18251267/ai-art-gans-mario-klingemann-auction-sothebys-technology. Accessed 9 October 2020.

[52] Wang, R., Ma, L., Juefei-Xu, F., Xie, X., Wang, J., and Liu, Y. 2019. FakeSpotter: A Simple Baseline For Spotting AI-Synthesized Fake Faces. *arXiv preprint arXiv:1909.06122*.

[53] Yang, C., Ding, L., Chen, Y., and Li, H. 2020. Defending Against GAN-Based Deepfake Attacks Via Transformation-Aware Adversarial Faces. *arXiv preprint arXiv:2006.07421*.

[54] Yi, R., Liu, Y.-J., Lai, Y.-K., and Rosin, P. L. 2019. APDrawingGAN: Generating Artistic Portrait Drawings From Face Photos With Hierarchical GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[55] Yu, N., Davis, L. S., and Fritz, M. 2019. Attributing Fake Images To GANs: Learning And Analyzing GAN Fingerprints. In *Proceedings of the IEEE International Conference on Computer Vision*, 7556–7566.

[56] Zhang, X., Karaman, S., and Chang, S.-F. 2019. Detecting and Simulating Artifacts in GAN Fake Images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, 1–6.

[57] Zhao, L., Mo, Q., Lin, S., Wang, Z., Zuo, Z., Chen, H., Xing, W., and Lu, D. 2020. UCTGAN: Diverse Image Inpainting Based on Unsupervised Cross-Space Translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5741–5750.

Anomalie Detektion in Bildtrainingsdaten

Jörn Hoffarth
Hochschule Reutlingen
Joern.Hoffarth@Student.Reutlingen-University.de

Abstract

Machine Learning ist in der Bildverarbeitung nicht mehr wegzudenken. Das Feature Engineering tritt durch Deep Learning Methoden in den Hintergrund, während die Datenaufbereitung an Bedeutung gewinnt. Anomalie Detektion kann bei dieser durch das Erzeugen von ausbalancierten Trainingsdaten helfen. Mit der vorliegenden Arbeit soll ein geeigneter Algorithmus gefunden werden, welcher der Hochschule Reutlingen im Motion Capture Labor hilft, ausbalanaciertere Datensätze zu erstellen. Nach dem Beurteilen verschiedener Algorithmen ist das GANomaly Framework [1] als geeigneter Kandidat ausgewählt und auf dem UCSD Datensatz [2] getestet worden. Mit einem erreichten ROC-Score von 77,8 % liegt GANomaly im Mittelfeld. Wahrscheinlich spielt das fehlende Lernen des zeitlichen Zusammenhangs eine bedeutende Rolle.

CCS Concepts

• **Computing methodologies** → *Scene anomaly detection*;

Keywords

Anomaly Detection, Novelty Detection, Cornercases, Deep Learning, KI Training

Betreuer Hochschule:	Prof. Dr.-Ing. Cristóbal Curio Hochschule Reutlingen Cristobal.Curio@Reutlingen-University.de
Betreuer Hochschule:	M.Sc. Michael Essich Hochschule Reutlingen Michael.Essich@Reutlingen-University.de

Informatics Inside Herbst 2020
25. November 2020, Hochschule Reutlingen

1 Einleitung

Für den Erfolg vieler Maschine Learning Methoden ist eine ausbalancierte Auswahl der Trainingsdaten wichtig. Gerade Grenzfälle werden aufgrund des seltenen Auftretens nicht gelernt. Könnten Grenzfälle in den Trainingsdaten gefunden werden, wäre es möglich, während der Datenaufbereitung diese stärker zu Gewichten oder weitere ähnliche Fälle zu generieren. Andererseits könnten diese auch absichtlich aussortiert werden, um das Lernen des Standardfalls zu stärken.
Solche Grenzfälle zu beschreiben, fällt schwer, denn eigentlich sind sie nach dem Abzug aller Standardfälle alles, was übrig bleibt. In einem Merkmalsvektorraum wären diese Fälle Außreißerpunkte. Domänenübergreifend wird von einer Anomalie und der Aufgabenstellung der Anomalie Detektion gesprochen. Allgemein lässt sich die Aufgabenstellung an die Algorithmen formulieren als das automatisierte Finden eines Musters in Daten, welches nicht zum erwarteten Verhalten passt. [3] Neben dem vorgeschlagenen Erzeugen von ausbalancierten Trainingsdaten, sind typische Anwendungsfälle einer Anomalie Detektion das Erkennen von Betrug im Finanzwesen (Kreditkartenbetrug, Versicherungsbetrug etc.) oder die Störungserkennung in der Informationstechnik. [3]
Im Rahmen dieser Arbeit sollen Ansätze verschiedener bildverarbeitender Anomalie Detektoren hinsichtlich ihrer Fähigkeit, der Erzeugung von ausbalancierten Trainingsdaten hilfreich zu sein, verglichen werden. Eine Auswahl soll nach Möglichkeit auf dem UCSD Anomaly Detection Datensatz [2] erprobt werden.

Der UCSD Datensatz bietet sich an, da er in vielen anderen Arbeiten [2], [4]–[8] als Benchmark verwendet wird und im Vergleich zu Datensätzen wie MINST oder CIFAR die benötigte Komplexität hat, um eine belastbare Vorhersage für die Eignung auf den im Motion Capture Labor der Hochschule Reutlingen erhobenen Bilddaten treffen zu können. In dem Motion Capture Labor werden Markierungen auf Objekten drei dimensional verfolgt. Neben ihrer Position im Motion Capture Volumen können zwischen Markierungen Winkel berechnen werden.
Mit dieser Information können beispielsweise Ort und Orientierung menschlicher Gliedmaßen verfolgt und damit die eingenommene Pose bestimmt werden. Wird nun noch eine ortsfeste klassische Kamera installiert, sollen mithilfe der Bilddaten durch den Anomalie Detektor indirekt beurteilt werden wie ausbalanciert die im Motion Capture Labor aufgenommenen Daten sind.

2 Vorangegangene Arbeiten

Aufgrund der verwendeten Deep Learning Algorithmen in der Videoanalyse bietet die Arbeit *An Overview of Deep Learning Based Methods for Unsupervised and Semi-Supervised Anomaly Detection in Videos* [9] einen guten Ansatzpunkt für die Recherche. Die häufig zitierte Veröffentlichung *Anomaly detection: A survey* [3] bietet einen großen Überblick mit hilfreichen Metriken zur Einordnung und Bewertung der Algorithmen. Allerdings stammt diese Veröffentlichung aus dem Jahr 2009 und erschien damit vor der in der Videoanalyse wichtigen Erfindung der Generative Adversarial Networks (GAN) [10]. Viele der bis dahin bekannten Algorithmen arbeiten auf einer statistischen Analyse von Merkmalsvektorräumen. So könnten beispielsweise im Motion Capture Labor vorverarbeitete Gelenkwinkel als Merkmalsvektoren übergeben und statistisch auffällige Gelenkstellungen erkannt werden. Ziel dieser Arbeit soll dagegen ein bildverarbeitender Anomalie Detektor sein ohne händische Merkmalsentwicklung. Möglich wären zum Beispiel folgende Ansätze:

GAN Das verwendete Generative Adversarial Network von Ravanbakhsh et al. [4] basiert auf dem Image-to-image GAN von Isola et al. [11]. Es detektiert Anomalien, indem es aus dem gegebenen Bild ein Bild des optischen Flusses vorhersagt und aus dem realen gegebenen optischen Fluss wiederum das Ausgangsbild generiert. Anschließend werden beide generierten Bilder mit den realen verglichen. Je stärker die Rekonstruktionsfehler, desto wahrscheinlicher die Anomalie.

AE Für den Autoencoder von Sabokrou et al. [5] werden Videobilder dreidimensional gestapelt. Anschließend werden Quader mit 10x10x5 dem Autoencoder übergeben und ein Anomaliescore berechnet. Ist dieser außerhalb eines Schwellenwerts, werden um diesen Würfel weitere 30x30x10 große Bilder an einen weiteren Autoencoder übergeben und der Rekonstruktionsfehler berechnet. Ist dieser erneut außerhalb eines Schwellenwerts, wird die Position als Anomalie gekennzeichnet.

LSTM Die von Jefferson et al. vorgestellte Long Short-Term Memory Network [6] komprimiert gegebene Videosequenzen in eine interne Repräsentation. Anschließend werden diese von zwei Decoder Netzwerken genutzt, um die bereits vergangenen Bilder und die zukünftigen Bilder vorherzusagen. Auch hier wird der Rekonstruktionsfehler als Anomaliescore verwendet.

Alle drei Ansätze wurden von den Autoren sehr erfolgreich auf dem UCSD-Datensatz getestet. Obwohl sich die Umsetzung sehr unterscheidet, ist die Grundidee sehr ähnlich. In allen Fällen werden die Bilder von einem Encoder in einen latenten Merkmalsvektorraum reduziert und anschließend wieder rekonstruiert. Dadurch, dass sich nicht gelernte

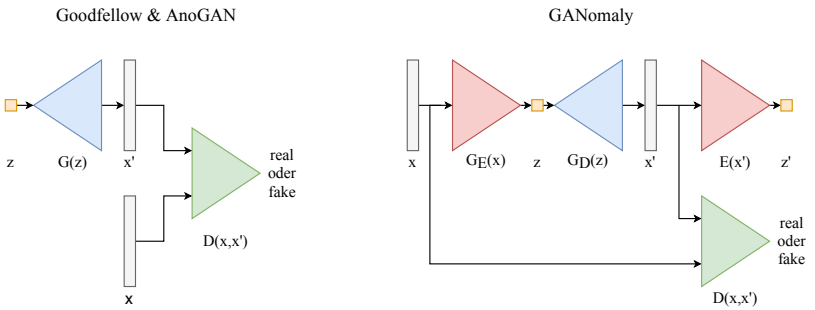

Abbildung 1: Links der Aufbau des von Goodfellow et al. entwickelten klassischen GAN. AnoGAN besitzt ebenfalls den klassischen Aufbau. Rechts der Aufbau des GANomaly Frameworks.

Muster nicht rekonstruieren lassen, entsteht bei Anomalien ein Rekonstruktionsfehler. Im Weiteren wird ein Ansatz basierend auf einer Kombination aus einem Autoencoder und einem GAN verfolgt. Eine zusätzliche Kombination mit LSTM-Layern wäre denkbar.

3 Generative Adversarial Network

Das von Goodfellow et al. [10] vorgestellte GAN ist eine unüberwachte Lernmethode, welche in ihrer ursprünglichen Form aus zwei Multilayer Perceptrons Netzwerken besteht. Der Generator erzeugt aus einem zufälligen Vektor z ein künstlich generiertes Bild[1]. Mit diesem künstlich erzeugten Bild versucht er das zweite Netz, den Diskriminator, zu täuschen. Der Diskriminator bekommt neben den künstlichen die realen Bilder und wird darauf trainiert, diese voneinander zu unterscheiden. Das Lernen erfolgt nun in zwei Schritten. Zunächst werden Bilder vom Generator erzeugt und diese zum Training des Diskriminator verwendet. Im zweiten Schritt werden die Gewichte des Diskriminators eingefroren und es wird zurückgerechnet, wie die Gewichte des Generators aussehen müssten, um den Diskriminator besser täuschen zu können. Anschließend werden wieder neue Bilder zum Trainieren des Diskriminator erzeugt. Ist das Training abgeschlossen, können durch die Wahl eines z Vektors z Kombinationen an Bildern erzeugt werden.

Dadurch, dass der Generator nicht direkt auf den gegebenen Bildern gelernt wird, sondern nur den Diskriminator täuschen muss, passen die entstandenen Bilder zwar in die Merkmalsverteilung der Trainingsbilder, sind aber höchstens zufällig einem Bild ähnlich. Neben der Hilfe beim Lernen der Merkmalsverteilung verhindert der Diskriminator offensichtliche Anzeichen eines generierten Bildes. Das führt zu fotorealistischen Bildern, die so realistisch sind, dass sie unter dem Sammelbegriff Deepfakes Schlagzeilen machen [12]. Neben der Angst vor dem Missbrauch, gibt es allerdings auch viele sinnvolle Anwendungsgebiete, wie zum Beispiel das automatische Umwandeln von Satellitenbildern in Kartenmaterial, das Umwandeln von Skizzen mit Stoffmustern zu fotorealistischen Darstellungen der skizzierter Kleidungsstücke [11] oder aber auch das Generieren von künstlichen Personenbildern zur freien Verfügung [13].

[1]Das Bild als häufige Anwendung steht hierbei exemplarisch für einen Tensor mit interpretierbaren Daten.

AnoGAN [14] zeigt umgekehrt, wie das Generieren der fotorealistischen Bilder auch zum Anomalie Finden geeignet ist. AnoGAN verwendet dabei denselben Aufbau wie Goodfellow et al. und ergänzt ihn um eine Anomaliescoreberechnung. [14] Nach dem abgeschlossenen Training auf normalen Daten werden die Gewichte sowohl des Generators als auch des Diskriminators eingefroren. Anschließend wird mit normalen und abnormalen Bildern getestet. Jedes Bild wird hierbei zunächst auf den ähnlichsten z Vektor wie im Trainingsfall zurückgerechnet und anschließend aus diesem z ein Bild erzeugt. Dieses neu generierte Bild wird nun dem Diskriminator zur Bewertung übergeben. Der Anomaliescore ergibt sich aus der gewichteten Summe der Abweichung zu dem ähnlichsten z Vektor und der Bewertung des Diskriminators.

4 GANomaly auf UCSD Datensatz

GANomaly von Akcay et al. [1] ist ein 2018 veröffentlichtes GAN basiertes Anomalie Detektion Framework. Es gehört zu den unüberwachten Lernmethoden, da es während dem Training nur normale Fälle gezeigt bekommt aber im Anschluss auch abnormale Fälle, die Anomalien erkennen muss. Im Gegensatz zu AnoGAN besitzt es zwei weitere Encoder Netze. Durch diese entfällt das Zurückrechnen auf den ähnlichsten z Vektor wie bei AnoGAN und durch das Zweite wird der Anomaliescore nicht aus der Qualität der Rekonstruktion auf Bildebene, sondern der Qualität der Rekonstruktion auf der zusammengefassten latenten Vektorebene z berechnet. GANomaly ist dafür folgendermaßen aufgebaut (vgl. Abbildung 1):
Das Eingangsbild wird von einem Encoder G_E auf den z Vektor reduziert. Mithilfe dieses Vektors wird eine möglichst ähnliche Kopie x' des Ursprungsbilds x generiert. Anschließend wird auch diese wieder durch einen gleichermaßen aufgebauten Encoder $E(G(x))$ zu einem neuen Vektor z' reduziert. Aus der Quadratsumme der Differenzen (L2-Norm) zwischen z und z' ergibt sich wie folgt der Anomaliescore:

$$\begin{aligned} A(x) &= \|z - z'\|_2 \\ &= \|G_E(x) - E(G(x))\|_2 \end{aligned} \tag{1}$$

Während des Trainings wird ein gewichteter Loss minimiert. Der Loss setzt sich aus der Summe über alle Grauwertdifferenzen (L1-Norm) des Eingangs- und Generiertenbildes x und x', dem Abstand der beiden Vektoren z und z' und der Ausgabe des Diskriminators zusammen.

$$L = \omega_1 L_{adv} + \omega_2 L_{con} + \omega_3 L_{enc}$$

$$\begin{aligned} L_{adv} &= D(x, x') \\ &= D(x, G(x)) \end{aligned}$$

$$\begin{aligned} L_{con} &= \|x - x'\|_1 \\ &= \|x - G(x)\|_1 \end{aligned} \tag{2}$$

$$\begin{aligned} L_{enc} &= \|z - z'\|_2 \\ &= \|G_E(x) - E(G(x))\|_2 \end{aligned}$$

Interessanterweise empfehlen die Autoren Akcay et al. die Grauwertsdifferenzen 50x und die beiden anderen einfach zu gewichten. Dabei tritt das eigentliche GAN Prinzip, das Täuschen des Diskriminators während des Trainings, in den Hintergrund. Mehr Gewicht bekommt das pixelweise Rekonstruieren der Bilder, ähnlich einem klassischen Autoencoder.

4.1 Datenaufbereitung

Der UCSD Datensatz enthält Videos ortsfester installierter Kameras, die aus zwei Ansichten Fußgängerzonen filmen. Anomalien sind hierbei Fahrradfahrer, Tretrollerfahrer, Skateboardfahrer, Rollstuhlfahrer aber auch eine Person, die mitten auf der Fußgängerzone stehen bleibt oder ein Transporter.

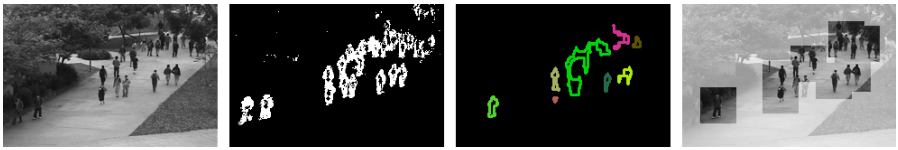

Abbildung 2: Vorverarbeitungsschritte für Variante 2 von links nach rechts. Eingangsbild; Hintergrundabzug, Clustering mit Anschließender Schwerpunktberechnung der Kontur, 40x40pixel Bildzuschnitte als neue Trainingsdaten

Grob ist der Datensatz in die zwei unterschiedliche Kameraeinstellungen, „Ped1“ und „Ped2“, aufgeteilt. Aufgrund der ausreichenden Menge an Bilddaten in „Ped1“ wurde nur dieser Teil des Datensatzes verwendet. Die darin enthaltenen Videosequenzen sind sortiert in kein abnormales Verhalten und mit abnormalem Verhalten. Zu den Videosequenzen mit abnormalem Verhalten gibt es zu allen Frame basierte Informationen, ob abnormales Verhalten vorhanden ist. Zusätzlich gibt es für einige Sequenzen auch Masken, in denen abnormales Verhalten Pixelweise markiert ist. Zur Weiterverarbeitung wurden drei Viertel der normalen Bilder als Trainingsdaten, ein Viertel der normalen Bilder als normale Testdaten und alle Sequenzen mit mehr als 10 abnormal markierten Pixeln als abnormale Testdaten verwendet.

Variante 1 GANomaly wurde direkt auf allen Bilder von „Ped1“ ohne weitere Vorbereitung ausgeführt.

Variante 2 Um den Mode Colapse zu vermeiden, bei dem der Generator den Diskriminator mit dem ständigen Erzeugen einer leeren Fußgängerzone täuscht, wurden für den zweiten Versuch bewegte Objekte ausgeschnitten. Hierfür werden durch einen Hintergrundabzug zweier aufeinanderfolgenden Bildern alle beweglichen Stellen in einer Maske markiert. Anschließend wird von den markierten Stellen der Schwerpunkt berechnet und dieser als Mittelpunkt eines 40x40 Pixel großen Bildausschnitts verwendet. Der Bildausschnitt zeigt nun im Normalfall Fußgänger mal einzeln, mal in Gruppen, aber vor allem vor wechselndem Hintergrund.
Für die abnormalen Testfälle wurden über die Ground Truth Maske ebenfalls 40x40 Pixel große abnormale Bilder erzeugt. Abbildung 2 zeigt die Verarbeitungsschritte mit exemplarischen Ergebnisbildern, welche zum Training verwendet werden.

4.2 Ergebnis

Nach 30 Epochen Trainingszeit erreicht Variante 1 eine Fläche unter der Grenzwertoptimierungskurve (ROC) von 77,8 % (vgl. Abbildung 4 und liegt damit im Mittelfeld. [4] Wenn es ausreichen würde nur 20 % der Anomalien zu finden, könnte der Schwellenwert so gelegt werden, dass so gut wie jeder Treffer eine Anomalie ist. Soll die Trefferquote erhöht werden, nehmen die fehlerhaft als Anomalie klassifizierten falsch Positiven Ergebnisse zu und die Genauigkeit des Klassifikators nimmt ab. Auch an den Histogrammen lässt sich dieses Verhalten gut ablesen. Das mittlere Diagramm aus Abbildung 3 zeigt den berechneten Anomaliescore, aufgeteilt mithilfe der Ground Truth in normale Fälle und Anomalien. An den Stellen, an denen sich die Verteilungen überlappen, werden zwangsweise falsch positive Fälle mitaufgenommen. Aufgefallen ist, dass alle vom Generator generierten Bilder nur noch den Hintergrund zeigen, möglicherweise mit kleinen fürs Auge

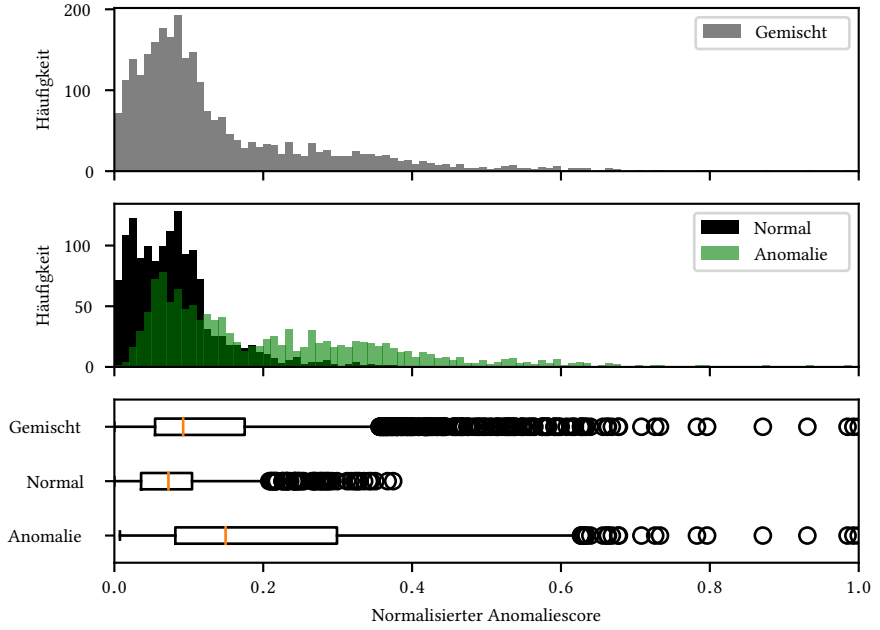

Abbildung 3: Oben Anomaliescore Verteilung aller Testbilder nach 30 Trainingsepochen; in der Mitte Verteilung der normalen Testbilder und der Anomalien nach 30 Epochen Training auf dem UCSD Datensatz (Variante 1); unten Boxplot mit Rechteck vom 1. Quartil bis zum 3. Quartil und orangenem Median.

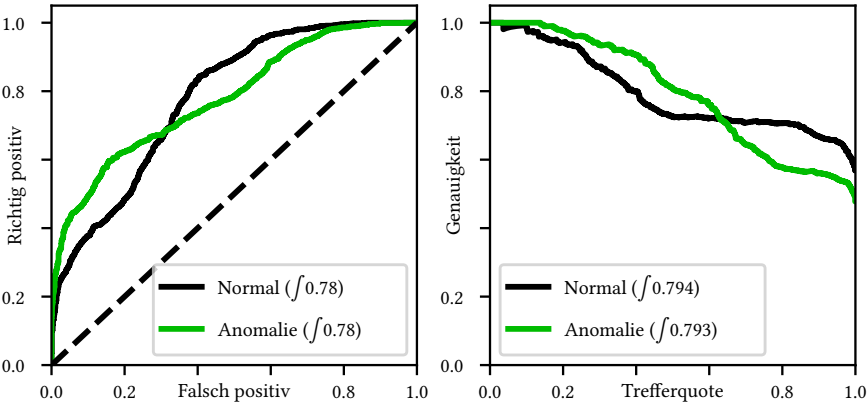

Abbildung 4: ROC und Precision-Recall Kurve nach 30 Epochen Training auf dem UCSD Datensatz (Variante 1). Die Fläche jeweils unter der Kurve als Kennzahl in der Legende.

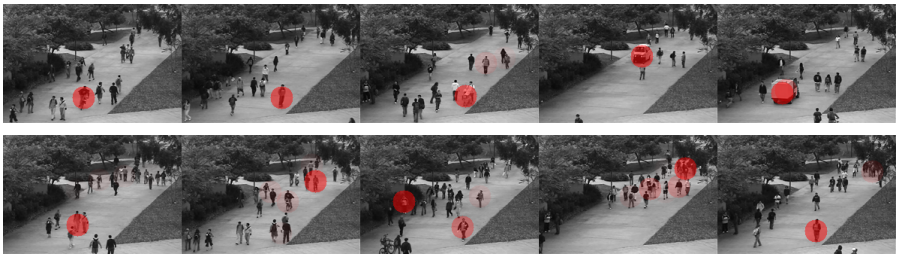

Abbildung 5: Ergebnisse aus Variante 2 nach 30 Epochen Training. Oben korrekt erkannte Anomalien, unten fälschlicherweise erkannte Anomalien.

nicht sichtbaren Nuancen, die für den überraschend guten ROC-Score verantwortlich sein könnten.
Um dem Mode Colapse vorzubeugen wurden bewegte Stellen im Bild ausgeschnitten und GANomaly zugeführt (Variante 2). Zwar konnte der Mode Colapse damit verhindert, aber mit 70,4 % das Gesamtergebnis nicht verbessert werden. Abbildung 5 zeigt in der oberen Reihe erfolgreiche Detektionen und in der unteren Reihe fälschlicherweise als Anomalien erkannte Beispiele. Eine Besonderheit der zweiten Variante ist, dass durch das Lernen und Bewerten von Bildausschnitten die Möglichkeit, wie in Abbildung 5 zu sehen besteht, die Anomalien räumlich zuzuordnen. Über die Intensität der roten Einfärbung wird der Anomaliescore veranschaulicht.

5 Diskussion

Mit einem ROC Score von 77,8 % entscheidet das trainierte GAN wesentlich besser als ein Zufallsgenerator. Nichtsdestotrotz sind die in Kapitel 2 vorgestellten Algorithmen besser.
Neben Ideen, die Leistung weiter zu steigern durch zum Beispiel mehr Trainingsdaten oder einen Sliding Window Ansatz, ist es sehr wahrscheinlich, dass viele Informationen durch das Weglassen des zeitlichen Zusammenhangs verloren gehen. Alle, der vorgestellten Ansätze binden diesen mit ein, sodass der Algorithmus Bewegungsmuster lernen kann. Lösungen hierfür wären die nicht benötigten Farbkanäle mit zeitlich aufeinanderfolgenden Bildern zu füllen oder das Generator Netzwerk mithilfe der Gedächtniszellen der Long Short-Term Memory Networks um eine Abhängigkeit zur Vergangenheit zu ergänzen.

Literatur

[1] S. Akcay, A. A. Abarghouei und T. P. Breckon, “GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training”, *CoRR*, Jg. abs/1805.06725, 2018. arXiv: 1805 . 06725. Adresse: http : //arxiv.org/abs/1805.06725.

[2] V. Mahadevan, W. Li, V. Bhalodia und N. Vasconcelos, “Anomaly detection in crowded scenes”, in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 2010, S. 1975–1981.

[3] V. Chandola, A. Banerjee und V. Kumar, “Anomaly detection: A survey”, *ACM computing surveys (CSUR)*, Jg. 41, Nr. 3, S. 1–58, 2009.

[4] M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. Regazzoni und N. Sebe, “Abnormal event detection in videos using generative adversarial nets”, in *2017 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2017, S. 1577–1581.

[5] M. Sabokrou, M. Fathy und M. Hoseini, "Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder", *Electronics Letters*, Jg. 52, Nr. 13, S. 1122–1124, 2016.

[6] J. R. Medel und A. Savakis, "Anomaly detection in video using predictive convolutional long short-term memory networks", *arXiv preprint arXiv:1612.00390*, 2016.

[7] H. Dhole, M. Sutaone und V. Vyas, "Anomaly Detection using Convolutional Spatiotemporal Autoencoder", in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2019, S. 1–5.

[8] J. Fu, W. Fan und N. Bouguila, "A Novel Approach for Anomaly Event Detection in Videos Based on Autoencoders and SE Networks", in *2018 9th International Symposium on Signal, Image, Video and Communications (ISIVC)*, 2018, S. 179–184.

[9] B. R. Kiran, D. M. Thomas und R. Parakkal, "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos", *Journal of Imaging*, Jg. 4, Nr. 2, S. 36, 2018.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville und Y. Bengio, "Generative adversarial nets", in *Advances in neural information processing systems*, 2014, S. 2672–2680.

[11] P. Isola, J.-Y. Zhu, T. Zhou und A. A. Efros, "Image-to-image translation with conditional adversarial networks", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, S. 1125–1134.

[12] T. T. Nguyen, C. M. Nguyen, D. T. Nguyen, D. T. Nguyen und S. Nahavandi, "Deep learning for deepfakes creation and detection", *arXiv preprint arXiv:1909.11573*, Jg. 1, 2019.

[13] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen und T. Aila, *Analyzing and Improving the Image Quality of StyleGAN*, 2020. arXiv: 1912.04958 [cs.CV].

[14] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth und G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery", in *International conference on information processing in medical imaging*, Springer, 2017, S. 146–157.

Autorenverzeichnis

Hochschule Reutlingen
Reutlingen University
Fakultät Informatik
Human-Centered Computing
Alteburgstraße 150
D-72762 Reutlingen

Telefon: +49 7121 / 271-4002
Telefax: +49 7121 / 271-4042

E-Mail: infoinside@reutlingen-university.de
Website: infoinside.reutlingen-university.de