

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
Department “Institut für Informatik”
Lehr- und Forschungseinheit Medieninformatik
Prof. Dr. Heinrich Hußmann

Masterarbeit

**Entwicklung eines Systems zur Nutzung von VR-Brillen im
Unterricht**

Veronika Fuchsberger
veronika.fuchsberger@campus.lmu.de

Bearbeitungszeitraum: 01. 08. 2018 bis 30. 01. 2018
Betreuer: Christoph Krichenbauer
Verantw. Hochschullehrer: Prof. Heinrich Hußmann

Zusammenfassung

In der folgenden Arbeit wurden aktuelle VR-Headsets verglichen und nach Preissegmenten und Anwendungsfunktionen eingeordnet. Zudem wurde bisher existierende Software, die Virtual Reality für den Schulunterricht zugänglich macht diskutiert und bewertet und auf die Challenges eingegangen, die Virtual Reality-Systeme generell und im Bezug auf die Nutzung im Schulunterricht momentan haben. Weiterhin wurde in der Arbeit auf die verschiedenen Inhalte für VR-Systeme eingegangen, welche Quellen es für diese Inhalte gibt und wie sie bearbeitet und selbst erstellt werden können.

Anschließend wurde basierend auf den zuvor erörterten Anforderungen an eine Software, die Lehrkräfte im Unterricht nutzen können, um 360°-Inhalte zu zeigen, VRClassroom entwickelt. Ein zwei-teiliges System, mit dem der Lehrer in einer App auf seinem Computer Inhalte hineinladen kann, die dann in einer Web-App auf VR-Brillen gezeigt werden. Die Lehrkräfte führen dabei das komplette VR-Erlebnis für die Schüler und können 360°-Fotos und -Videos hineinladen, Markierungen setzen und das Video synchronisiert für alle abspielen und pausieren. Außerdem können auch 3D-Modelle in die App geladen werden, die die Lehrkraft dann je nach Bedarf skalieren und drehen kann und an interessanten Stellen Markierungen setzen kann. Um zu sehen, welche Geräte verbunden sind, wird in der Lehrer-App eine Liste mit allen Geräten mit einem Aktivitätsindikator angezeigt.

Anschließend wurde eine Online-Befragung mit Lehrkräften durchgeführt, bei der die Teilnehmer ein Video gezeigt wurde, das die Nutzung von VRClassroom erklärt, zu dem sie nachfolgend einige Fragen beantworteten. Die Mehrheit der Teilnehmer war dem System gegenüber sehr aufgeschlossen und konnte sich gut vorstellen VRClassroom zukünftig im Unterricht zu verwenden, um 360°-Inhalte zu zeigen.

Abschließend wurde diskutiert wie Zukunft von Virtual Reality im Schulunterricht aussehen könnte und wie VRClassroom weiterentwickelt werden könnte, um die Bedürfnisse von Lehrkräfte noch besser zu erfüllen beziehungsweise welche weiteren Funktionen das System noch interessanter machen würden.

Abstract

Short abstract of the work, maximum of 250 words.

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

München, 8. Januar 2019

.....

Inhaltsverzeichnis

1 Einleitung	1
2 Existierende VR-Hardware-Systeme	3
2.1 Merkmale von VR-Systemen	3
2.1.1 Degrees of Freedom	3
2.1.2 Display	3
2.1.3 Frequenz	3
2.1.4 Field of View	3
2.1.5 Rechenleistung	3
2.1.6 Inputmethoden	3
2.1.7 Tracking	4
2.1.8 Gewicht	4
2.1.9 Preis	4
2.1.10 Liste Quellen etc	4
2.2 Computer-gestützte VR-Hardware	4
2.2.1 Oculus Rift	4
2.2.2 HTC Vive	4
2.2.3 HTC Vive Pro	4
2.2.4 Playstation VR	4
2.3 Stand-alone VR-Hardware	4
2.3.1 Oculus Go	6
2.3.2 Oculus Quest	6
2.3.3 Lenovo Mirage Solo	6
2.3.4 HTC Vive Focus	6
2.4 Smartphone gestützte VR-Hardware	6
2.4.1 Google Cardboard	6
2.4.2 Samsung Gear VR	6
2.4.3 Google Daydream View	6
3 Aktuelle Einsatzgebiete von VR	7
3.1 3D in VR	7
3.2 Eingabemethoden in VR	7
3.3 VR-Systeme im Bildungsbereich	7
4 Aktuelle Challenges an VR-Systemen	9
4.1 Virtual Reality Sickness	9
4.2 Mindestalter zur Nutzung von VR-Systemen	9
5 360°-Inhalte	11
5.1 360°-Fotos	11
5.1.1 Projektionen von 360°-Fotos	11
5.1.2 Quellen für 360°-Fotos	11
5.1.3 Erstellung und Bearbeitung von 360°-Fotos	11
5.2 360°-Videos	11
5.2.1 Quellen für 360°-Videos	11
5.2.2 Erstellung und Bearbeitung von 360°-Videos	11
5.3 3D-Modelle	11
5.3.1 Quellen für 3D-Modellen	11
5.3.2 Erstellung und Bearbeitung von 3D-Modellen	11
5.4 360° Sound	11

5.4.1	Quellen für 360°-Sounds	11
5.4.2	Erstellung und Bearbeitung von 360°-Sounds	11
5.5	Virtual Reality Spiele und Anwendungen	11
5.5.1	Quellen für Games	11
5.5.2	Erstellung und Bearbeitung von Spielen und Anwendungen	12
5.6	360°-Inhalte in 3D	12
5.7	360°-Inhalte in React360	12
6	Software Entwicklung von VR-Systemen	13
6.1	Native VR-Applikationen	13
6.2	WebVR + Mobile VR: VR im Browser	13
6.2.1	A-Frame	13
6.2.2	React360	13
6.2.3	Three.js	13
7	Software Projekt: VRClassroom	15
7.1	Nutzungsszenario und Anwendungsfokus	15
7.2	Anwender	15
7.3	Grundstruktur	16
7.4	Lehrer-Applikation	16
7.4.1	Verbundene Geräte	16
7.4.2	Hineinladen von Inhalten	17
7.4.3	Controls	17
7.5	Kommunikation zwischen Lehrer-App und Schüler-App	19
7.6	QR-Code Fenster	19
7.7	Schüler-App	19
7.7.1	Eingabe des Namens	19
7.7.2	Anzeigen von 360°-Fotos	21
7.7.3	Abspielen von 360°-Videos	21
7.7.4	Abspielen von Ton in Videos	21
7.7.5	Anzeigen von 3D-Modellen	22
7.7.6	WebVR Polyfill	22
8	Nutzerstudie und Evaluation	23
8.1	Ursprünglich geplanter Ablauf der Studie	23
8.1.1	Freigabe Kulturministerium	23
8.2	Qualitative Nutzerstudie	23
8.2.1	Fragebogen	23
8.2.2	Erkenntnisse	23
8.3	Online-Umfrage	23
9	Ausblick	25
9.1	Zukunft von VR	25
9.2	Probleme von VRClassroom	25
9.2.1	Anzeigen großer Dateien	25
9.2.2	Synchronisierung	25
9.3	Mögliche Weiterentwicklungen an VRClassroom	25
9.3.1	Hosten von VRClassroom Online statt Verbindung mit Lehrer-Rechner	25
9.3.2	Skalierung von 3D-Modellen auf echte Größe	26
9.3.3	Navigation von 3D-Modellen aus Schüler-App	27
9.3.4	Ausfragemodus: Ausgewählter Schüler setzt Markierung	27

1 Einleitung

2 Existierende VR-Hardware-Systeme

Es gibt inzwischen eine große Zahl verschiedener VR-Hardware-Systeme, die sich in drei Gruppen mit unterschiedlichen Anwendungsszenarien unterteilen: Die Computer-gestützten VR-Systeme, die Stand-alone VR-Systeme und die Smartphone-gestützten VR-Systeme.

2.1 Merkmale von VR-Systemen

Um die verschiedenen VR-Hardware-Systeme einordnen zu können, gibt es einige Merkmale, auf die ein Käufer achten sollte. Denn die verschiedenen Headsets sind für unterschiedliche Anwendungsszenarien entwickelt worden, sodass ein potenzieller Käufer zuerst entscheiden sollte wie er das VR-Headset einsetzen möchte.

Diese Merkmale können dazu herangezogen werden: Degrees of Freedom (DoF), die Displaygröße und -auflösung, die Frequenz des Displays, Field of View, Rechenleistung und Gewicht des Headsets, die verwendete Tracking-Methode, die mitgelieferten Controller beziehungsweise mögliche Input-Methoden, das Gewicht des Headsets und der Verkaufspreis.

2.1.1 Degrees of Freedom

3DoF/ 6DoF Was bedeutet DoF?

<https://www.roadtovr.com/introduction-positional-tracking-degrees-freedom-dof/> <https://www.cs.cmu.edu/rapidproto/mechanisms/chpt4.html>

2.1.2 Display

Maße, Auflösung, 2 einzelne/ ein großes Vielleicht Display und Frequenz zu einem Punkt verbinden?

2.1.3 Frequenz

2.1.4 Field of View

2.1.5 Rechenleistung

Welche Chips: CPU & GPU ?

2.1.6 Inputmethoden

Wie in Kapitel XXX (hier Ref einfügen) beschrieben, gibt es drei grundsätzliche Input-Methoden für VR-Headsets: Input über Controller mit verschiedenen Buttons und einem “Raycaster” in der VR-Welt, das Auslösen von Touch-Events auf dem Display des VR-Headsets und Voice-Input.

Da nur solche Headsets Touch-Events auf dem Display registrieren, die ein Smartphone als Display nutzen, sind es ebenso nur diese, die Input durch Touch-Interaktionen mit dem Display als Input erlauben. Das sind vor Allem Google Cardboards, die eine Art “Arm” nutzen um auf das Display zu drücken oder einige Nachbauten des ursprünglichen Google Cardboards, die Magneten verwenden um ein Touch-Event auf dem Display auszulösen. (Diese Technik noch genauer erläutern?) Bei beiden Möglichkeiten ist es allerdings nur möglich an einer bestimmten Stelle ein Touch-Event auszulösen, sie müssen also in Verbindung mit einem Art Gaze-Punkt verbunden werden, um dem Nutzer sinnvollen Input zu erlauben.

Gaze Input? (-> EyeVR)

Bisher gibt es keine Geräte, die mit Voice-Input kontrolliert werden können. Allerdings wäre das sehr wünschenswert, da damit die Nutzung von VR-Systemen auch für Menschen mit körperlichen Behinderungen möglich wird.

2.1.7 Tracking

Trackingmethode: kein tracking/ inside out/ Sensoren im Raum

2.1.8 Gewicht

2.1.9 Preis

Die Preise sind in Dollar, wie sie auf dem amerikanischen Markt zu kaufen sind. Darin sind keine Umsatzsteuern enthalten.

2.1.10 Liste Quellen etc

Augenabstand: Einstellung der Pupillendistanz und des Objektivabstands
 Sensoren: SteamVR Tracking, G-Sensor, Gyroskop, Nähesensor
 Controller: Multifunktions-Trackpad, Greifknöpfe, zweistufiger Abzug, Systemknopf, Menütaste Outside Tracking, Kamera in Headset

Oculus Rift: <https://www.vrbound.com/headsets/oculus/rift> Preis: Oculus.com/rift

Playstation VR: <https://www.vrbound.com/headsets/sony/playstation-vr>,
<https://www.playstation.com/de-de/explore/playstation-vr/tech-specs/>

HTC Vive Focus: <https://www.vive.com/cn/product/vive-focus-en/>

Oculus Go: <https://www.vrbound.com/headsets/oculus/go> Sensoren: Orientational Tracking has no IPD adjustment

Oculus Quest: <https://uploadvr.com/oculus-quest-specs-price-release-date/> has an IPD adjustment
 Oculus says Quest's weight isn't locked in just yet, but presently it's about 100 grams heavier than the Rift, which would put it around 570 grams

Mirage Solo: <https://www.lenovo.com/us/en/virtual-reality-and-smart-devices/virtual-and-augmented-reality/lenovo-mirage-solo/Mirage-Solo/p/ZZIRZRHVR01> Preis 400\$ General Usage: 2.5 hours Dimensions: 204 mm x 269.5 mm x 179.86 mm Sensoren: P-Sensor, Gyroscope, Accelerometer, Magnetometer
 Controller: 3DoF Daydream Motion Controller

Vergleich: <https://www.vrnerds.de/vr-brillen-vergleich/> <https://uploadmagazin.de/blog/25071-standalone-vr-headsets/>

2.2 Computer-gestützte VR-Hardware

Die erste Gruppe sind die Computer-gestützten Hardware-Systeme: Sie sind mit einem Kabel mit dem Rechner verbunden, der die Rechenleistung für die eigentlich Brille übernimmt. Sie eignen sich besonders für extrem rechenaufwändige Anwendungen, aber schränken durch ihre Kabel die Bewegungsfreiheit des Nutzers ein.

2.2.1 Oculus Rift

2.2.2 HTC Vive

2.2.3 HTC Vive Pro

2.2.4 Playstation VR

2.3 Stand-alone VR-Hardware

Die zweite Gruppe sind die Stand-alone Hardware-Systeme wie etwa die Oculus Go, Oculus Quest oder Google Daydream. Es handelt sich hierbei um vollumfängliche Systeme, die ohne weiteres Equipment auskommen und auch keinen Computer benötigen. Da sie mit einem Akku betrieben werden, ist die Betriebsdauer allerdings eingeschränkt und auch die Rechenleistung ist deutlich geringer als die der Computer-gestützten Systeme.

VR-Headset	HTC Vive	Oculus Rift	Playstation VR	HTC Vive Focus	Oculus Go
DoF	3Dof?	gut	hoch	weit	3DoF
Display	1080 x 1200 Pixel pro Auge (2160 x 1200 Pixel zusammen), Dual AMOLED 3,6?? diagonal	2160x1200, 3.5in, OLED	5.7in, OLED, 1920x1080	weit	2560x1440
Frequenz	90 Hz	90 hz	120 hz	weit	60-72Hz
Field of View	110°	100 °	100 °	weit	100°
Inputmethoden	3Dof?	Oculus Touch, Xbox One Controller	Accelerometer, Gyroscope, Magnetometer	weit	Oculus Go Controller
Tracking	3Dof?	Accelerometer, Gyroscope, Magnetometer	Accelerometer, Gyroscope		orientational
Gewicht	h	380-470g	610g	weit	468g
Preis	599 Euro	449 Euro	hoch	weit	199 Euro
VR-Headset	Oculus Quest	Lenovo Mirage Solo	Google Cardboard	Samsung Gear VR	Google Daydream
DoF	6 DoF	6DoF	hoch	weit	sechs
Display	1600 x 1440 per eye, OLED	gut	4 bis 6 Zoll, hängt von genutztem Smartphone ab	hängt von genutztem Smartphone ab	hängt von genutztem Smartphone ab
Frequenz	72Hz	gut	hängt von genutztem Smartphone ab	hängt von genutztem Smartphone ab	hängt von genutztem Smartphone ab
Field of View	3Dof?	gut	abhängig von Smartphone ab	101°	abhängig von Smartphone ab
Inputmethoden	3Dof?	gut	Touch-Taste an Cardboard	Gear VR Controller	Daydream Controller
Tracking	3Dof?	gut	abhängig von Smartphone	Beschleunigungsbeschleunigungssensor, Lagesensor, Annäherungssensor + abhängig von Smartphone	abhängig von Smartphone
Gewicht	3Dof?	645g	96 g	345g	261 g
Preis	399 Dollar	400 Dollar	20 Euro	weit	hoch

Tabelle 2.1: Übersicht der Merkmale der verschiedenen VR-Headsets

2.3.1 Oculus Go

2.3.2 Oculus Quest

2.3.3 Lenovo Mirage Solo

2.3.4 HTC Vive Focus

2.4 Smartphone gestützte VR-Hardware

Die dritte Gruppe bilden die Smartphone-gestützten VR-Systeme. Damit sind alle Systeme gemeint, bei denen das Smartphone integriert wird, um die VR-Inhalte zu zeigen. Diese liegen preislich auf einem sehr niedrigen bis mittlerem Niveau und bilden so eine gute Möglichkeit für alle Leute in die Welt der VR-Systeme einzutauchen, ohne direkt mehrere Hundert Euro ausgeben zu müssen.

2.4.1 Google Cardboard

2.4.2 Samsung Gear VR

2.4.3 Google Daydream View

3 AKTUELLE EINSATZGEBIETE VON VR

3 Aktuelle Einsatzgebiete von VR

medizin Gaming Forschung verschiedene Therapien zB Angsttherapien

3.1 3D in VR

WebVR macht das schon. Dadurch muss die Szene doppelt gerendert werden. Wie entsteht 3d generell? Augenabstand

3.2 Eingabemethoden in VR

Controller Voice Control

3.3 VR-Systeme im Bildungsbereich

Google Expeditions Hier kommen noch mehr Sachen aus den Papern rein!

4 Aktuelle Challenges an VR-Systemen

5 Major Challenges in VR: <https://channels.theinnovationenterprise.com/articles/5-major-challenges-of-vr-industry>

Hier wird gerade besonders daran geforscht oder das gibt es noch nicht, ist aber sehr gewünscht.

4.1 Virtual Reality Sickness

<https://vrodo.de/virtual-reality-laut-umfrage-haben-60-prozent-probleme-mit-vr-uebelkeit/> Umfrage auf reddit: <https://www.strawpoll.me/12015620/r> Unterschied zu motion sickness Symptome Lösungen?

<https://www.ncbi.nlm.nih.gov/pubmed/6847562> <https://www.twentymilliseconds.com/post/all-about-motion-sickness/> <http://fortune.com/2018/02/06/virtual-reality-motion-sickness/>

<https://www.iflscience.com/technology/turns-out-answer-virtual-reality-sickness-right-front-your-face/>

4.2 Mindestalter zur Nutzung von VR-Systemen

Fast alle Hersteller geben in ihren Nutzungsbedingungen oder Sicherheitsanweisungen ein Mindestalter für die Benutzung ihrer VR-Systeme an. Wie in ?? zu sehen, sind sich die Hersteller sehr einig, dass VR-Headsets nicht für kleine Kinder geeignet sind und frühestens für Jugendliche in Frage kommen.

Oculus weist konkret darauf hin, dass eine Nutzung ihrer Geräte unter 13 Jahren ihren Nutzungsbedingungen widerspricht und diese erst für diese Altersgruppe entwickelt sind. “The Services are intended solely for users who are aged 13 or older. Any registration for, or use of, the Services by anyone under the age of 13 is unauthorised, unlicensed and in breach of these Terms.” Es werden allerdings keine genauen Gründe für diese Altersrestriktion angegeben. [1]

Samsung geht dabei noch einen Schritt weiter und warnt vor einer Nutzung unter 13 Jahren, da sich jüngere Kinder in einer “critical period in visual development” [2] befinden. Zudem sollen auch Kinder über 13 Jahren nur unter Aufsicht einer erwachsenen Person die Gear VR benutzen und dabei darauf achten regelmäßig Pausen zu machen. Eine lange Nutzung soll generell vermieden werden und die Kinder sollen während und nach der Nutzung beobachtet werden, ob sich ihre Fähigkeiten in der Hand-Augen-Koordination, Balance oder Multi-Tasking verschlechtern.

Außerdem wird eine Liste an Symptomen aufgeführt bei deren Anzeichen eine Nutzung sofort unterbrochen werden soll. Das sind: “seizures, loss of awareness, eye strain, eye or muscle twitching, involuntary movements, altered, blurred, or double vision or other visual abnormalities, dizziness, disorientation, impaired balance, impaired hand-eye coordination, excessive sweating, increased salivation, nausea, lightheadedness, discomfort or pain in the head or eyes, drowsiness, fatigue, or any symptoms similar to motion sickness”, also verschiedenste Probleme beim Sehen und an den Augen sowie Probleme der Konzentration, Koordination und Balance. [2]

HTC dagegen gibt für die Nutzung der HTC Vive beziehungsweise Vive Solo kein genaues Mindestalter an. Sie geben allerdings an, dass das Gerät nicht dafür ausgelegt ist von kleinen Kindern genutzt zu werden. Sie warnen davor, dass Kinder Kleinteile verschlucken könnten oder sich und Andere auf anderem Wege damit verletzen können. Für ältere Kinder empfehlen sie die Aufsicht einer erwachsenen Person und dass die Nutzungszeit nicht zu lang ist. [3]

Zudem wird für die Nutzung der HTC Vive ein HTC Account benötigt, der laut HTC erst ab 14 Jahren erlaubt ist. [4]

In ihren FAQs gibt Sony an, dass man zur Nutzung ihrer Playstation VR Konsole mindestens zwölf Jahre oder älter sein sollte. Weitere Angaben oder Gründe dieses Mindestalters sind auch hier nicht zu finden. [5]

VR-Gerät	Mindestalter	Weitere Angaben
Oculus Rift	13 Jahre	keine
Oculus Go		
Oculus Quest		
HTC Vive	keine genaue Altersangabe	HTC Account erst ab 14 Jahren
HTC Vive Solo		
Google Daydream	13 Jahre	keine
Samsung Gear VR	13 Jahre	nur unter Aufsicht eines Erwachsenen regelmäßig Pausen machen Warnung vor einer Vielzahl an Symptomen aus dem Bereich Koordination, Balance und Sehen
Playstation VR	12 Jahre	keine
Google Cardboard	keine Angabe	nur unter Aufsicht eines Erwachsenen

Tabelle 4.1: Übersicht der verschiedenen VR-Headsets mit ihren jeweiligen Nutzungsmindestaltern

Gegenüber all der Warnungen der Geräte-Hersteller gibt Martin Banks, Professor of Optometry, Vision Science, Psychologie, and Neuroscience an der University of California in Berkeley in einem Interview im Frühling 2016 an, dass er “no concrete evidence that a child of a certain age was somehow adversely affected by wearing a VR headset,” [keine konkreten Beweise, dass ein Kind in einem gewissen Alter durch das Tragen von VR-Brillen negativ beeinflusst wurde] gefunden hat. Er ist überzeugt, dass die Hersteller der VR-Headsets die Nutzung durch Kinder ausschließen, um sicher sein zu können, dass nicht später bekannt werdende Probleme bei Kindern, die VR-Headsets nutzen, ihnen angelastet werden können.

Weiter gibt er an, dass die Angst, dass die Entwicklung des Auges negativ beeinflusst wird im Gegensatz zur Nutzung von Büchern oder Smartphones viel unproblematischer ist, das durch die in die VR-Brillen eingebauten Optiken das Auge gar nicht auf eine so nahe Sache fokussiert, sondern auf weiter entfernte und somit keine Schäden der Augen nach sich zieht.

Banks sieht als Gefahren lediglich die gleichen, die auch für Erwachsene bestehen: Das sind hauptsächlich Virtual Reality Sickness, auch bekannt als Cybersickness, und die Gefahr mit Personen oder Gegenständen im Raum zu kollidieren, während das VR-Headset getragen wird. Ansonsten bewertet er die Nutzung der VR-Brillen von Kindern unproblematisch. [6]

Momentan gibt es keine veröffentlichten Forschungsarbeiten zu den Gefahren für Kinder bei der Nutzung von VR-Brillen, dagegen sind viele Arbeiten zu finden, die VR-Systeme in der Therapie von verhaltensauffälligen, lernverzögerten oder behinderten Kindern erfolgreich einsetzen. Es gibt beispielsweise Arbeiten, die (HIER NOCH GUTE BEISPIELE RAUSSUCHEN)

5 360°-Inhalte

Neben der verwendeten Hardware spielen die verfügbaren Inhalte für das Virtual Reality Erlebnis eine erhebliche Rolle. Nur wenn die gezeigten Inhalte richtig in der 360°-Umgebung dargestellt werden bringt die Darstellung Nutzung von VR-Headsets die gewünschte Immersion.

Für 360°-Inhalte gibt es keine speziellen Dateiformate, die Inhalte werden in den gewohnten Formaten gespeichert. Das Panorama wird dazu auf ein zwei-dimensionales Bild gemappt, um problemlos in den bekannten Formaten speicherbar zu sein. Die Information, wie das Panorama gemappt wurde, wird dann in den Metadaten der Datei gespeichert und 360°-fähige Software kann mit diesen Informationen die Panoramas dann wiederum richtig darstellen.

5.1 360°-Fotos

5.1.1 Projektionen von 360°-Fotos

Equirectangular Rectilinear Cylindrical Mercator Circular Pannini/ Vedutismo Stereographic Cubemap/ Skybox

5.1.2 Quellen für 360°-Fotos

Flickr-Group: <https://www.flickr.com/groups/360images4schools/> Flickr Search equirectangular + Creative Commons:

Google Photo Sphere Community: <https://plus.google.com/communities/115970110085205516914/stream/abbc1e718239-4ab0-9de3-3f6429e7681f> Download nicht immer möglich

www.360cities.net Relativ teuer, dafür tolle Fotos

<https://www.airpano.com> Teuer, hat auch Videos

5.1.3 Erstellung und Bearbeitung von 360°-Fotos

Metadaten wichtig: <https://www.panotwins.de/technical/how-to-add-mandatory-photo-sphere-meta-data-to-an-equirectangular-image/> <https://developers.google.com/streetview/spherical-metadata>

5.2 360°-Videos

5.2.1 Quellen für 360°-Videos

5.2.2 Erstellung und Bearbeitung von 360°-Videos

5.3 3D-Modelle

5.3.1 Quellen für 3D-Modelle

5.3.2 Erstellung und Bearbeitung von 3D-Modellen

5.4 360° Sound

5.4.1 Quellen für 360°-Sounds

5.4.2 Erstellung und Bearbeitung von 360°-Sounds

5.5 Virtual Reality Spiele und Anwendungen

5.5.1 Quellen für Games

HTC: SteamVR Store Listings für alle Geräte: <https://www.wearvr.com/> Oculus devices: <https://www.oculus.com/experiences> Playstation VR Store: <https://store.playstation.com/de>

de/grid/STORE-MSF75508-PLAYSTATIONVR/1 WebVR: <https://itch.io/games/tag-webvr>,

5.5.2 Erstellung und Bearbeitung von Spielen und Anwendungen

Unity: C# Unreal Engine: C++, Blueprints

mobile VR can be JS or native WebVR is JS Google Cardboard: Android

5.6 360°-Inhalte in 3D

5.7 360°-Inhalte in React360

Photos+Videos: <https://facebook.github.io/react-360/docs/photos-and-videos.html>

Audio: <https://facebook.github.io/react-360/docs/audio.html> 3D-Modelle:
<https://facebook.github.io/react-360/docs/objects.html>

6 Software Entwicklung von VR-Systemen

6.1 Native VR-Applikationen

C++

6.2 WebVR + Mobile VR: VR im Browser

6.2.1 A-Frame

6.2.2 React360

6.2.3 Three.js

7 Software Projekt: VRClassroom

Aufbauend auf den Erkenntnissen aus ANDEREN KAPITELN wurde im Rahmen dieser Arbeit VRClassroom entwickelt. VRClassroom ist ein zweiteiliges System, das es Lehrkräften ermöglichen soll 360°-Fotos, -Videos und 3D-Modelle im Unterricht zeigen zu können, dabei die Anzeige auf allen Geräten steuern zu können und trotzdem den Überblick in der Klasse behalten zu können.

VRClassroom bietet dafür die Lehrer-App, die auf einem Rechner läuft und die Schüler-App, die über den Browser der mobilen Geräte beziehungsweise VR-Headsets erreicht werden kann.

Die Lehrer-App ist eine Electron-App, die wiederum eine React-App für das Interface enthält und die Schüler-App startet. Die Schüler-App ist eine React360-Applikation. Der Code von VRClassroom ist also komplett in Javascript gehalten, nur unterschieden durch die verschiedenen Libraries Node.js, React.js, React360. Im Anschluss an diese Arbeit wird VRClassroom unter einer Creative Commons Lizenz veröffentlicht und so kann ein Entwickler schnell alle Teile des Codes verstehen ohne sich in mehrere Programmiersprachen einarbeiten zu müssen.

7.1 Nutzungsszenario und Anwendungsfokus

VRClassroom wurde speziell für die Nutzung in der Schule entwickelt. Dabei kann die Lehrkraft das VR-Erlebnis für die SchülerInnen führen und ihnen so komplexe 3-dimensionale Inhalte besser vermitteln und eine spannende neue Art des Lernens zu erleben. Außerdem wurde Funktionen wie etwa die Seitenleiste mit den verbundenen Geräten entwickelt, die es der Lehrkraft erleichtern sollen VRClassroom in einer Schulkasse einzusetzen und sicher zu stellen, dass alle Geräte verbunden sind und die Inhalte angezeigt werden.

Da viele Schulen noch überhaupt keine Ausrüstung an VR-Headsets haben, lag der Fokus besonders auf der Nutzung des VRClassroom Systems mit einem Smartphone in einem Google Cardboard. Allerdings wurde es bewusst so entwickelt, dass auch mit "echten" VR-Headsets das System problemlos weiter genutzt werden kann. So wird den Schulen ermöglicht mit einer sehr geringen Investition zu testen, ob es für sie in Frage kommt und können zu einem späteren Zeitpunkt zu einem elaborierteren Hardware-System wechseln ohne auf neue Software umsteigen zu müssen.

Weitere Einsatzszenarien für VRClassroom könnten Besprechungen im Arbeitsumfeld gehen, bei denen es sich um plastische Inhalte handelt wie zum Beispiel Architekturbüros, Designagenturen oder Landschaftsgärtner. Mit einem 360°-Foto oder -Video könnte der Ist-Zustand besprochen werden und anschließend die Entwürfe in 3D-Modellen vorgeführt werden. Dadurch könnten sich Kunden besser in die Entwürfe hineinversetzen und bewusster entscheiden, was sie letztendlich haben möchten.

7.2 Anwender

Da an Schulen Lehrkräfte und Schülerinnen und Schüler verschiedenster Altersstufen und Leveln an Technikaffinität zu finden sind sollte die Software

- Hauptszenario: Lehrer und Schüler im Klassenzimmer
- mögliche andere Szenarien: Lerngruppen, Workmeetings
- Anwender: Lehrer und Schüler aller Fachrichtungen und mit unterschiedlichsten Skillleveln
- deshalb muss es sowohl für Lehrer als auch für Schüler möglichst intuitiv und einfach zu bedienen sein!
- Augenmerk bei der Entwicklung lag besonders darauf

7.3 Grundstruktur

Das VRClassroom System besteht aus zwei verschiedenen Applikationen: Zum einem der Schüler-App, die auf den VR-Systemen läuft mit denen die Schüler sehen können, was der Lehrer ihnen präsentiert, und die Lehrer-App, in der die Lehrkraft die verschiedenen Inhalte hineinladen, Markierungen auf die Inhalte setzen kann und sehen kann, welche Schüler-Geräte aktuell verbunden sind.

Da viele Lehrer in ihrer Ausbildung oft nicht mit vielen neuen Technologien in Berührung gekommen sind, sondern es gewohnt sind mit den “klassischen” Medien zu arbeiten, lag der Augenmerk bei der Entwicklung darauf, dass das benötigte technische Verständnis möglichst niedrig ist und auch Personen, die sich selbst als nicht technikaffin bezeichnen würden, keinerlei Probleme bei der Nutzung haben. Gleichermaßen gilt selbstverständlich auch auf der Seite der Schüler. Da diese aber hauptsächlich passiv agieren, standen hier die Lehrkräfte im Mittelpunkt der Aufmerksamkeit.

7.4 Lehrer-Applikation

Die Lehrer-Applikation besteht aus einer Electron-App, die in zwei logische Teile zerlegt ist. Das ist zum einen der main-Prozess, der es erlaubt Zugriffe auf das File-System des Rechners zu machen und für rechenaufwändige Hintergrundprozesse genutzt wird, und zum Anderen der render-Prozess. Der render-Prozess ist der Teil des Programms, das die Lehrkraft letztendlich auf ihrem Bildschirm sieht.

Die Lehrer-App enthält genau genommen zwei render-Prozesse: Die teacher-App, in der die Lehrkraft alle verbundenen Geräte sehen kann und verschiedene Inhalte hineinladen kann, und die student-App, die als iFrame in die teacher-App eingebunden ist und auf den Geräten der Schülern läuft.

Die teacher-App enthält außerdem noch den QR-Code Generator, der in einem zweiten Fenster geladen wird.

Um eine Liste der verbundenen Geräte zu halten und Veränderungen der Inhalte auf die Schüler-Geräte zu synchronisieren, startet die teacher-App einen WebSocket-Server, mit dem sich alle Schüler-Geräte verbinden.

7.4.1 Verbundene Geräte

Wie in der Grafik zu sehen hält die teacher-App eine Liste mit allen verbundenen Geräten dieser Session. Sind die Geräte gerade aktiv, werden sie mit einem grünen Icon dargestellt, sind sie inaktiv, mit einem Roten.

Das soll der Lehrkraft erleichtern zu überprüfen, ob die Schüler den gezeigten Stoff verfolgen oder sich anderweitig beschäftigen.

Haben die Schüler bereits einen Namen eingegeben, wird dieser in der Liste angezeigt. Ist dies nicht der Fall wird aus dem user-agent versucht möglichst genau zu schließen, um welches Gerät es sich handelt, sodass der Lehrer zumindest einschränken kann, um welchen Schüler bzw welche Schülerin es sich handeln könnte.

NEU: Bei Modellen und Videos der loading-Status der einzelnen Geräte

Ein User-Agent kann zum Beispiel wie folgt aussehen:

```
Mozilla/5.0 (iPhone; CPU iPhone OS 5_0 like Mac OS X) AppleWebKit/534.46  
(KHTML, like Gecko) Version/5.1 Mobile/9A334 Safari/7534.48.3
```

Daraus lässt sich schließen, dass es sich um eine iPhone handelt, welches Safari benutzt, um die Schüler-App zu laden. Die Oculus Geräte hingegen geben in ihrem User-Agent an, den Oculus Browser zu verwenden und sind so auch gut von den anderen verbundenen Geräten zu unterscheiden. Da aber hier nicht ersichtlich wird welches Oculus-Gerät es genau ist wird nur “Oculus device” angegeben und nicht genauer spezifiziert, ob es sich dabei um eine Go, Quest oder Rift handelt.

- startet Websocket Server
- generiert QR-Code
- Lehrer kann verschiedene Apps starten:
 - 360° Photos und 3D-Modelle anzeigen und Markierungen einfügen
 - 360° Videos synchronisiert auf allen Geräten zeigen und play/pause aus Lehrer-App steuern
 - Streetview photos von locations laden und anzeigen (?)
- hält history der zuvor gezeigten Inhalte, um sie vereinfacht wieder anzuzeigen

7.4.2 Hineinladen von Inhalten

7.4.3 Controls

Wie in Grafik 7.1 zu sehen sind die Controls, mit denen Aktionen des aktuell geladenen Inhalts ausgelöst werden können durch einen Overlay über dem iFrame der React360-App dargestellt.

Für jeden der drei Typen an Inhalten, 360°-Fotos, -Videos und 3D-Modelle unterscheiden sich die Bedienelemente und Funktionen, das Setzen einer Markierung ist allerdings bei allen Medientypen gegeben.

Marker Soll ein Marker gesetzt werden, muss zuerst in den “Markierung setzen”-Modus gewechselt werden, indem in der Control Bar auf “Marker setzen” geklickt wird. Danach kann an beliebiger Stelle eine Markierung gesetzt werden.

Um die 3D-Koordinaten des Markers zu bekommen wird mit Hilfe von Mouseposition und Fenstergröße ein Strahl berechnet, der aus der 2D-Koordinate der Mouseposition ausgeht. Bei 360°-Fotos und -Videos wird die Entfernung zur Kamera auf einen festen Wert gesetzt, der so gewählt wurde, dass er gerade noch im Zylinder der Welt liegt. Mit der Entfernung zur Kamera lässt sich die 3D-Koordinate dann leicht berechnen.

Um Markierungen auf 3D-Modellen setzen zu können muss ein Schnittpunkt des 3D-Modells mit dem virtuellen Strahl berechnet werden. (Soll das wirklich rein?) In React360 gibt es momentan keine Möglichkeit auf diese Weise auf die in der 3D-Welt platzierten 3D-Modelle zuzugreifen, sodass

Ist der Schnittpunkt des Strahls mit dem 3D-Modell berechnet, wird derjenige Schnittpunkt als Koordinate verwendet, der am nächsten zur Kamera ist. An dieser Stelle wird dann der Marker platziert. Zudem werden die Marker invers zur Nähe zur Kameraposition skaliert, sodass sie immer in der gleichen Größe dargestellt werden, egal wo am Modell sie gesetzt sind.

Das verwendete Modell erinnert wie in Grafik YYY zu sehen an Stecknadeln, sodass Nutzern die Bedeutung direkt verständlich ist. Um die Markierungen leichter erkennbar zu machen sind sie in einer leuchtenden Farbe und drehen sich, sodass die Aufmerksamkeit direkt auf die markierte Stelle gelenkt wird.

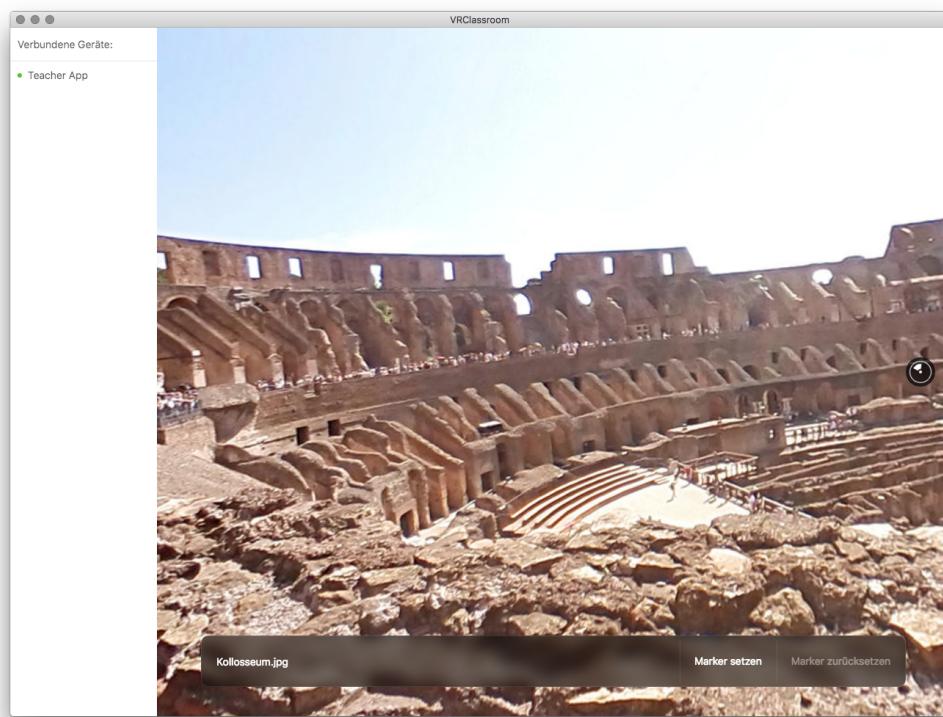


Abbildung 7.1: Die Kontrollleiste als Overlay über den iFrame der React360-App.



Abbildung 7.2: Die Kontrolleiste für 360°-Photos.

Photo Controls Ist ein Foto geladen beinhaltet die Kontrollzeile nur den Dateinamen des 360°-Fotos und die Buttons zum Setzen von Markierungen, die wie bereits beschrieben bei allen Medientypen gegeben sind. Grafik 7.2 zeigt die Kontrollleiste für 360°-Fotos.

Video Controls Wird ein Video gezeigt, hat die Lehrkraft mehrere unterschiedliche Funktionen in der Kontrollleiste: Wie in Grafik 7.3 zu sehen sind die Elemente ähnlich wie bei bekannten Videoplayern wie Quicktime oder Netflix gehalten. Ganz links ein kombinierter Play/Pause-Button, der für alle verbundenen Geräte synchronisiert das Abspielen beziehungsweise Pausieren des Videos auslöst, daneben die aktuelle Abspielzeit, gefolgt von einem Slider, der grafisch die aktuelle Position im Video darstellt. Außerdem kann mit dem Slider zu anderen Zeitpunkten im Video gesprungen werden. Rechts daneben wird die Gesamtdauer des Videos angezeigt. Der letzte Videospezifische Button ist der Sound-Button, der im aktiven Zustand auf alle Geräten den Ton des Videos abspielt. Der Sound-Button ist standardmäßig deaktiviert.

Zudem sind wiederum die Buttons für das Setzen von Markierungen vorhanden. Sie sind während dem Abspielen des Videos deaktiviert und können nur genutzt werden, wenn das Video pausiert ist. Wird das Video dann wieder weiter abgespielt, werden alle gesetzten Markierungen automatisch zurückgesetzt.

Model Controls Wie in Grafik 7.4 zu sehen ist auch in der Kontrollleiste bei 3D-Modellen ein Slider vorhanden. Dieser kann sowohl zum Drehen des Modells als auch zum Skalieren benutzt

7 SOFTWARE PROJEKT: VRCLASSROOM



Abbildung 7.3: Die Kontrolleiste für 360°-Videos.



Abbildung 7.4: Die Kontrolleiste für 3D-Modelle.

werden. Dazu sind links vom Slider die Buttons “Drehen” und “Skalieren” mit denen zwischen den zwei Funktionalitäten des Sliders gewechselt werden kann. Der aktive Modus wird durch die blaue Farbe dargestellt.

Auch auf 3D-Modellen können Markierungen gesetzt werden. Dabei ist zu beachten, dass nur auf dem Modell eine Markierung gesetzt werden kann. Wird außerhalb des 3D-Modells geklickt, wird keine Markierung gesetzt.

7.5 Kommunikation zwischen Lehrer-App und Schüler-App

WebSocket Server from Teacher App -> Schüler Gerät meldet sich bei Laden der URL an Teacher App antwortet mit aktuellem Content Teacher App schickt bei jeder Änderung ein komplettes Content-Objekt (einige Einträge ggf. leer) Nachrichten von Schüler Geräten: visibilitychange, loading, Name Nachrichten von Teacher-App: Jede Veränderung des Inhalts, URL zu Inhalt

7.6 QR-Code Fenster

Das QR-Code Fenster ist ein zweites Browserfenster, das aus dem main-Prozess der Electron App auf dem Lehrer-Computer gestartet wird. Es zeigt, wie in 7.5 zu sehen, einen QR-Code, den die SchülerInnen mit ihren Smartphones scannen können, um bequem die URL zu laden, auf der sie die Schüler-Applikation erreichen können. Der QR-Code wird dynamisch beim Öffnen der App generiert.

Für Geräte, die keine Kamera haben oder wenn das Scannen des QR-Codes fehlschlägt, wird zudem unterhalb des QR-Codes die URL angezeigt, unter der die Schüler-Applikation zu erreichen ist.

Der QR-Code wird in diesem extra Fenster generiert und angezeigt, damit die Lehrkraft dieses Fenster auf einem Beamer anzeigen kann, um den SchülerInnen den Zugang zur Schüler-App ohne umständliches URL-Abtippen zu ermöglichen.

7.7 Schüler-App

In der VRClassroom App nehmen die Schüler eine passive Rolle ein und können selbst nicht in der 3D-Welt navigieren.

Die WebApp für die Schüler ist eine React360-App, sie stellt eine 3D-Welt da, in deren Mitte sich die Kamera, also der Viewport der Geräte, befindet. Sie stellt die in der Lehrer-App hineingeladenen 360°-Fotos und Videos in einer Kugel um den Viewport da, sodass die Nutzer sich in alle Richtungen umsehen können. 3D-Modelle werden in kurzer Entfernung vor dem Viewport angezeigt, als befänden sich sich im Raum vor der Person.



Abbildung 7.5: Qr-Code Fenster.

7.7.1 Eingabe des Namens

Läßt ein Gerät zum ersten Mal die VRClassroom-App wird dem Nutzer neben der Begrüßung, die immer angezeigt wird bevor Inhalte hineingeladen werden, eine Tastatur im VR-Raum angezeigt. Damit können die Schüler ihren Namen eingeben, der dann in der Liste der verbundenen Geräte angezeigt wird. Wird ein Smartphone als VR-Headset genutzt, muss die Tastatur mittels der Toucheingabe auf dem Display gemacht werden, bevor die WebVR Ansicht geladen wird. VR-Headsets, die einen Controller haben, können den Namen mit dem Controller auf der Tastatur tippen.

Damit die Schüler nicht jedes Mal wieder ihren Namen eingeben müssen, wird der eingegebene Name im Browser des Schüler-Gerätes gespeichert und beim erneuten Laden der VRClassroom Applikation direkt wieder an die Lehrer-App übermittelt.

7.7.2 Anzeigen von 360°-Fotos

Um 360°-Fotos in der 3D-Szene anzuzeigen lädt die React360-App das Foto von dem Link, den die Teacher-App geschickt hat und setzt sobald das gesamte Foto geladen wurde das Bild als Hintergrund der Szene. Gleichzeitig wird bei erfolgreichem Laden des Fotos eine Nachricht an die Teacher-App geschickt, um dem Status des Geräts von “loading” wieder auf “active” zu setzen. Das 360°-Bild wird dazu von innen an die Kugel in der die React360-Szene ist projiziert und ergibt so durch seine equirectanguläre Projektion ein unverzerrtes 360°-Bild.

Das Bild wird angezeigt bis VRClassroom von der Lehrkraft beendet wird oder etwas anderes in die App hineingeladen wird.

7.7.3 Abspielen von 360°-Videos

Sendet die Teacher-App die Nachricht, dass ein Video angezeigt werden soll, startet die React360-App sofort mit dem Laden des Videos und lädt soviel von dem Video wie es geht. Ist eine Zeit des Videos vorgeladen, sodass React360 davon ausgeht, dass das Video ruckelfrei abgespielt werden kann, sendet wird wiederum die Nachricht, dass das Gerät aus dem “loading”-Status wieder auf “active” gesetzt wird. Ähnlich wie bei 360°-Fotos wird das 360°-Video dann als Hintergrund-Video der Szene gesetzt, um als 360°-Video abgespielt werden zu können

Die React360-App spielt das Video nicht automatisch ab, sobald genug geladen ist, sondern wartet auf das Signal der Teacher-App, um das Abspielen zu starten. Alle Funktionen werden erst ausgeführt wenn das Signal der Teacher-App kommt, die Funktion auszuführen. Zu den Funktionen zählen: Play, Pause, Springen zu einer anderen Stelle im Video und den Ton des Videos abspielen.

Falls ein Schüler-Gerät sich verspätet verbindet oder ein Paket verloren gegangen ist, sendet die Teacher-App jede Sekunde eine neue Nachricht in der immer der Link zum aktuellen Video und die aktuelle Abspielposition enthalten ist. Unterscheidet sie sich mehr als eine Sekunde von der Abspielposition in der React360-App wird zu der Abspielposition, die in der Nachricht steht gesprungen.

7.7.4 Abspielen von Ton in Videos

Da die meisten Browser das automatische Abspielen von Videos mit Ton verbieten, werden 360°-Videos von React360 standardmäßig beim erstellen des Videoplayer-Komponenten “muted” auf “true” gesetzt. Die Hersteller wollen dadurch die Ablenkungen, die beim surfen auf den Nutzer zukommen, abmildern. Erst wenn der Nutzer ein “user gesture click” also einen Klick auf der Website gemacht hat, darf die Tonspur automatisiert abgespielt werden. Wird versucht ein Video mit Ton abzuspielen, ohne dass ein Klick gemacht wurde, wird der Videoplayer blockiert und kann nicht mehr abspielen. [7]

Damit React360 in einer single-threaded Umgebung wie einem Webbrower flüssig ablaufen kann und nicht durch “blocking behavior” irgendeiner Art das Rendern unterbrochen wird, ist eine React360-App in zwei Teile aufgeteilt: Die React-Applikation und den Code, der die React Komponenten in 3D Elemente auf dem Bildschirm umwandelt. Die App selbst läuft in einem Webworker, einem anderen Prozess als der des Hauptbrowserfensters. [8]

Das führt dazu, dass die React360-Elemente nicht als html-Elemente gelten und ein Click-Event auf einem VRButton nicht als Interaktion zählt, um die Erlaubnis zu haben Ton abzuspielen.

Um dieses Hindernis zu umgehen ist nun ein durchsichtiger, Bildschirm-füllender Button über die React360-App gelegt, der bei einem Klick verschwindet, um die Erlaubnis vom Browser zu bekommen Ton abzuspielen. Wurde der Button am Schüler-Gerät geklickt wird ein flag gesetzt, dass die App Ton abspielen darf. Ob dann tatsächlich Ton beim Video abgespielt wird, kann der Lehrer aus der teacher-App einstellen. Auch dort ist das Abspielen von Ton an den Schüler-Geräten standardmäßig erst einmal abgestellt, damit der Lehrer entscheiden kann, ob er nur den Ton aus dem Rechner über Boxen für Alle abspielen möchte oder aus jedem Schüler-Gerät einzeln der Ton kommen soll.

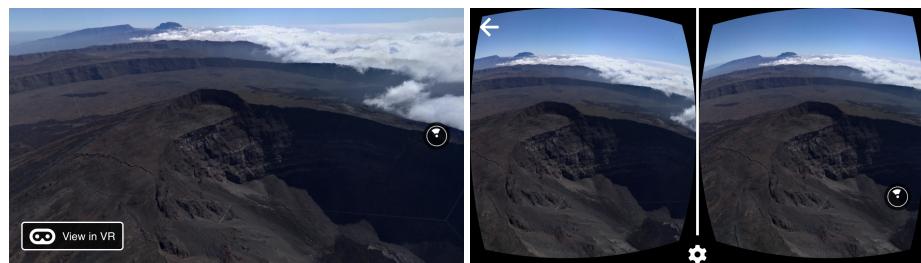
7.7.5 Anzeigen von 3D-Modellen

7.7.6 WebVR Polyfill

WebVR ist eine Javascript-API, um Virtual Reality-Inhalte im Browser anzuzeigen. [9]

Wie in Grafik 7.6a links zu sehen wird auf einer Web-App, die mit WebVR angesehen werden kann ein Icon in der unteren rechten Ecke des Fensters zu angezeigt, mit dem dann die WebVR-Ansicht geladen werden kann.

Wie Grafik 7.6b zeigt teilt WebVR dafür den Bildschirm in zwei Bilder für die Linsen in einem VR-Headset wie zum Beispiel dem Google Cardboard auf. Die zwei Bilder sind dabei nicht Bildschirm-füllend, sondern in einer annähernd ovalen Form, die von einem schwarzen Rand umgeben wird, sodass sie gut auf die Linsen passen. Durch die Krümmung der Linsen wird daraus dann eine drei dimensionale 360°-Welt, in der der Nutzer sich umsehen kann.



(a) “View in VR” Button in VRClassroom. (b) VR-Szene mit WebVR Ansicht.

Abbildung 7.6: VR-Szene mit und ohne WebVR Ansicht.

Bisher ist weder in Chrome noch in Safari WebVR standardmäßig unterstützt. In Chrome auf Android kann es durch setzen der “WebVR”- und “Gamepad-Extension”-flags auf “enabled” eingeschaltet werden. [10] In Safari ist es bisher gar nicht möglich WebVR zu nutzen.

Da aber VRClassroom auf allen mobilen Geräten funktionieren soll und möglichst auch keine Einstellungen auf der Nutzerseite benötigen soll, wurde ein Polyfill benutzt, um die WebVR Funktionen für die VRClassroom-App auf allen Geräten nutzen zu können. Dafür

8 Nutzerstudie und Evaluation

8.1 Ursprünglich geplanter Ablauf der Studie

Da das System speziell für die Nutzung im Schulunterricht entwickelt wurde, sollte auch die Nutzerstudie in diesem Szenario durchgeführt werden. Dafür wurden Lehrkräfte angefragt, die dann einen Teil ihrer Unterrichtszeit für einen Test der Software verwenden wollten und im Anschluss bereit waren einen kurzen Fragebogen dazu zu beantworten.

Insgesamt hatten sich 15 Lehrkräfte bereit erklärt mit ihrer Klasse das VRClassroom System auszuprobieren und im Anschluss einen Fragebogen auszufüllen.

Für den Test sollte allen SchülerInnen ein Cardboard zur Verfügung gestellt werden, das die Kinder mir ihrem eigenen Smartphone als VR-Brille genutzt sollten. Der Rechner, auf dem das Programm installiert war, sollte den Lehrkräften bereitgestellt und den Lehrkräften im Vorfeld eine kurze Einführung in die Software gegeben werden. Dies ist trotz Absage der Studie durch ein Video geschehen. Das Video wurde zudem für die Online-Umfrage benutzt.

Für die Durchführung der Studie war ein Zeitrahmen von 15 bis 20 Minuten veranschlagt. In dieser Zeit sollte die Lehrkraft die App starten, sich alle Schüler-Geräte damit verbinden und die Lehrkraft die eigentlichen Inhalte präsentieren.

Damit alle Studienteilnehmer, Lehrer und Schüler gleichermaßen, die gleichen Erlebnisse haben und sich zu allen Funktionen des Systems eine Meinung bilden können, hatte jede Lehrkraft ein vollständiges Set an Inhalten mit jeweils mindestens einem 360°-Fotos, einem 360°-Video und einem 3D-Modell.

8.1.1 Freigabe Kulturministerium

Da alle Studien, die an Schulen gemacht werden, einer Freigabe des Kultusministeriums bedürfen, wurde im Vorfeld der Studie beim Kultusministerium eine solche Freigabe beantragt. Für den Antrag werden alle Fragebögen und der gesamte Ablauf der Studie an das Kultusministerium zur Prüfung vorgelegt. Das Kultusministerium hat die Durchführung von Studien beschränkt, um den Datenschutz der Schülerinnen und Schüler zu sichern. Da allerdings in der Studie nur die Lehrkräfte befragt werden und die Fragebögen komplett anonym gehalten sind, sind die Daten der Schüler zu keinem Zeitpunkt in Gefahr.

Leider wurde der Antrag nie beantwortet, sodass die gesamte Studie kurzfristig abgesagt werden musste. Es hatten sich bereits 15 Lehrkräfte bereit erklärt mit ihrer Klasse das VRClassroom System auszuprobieren und im Anschluss den Fragebogen zu beantworten, die sehr an dem System interessiert waren. Da das VRClassroom-System speziell für den Einsatz im Unterricht entwickelt wurde, wäre eine Nutzerstudie im Unterricht sehr sinnvoll gewesen, um abschätzen zu können, ob es im aktuellen Zustand einen guten Mehrwert bietet oder noch Anpassungen braucht um gut im Unterrichtsablauf zu funktionieren.

8.2 Qualitative Nutzerstudie

8.2.1 Fragebogen

8.2.2 Erkenntnisse

8.3 Online-Umfrage

Um die Meinung möglichst vieler verschiedener Lehrkräfte zum entwickelten VRClassroom System zu erfahren, wurde neben der qualitativen Studie, in der die Studienteilnehmer das System ausprobieren und im Anschluss Feedback geben konnten, noch eine Online-Befragung von Lehrkräften in Bayern durchgeführt.

Dafür wurde ein Video produziert, welches die Idee und Nutzung von VRClassroom erklärt und die einzelnen Funktionen zeigt, die die App anbietet. Im Anschluss haben die teilnehmenden Lehrkräfte einen kurzen Fragebogen ausgefüllt.

- Teilnehmer: Anzahl, geschlechterverteilung, Alter, Schulen
- Fragen der Studie
- Antworten der Lehrer
- Schlüsse daraus

9 Ausblick

9.1 Zukunft von VR

9.2 Probleme von VRClassroom

In Tests haben sich noch ein paar Probleme gezeigt, die das VRClassroom System in seinem aktuellen Zustand noch hat, die beim “echten” Einsatz im Schulunterricht das Erlebnis stören könnten. Diese Problem konnten aber leider nicht im Rahmen dieser Arbeit gelöst werden, das sie nicht wirklich Probleme der entwickelten Software sind, sondern technischen Grenzen von Geräten, die sich in der Zukunft aller Voraussicht nach weiter verschieben werden, sodass sie dann kein Problem mehr darstellen.

9.2.1 Anzeigen großer Dateien

Eines der größten Probleme, die VRClassroom momentan hat, ist die Anzeige von komplexen 3D-Modellen oder 360°-Videos mit einer großen Dateigröße. 360°-Videos sind besonders problematisch, wenn sie eine extrem hohe Auflösung (>4096) und eine hohe Framerate haben.

Für diese Probleme gibt es allerdings momentan nicht wirklich eine Lösung, da die Daten über das WLAN übertragen werden müssen und bei 25-30 Geräten, die verfügbare Bandbreite zwischen allen Geräten geteilt werden muss und das einfach nicht genug pro Gerät hergibt, um so hohe Datenmengen schnell genug zu übertragen.

(Hier vielleicht noch eine Vorrechnung mit einem geläufigen Router einfügen?)

Zudem war bei Tests vereinzelt zu sehen, dass bei extrem komplexen Modellen WebVR crasht und ein Neuladen der Schüler-App herbeigeführt wird, sodass das Gerät aus dem Cardboard genommen und die WebVR-Ansicht neu aktiviert werden muss.

9.2.2 Synchronisierung

Ein weiteres Problem bei VRClassroom ist es, alle Veränderungen exakt synchron auf allen Geräten darzustellen. Besonders beim Abspielen von 360°-Videos fällt dieses Problem auf.

Der WebSocket-Server broadcastet die Nachricht über den veränderten Inhalt an alle Clients und da die Nachricht nur eine kurze JSON-Datei ist, wird sie auch instantan auf allen Geräten empfangen. Trotzdem kommt es zu geringen Verschiebungen der Anzeigeaktualisierung bei den einzelnen Geräten. Der Ton der Videos ist dann um einen Bruchteil einer Sekunde verschoben, was zu unangenehmen Vermischungen der Sounds der verschiedenen Geräte und Halleffekten führt.

Das kann dadurch umgangen werden, dass nur der Sound aus der Teacher-App abgespielt wird, die Schüler Ohrstöpsel tragen oder sie die Abspiellautstärke auf ihren Geräten weit herunterschrauben, dass es nicht durch den ganzen Raum schallt. Eine echte Lösung dafür gibt es allerdings nicht.

9.3 Mögliche Weiterentwicklungen an VRClassroom

VRClassroom ist ein vollfunktionsfähiges System, dass laut Umfrageergebnissen bereits bei einigen Lehrkräften das Interesse geweckt hat, es in ihrem Unterricht einzusetzen. Trotzdem sind in der Entwicklung und bei der Online-Befragung noch mögliche Weiterentwicklungen von VRClassroom aufgefallen, die zukünftig noch integriert werden könnten.

9.3.1 Hosten von VRClassroom Online statt Verbindung mit Lehrer-Rechner

Eine mögliche Weiterentwicklung von VRClassroom könnte sein das ganze VRClassroom System nicht lokal auf dem Rechner der Lehrkraft laufen zu lassen, mit dem sich dann alle VR-Geräte verbinden, sondern auch eine Version anzubieten, die Online gehostet ist.

Die Entscheidung VRClassroom so zu entwickeln, dass es lokal läuft wurde allerdings ganz bewusste gefällt: Durch Vorgespräche mit den Lehrkräften, die an der Studie teilnehmen wollten und aus eigener Erfahrung wurde schnell klar, dass jede Schule unterschiedlich gut mit Netzwerk versorgt ist. Es gibt Schulen, die für Schüler gar kein WLAN anbieten oder dieses stark beschränken, sodass direkte Verbindungen zwischen den Geräten im Netzwerk unterdrückt werden, oder andere Restriktionen in ihrem Netz haben. Um diese Probleme zu umgehen, wurde dann die Entscheidung gefällt, VRClassroom so zu entwickeln, dass es unabhängig vom Internet funktionieren kann. Einzig die Streetview-Funktion fällt dann weg. So kann dann einfach ein lokales Netzwerk eröffnet werden mit dem sich alle Geräte verbinden und VRClassroom problemlos genutzt werden. Dies könnte zum Beispiel mit einem Aufbau gelöst werden, bei dem der Rechner auf dem VRClassroom installiert ist und ein vorkonfigurierter Router immer gemeinsam genutzt werden und sobald Strom angesteckt wird der Router sein lokales Netzwerk aufmacht.

Ein weiteres Argument für die aktuelle Implementierung ist das schon zuvor in (HIER REF) besprochene Problem des Ladens von komplexen Modellen oder hochaufgelösten Videos, also großen Datenmengen. Laden alle Geräte lokal vom Lehrercomputer wird nicht die gesamte Internetverbindung der Schule lahmgelegt und auch die Übertragungsrate ist höher als beim Laden aus dem Internet.

Wird VRClassroom nur in einem lokalen Netzwerk benutzt fällt allerdings wie bereits oben erwähnt die Funktionen zum Anzeigen von Google Streetview Fotos weg, denn dafür wird die Verbindung zum Internet benötigt. Zudem könnte es für technisch weniger verzierte Lehrkräfte schwieriger sein nachzuvollziehen, wo das Problem liegt, wenn etwa ein Schüler Gerät sich nicht verbinden kann, weil es nicht im Netzwerk ist, in dem VRClassroom läuft. Diese Punkte würden wiederum eher dafür sprechen die Struktur zu ändern, sodass VRClassroom online gehostet wird.

Für den Rahmen der Arbeit ist die Entscheidung dafür gefallen, VRClassroom über den Rechner der Lehrkraft zu hosten, aber eine Online-Version wäre definitiv eine interessante Weiterentwicklung.

9.3.2 Skalierung von 3D-Modellen auf echte Größe

Um den Schülern zu vermitteln wie groß beziehungsweise klein die Dinge in echt sind, die gerade besprochen werden, könnte eine Funktion in der Lehrer-App eingefügt werden, die die 3D-Modelle auf Originalgröße skaliert. Dazu müsste selbstverständlich alle 3D-Modelle in Originalgröße vorliegen, was nur sehr selten der Fall ist, da diese Funktion für die meisten Verwendungszwecke nicht benötigt wird und daher einfach unbeachtet bleibt.

Um die gezeigten Inhalte gut zu begreifen und auch in der echten Welt einschätzen zu können, würden Schüler allerdings sehr davon profitieren diesen Einblick zu haben. Denn erst wenn man direkt davor steht, kann man begreifen wie groß der Eiffelturm ist oder wie klein eine extrem giftige Spinne wie die schwarze Witwe in Wahrheit ist.

Wie schon erwähnt müssen dazu zum einen die Modelle in der richtigen Größe vorliegen, aber zum anderen müsste auch die VR-Classroom-App angepasst werden, um diese Funktion sinnvoll zu erfüllen. Wird das Modell auf Originalgröße skaliert, die kleiner ist als die Größe, in der man sie genauer betrachten würde ist das kein Problem und könnte direkt erfüllt werden. Problematisch wird die Skalierung wenn sie extrem viel größer wird als das Modell normal angesehen wird, wie zum Beispiel beim Betrachten eines Gebäudes. Wird das Modell in der App extrem groß skaliert, wird es ab einer gewissen Größe nicht mehr vollständig angezeigt und Teile abgeschnitten. Das passiert, da React360 darauf ausgelegt ist, dass die Kamera in einer Art Kugel ist, in der sie sehen kann. Das dient dazu den Rendering-Aufwand nicht zu extrem werden zu lassen. (QUELLE dafür?) Alles was die Kugel durchbricht wird nur bis zum Rand der Kugel gerendert und alles was außerhalb liegt abgeschnitten.

9.3.3 Navigation von 3D-Modellen aus Schüler-App

Momentan können Schüler keinen Einfluss darauf nehmen, welchen Teil eines 3D-Modells sie sehen können und welchen nicht, da nur aus der Teacher-App gesteuert werden kann wie das Modell skaliert und gedreht wird. Eine mögliche Erweiterung von VRClassroom könnte sein, dass die Schüler die 3D-Modelle selbstständig drehen und skalieren können beziehungsweise zu anderen Punkten am Modell springen können.

9.3.4 Ausfragemodus: Ausgewählter Schüler setzt Markierung

Eine mögliche Weiterentwicklung von VRClassroom wäre eine Art “Ausfragemodus”, bei dem die Lehrkraft aus der Liste der verbundenen Geräte eins auswählen kann, das dann eine Markierung setzen kann. Das ausgewählte Gerät hätte dann die Möglichkeit einmalig eine Markierung zu setzen.

Dafür wäre es notwendig für die Schüler-Geräte eine Möglichkeit zu geben einen Punkt auszuwählen, an dem die Markierung gesetzt werden soll. Bei der aktuellen Implementierung mit einem Smartphone in einem Google Cardboard ist das noch nicht möglich. Dazu müsste mit dem Touch-Event vom Cardboard an der aktuellen Gaze-Position des Nutzers an dieser Stelle ein Marker gesetzt werden. Mit einem Headset, das bereits einen zugehörigen Controller hat, wäre das einfacher zu lösen, da lediglich die Position des “Rays” des Controllers abgefragt werden müsste.

Da die Markierungen auch in der Teacher App über das iFrame mit der Student-App gesetzt werden ist die Kommunikation der Markerposition an die Teacher-App bereits implementiert und müsste nur leicht abgeändert werden. Möglicherweise wäre es gut die von Schülern gesetzten Marker noch in einer anderen Farbe darzustellen wie die der Lehrkraft, um für andere Schüler leichter erkennbar zu machen, welche Marker von welcher Person kommen.

Momentan können Schüler nicht beeinflussen wie sie 3D-Modelle sehen, sondern nur den von der Lehrkraft ausgewählten Blickwinkel. Um auch auf 3D-Modellen sinnvoll Markierungen setzen zu können wäre es dann auch sehr sinnvoll den Schülern die Möglichkeit zu geben den Blickwinkel des 3D-Modells zu verändern.

Inhalt der beigelegten CD

Literatur

- [1] Facebook Technologies LLC, “Oculus Terms of Service,” 2018. [Online]. Available: <https://www.oculus.com/legal/terms-of-service/>
- [2] SAMSUNG, “Samsungs FAQs - Is the Gear VR safe for children?” [Online]. Available: <https://www.samsung.com/uk/support/mobile-devices/is-the-gear-vr-safe-for-children/>
- [3] HTC Corporation, “Safety and regulatory guide,” 2016.
- [4] ——, “HTC Vive FAQs.” [Online]. Available: https://www.vive.com/us/support/viveport/category{_}howto/age-limit-for-htc-account-for-vive.html
- [5] Sony Entertainment LLC, “Playstation VR FAQs,” 2017. [Online]. Available: <https://blog.us.playstation.com/2017/10/02/playstation-vr-the-ultimate-faq/>
- [6] S. Hill, “Is VR too dangerous for kids? We asked the experts,” 2016. [Online]. Available: <https://www.digitaltrends.com/virtual-reality/is-vr-safe-for-kids-we-asked-the-experts/>
- [7] K. Decker, “Auto-Play Policy Changes for macOS,” 2017. [Online]. Available: <https://webkit.org/blog/7734/auto-play-policy-changes-for-macos/>
- [8] Facebook Inc., “React360 Runtime,” 2018. [Online]. Available: <https://facebook.github.io/react-360/docs/runtime.html>
- [9] WebVR, “WebVR.” [Online]. Available: <https://webvr.rocks>
- [10] B. Jones and K. Weidner, “WebVR in Chrome for Android mobile,” 2017. [Online]. Available: https://webvr.rocks/chrome{_}for{_}android