



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Bachelorarbeit

Jan Stieglitz

**Framework zur Integration von Hardware-Devices für  
haptisches Feedback in AR und VR-Anwendungen**

*Fakultät Technik und Informatik  
Studiendepartment Informatik*

*Faculty of Engineering and Computer Science  
Department of Computer Science*

Jan Stieglitz

**Framework zur Integration von Hardware-Devices für  
haptisches Feedback in AR und VR-Anwendungen**

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Bachelor of Science Angewandte Informatik  
am Department Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Birgit Wendholt  
Zweitgutachter: Prof. Dr. Martin Becke

Eingereicht am: 17. Mai 2017

**Jan Stieglitz**

**Thema der Arbeit**

Framework zur Integration von Hardware-Devices für haptisches Feedback in AR und VR-Anwendungen

**Stichworte**

Framework, Haptik, Forcefeedback, virtuelle Realität, augmentierte Realität

**Kurzzusammenfassung**

Die Entwicklung im Bereich der virtuellen Realität ist in den letzten Jahren stark weiterentwickelt worden. Neben den VR-Brillen entwickeln sich auch die Geräte für haptisches Feedback ständig weiter. Dabei entwickeln die Hersteller dieser Feedback-Geräte Schnittstellen, welche auf die entsprechende Anwendung abgestimmt werden und keine Rücksicht auf Kompatibilität und Austauschbarkeit nehmen. Diese Arbeit untersucht zunächst verschiedene Ansätze zur Erzeugung von haptischem Feedback um daraus eine generalisierende Schnittstelle zur Anbindung an VR und AR Anwendung zu entwickeln

**Jan Stieglitz**

**Title of the paper**

Framework for the integration of hardware devices for haptic feedback in AR and VR applications

**Keywords**

Framework, haptic, feedback, virtual-reality, augmented-reality

**Abstract**

The development in the field of virtual reality has been greatly developed in recent years. In addition to the VR glasses, the devices for haptic feedback are constantly evolving. The manufacturers of these feedback devices develop interfaces, which are adapted to the respective application and do not take any consideration for compatibility and interchangeability. This work examines various approaches to the generation of haptic feedback in order to develop a generalizing interface for linking to VR and AR applications

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	2
1.2	Zielsetzung . . . . .	2
1.3	Gliederung . . . . .	3
<b>2</b>	<b>Stand der Forschung und Entwicklung</b>	<b>4</b>
2.1	Physiologie . . . . .	4
2.1.1	physiologische Grundlagen . . . . .	4
2.1.2	Taktiler Feedback . . . . .	5
2.1.3	Force Feedback . . . . .	6
2.1.4	Semantisches Feedback . . . . .	6
2.2	Beispielapplikationen . . . . .	6
2.2.1	Stationäre Lösungen . . . . .	7
2.2.2	Mobile Lösungen . . . . .	10
2.3	Handmodell . . . . .	11
2.4	Immersion . . . . .	12
2.4.1	Echtzeitaspekt . . . . .	12
2.4.2	Geringere Fehlerrate . . . . .	13
2.5	Frameworks mit Multideviceunterstützung . . . . .	13
2.6	Zusammenfassung . . . . .	13
<b>3</b>	<b>Anforderung</b>	<b>15</b>
3.1	Projektstruktur . . . . .	15
3.2	Framework . . . . .	15
3.2.1	Funktionale Anforderungen . . . . .	15
3.2.2	Plattformunabhängigkeit . . . . .	16
3.2.3	Kommunikation . . . . .	16
3.2.4	Echtzeitanforderung . . . . .	16
3.2.5	Handmodell abstraktion . . . . .	16
3.3	Hardware . . . . .	17
3.4	Zusammenfassung . . . . .	17
<b>4</b>	<b>Entwurf</b>	<b>18</b>
4.1	Framework . . . . .	18
4.1.1	Software-Architektur . . . . .	18
4.1.2	Handmodell . . . . .	19

4.1.3	Schnittstellen . . . . .	20
4.1.4	Komponenten . . . . .	21
4.2	Hardware . . . . .	22
4.2.1	Plattform . . . . .	23
4.2.2	Aktuatoren . . . . .	23
4.2.3	Kommunikation . . . . .	25
4.3	Zusammenfassung . . . . .	26
<b>5</b>	<b>Prototypische Umsetzung</b>	<b>27</b>
5.1	Framework . . . . .	27
5.1.1	Steuersoftware . . . . .	27
5.2	Hardware . . . . .	28
5.2.1	Elektronik . . . . .	28
5.2.2	Firmware . . . . .	29
5.3	Demoapplikation . . . . .	30
5.3.1	Testapplikationen . . . . .	30
5.3.2	Tracking . . . . .	30
5.3.3	DemoApp . . . . .	31
5.3.4	Zusammenfassung . . . . .	31
<b>6</b>	<b>Zusammenfassung und Ausblick</b>	<b>32</b>
6.1	Ausblick . . . . .	32

# 1 Einleitung

Mit Voranschreiten der technischen Entwicklung werden regelmäßig vergangene Entwicklungen erneut aufgegriffen und unter dem Einfluss der weiterentwickelten Technologie wieder aufgelegt. Mit stark weiter entwickelten Display-Technologien die sich, vor allem durch die Weiterentwicklung der Smartphones getrieben, in Qualität und Leistungsfähigkeit sehr stark entwickelten, wie auch durch immer stärkere Prozessoren, erlebt die Virtuelle-Realität ihren Massendurchbruch als ein Comeback. Die Miniaturisierung lässt jedoch die ersten Systeme aus den frühen 1990er Jahren alt aussehen. Smartphones lassen sich heute schon mit einfachen Mitteln zu einer VR-Brille umrüsten. Nach und nach kommen auch immer leistungsstärkere Systeme mit verbesserter Sensorik und optischer Darstellung auf den Markt.

Neben der Spieleindustrie wächst auch in der Industrie und im Service ein Interesse an diesen Systemen. Zum Beispiel bei der Montage oder Reparatur von Geräten, um dem Arbeiter die notwendigen Handgriffe zu zeigen. Dabei werden dann Systeme für augmentierte (erweiterte) Realität eingesetzt, die dem Anwender durch eine transparentes Display oder einen halbtransparenten Spiegel visuelle Informationen in das Sichtfeld einblenden.

Für die Interaktion mit der virtuellen Welt ist jedoch längst kein Joystick mehr nötig, denn mittels Handtracking kann der Nutzer schon direkt mit der Simulation interagieren. Durch das Bewegen der Hand in Form spezieller Gesten, kann der Anwender dem Computer Befehle geben, um damit in einem Spiel beispielsweise eine Tür zu öffnen. Damit die Handlung auch begreifbar wird, werden Geräte für haptisches Feedback entwickelt. Der Nutzer taucht durch dieses Feedback tiefer in die virtuelle Realität ab, was als Immersion bezeichnet wird. Die Wahrnehmung für den realen Ort tritt in den Hintergrund und wird vom Anwender nicht mehr so stark wahrgenommen. Dafür richtet sich die Aufmerksamkeit verstärkt auf die Simulation.

Es gibt bereits Geräte, die ein solches Feedback erzeugen, jedoch sind deren Schnittstellen herstellerabhängig, deshalb ist generisches Interface Wünschenswert. Hierbei stellt sich die Frage, ob es möglich ist, für verschiedene Systeme eine gemeinsame abstrakte Schnittstelle zu entwickeln. Dies soll im folgenden genauer analysiert und dargelegt werden.

### 1.1 Motivation

Eine der wichtigsten Organe des Menschen sind die Hände, denn sie haben die Fähigkeit, filigrane Werkzeuge zu bauen und diese einzusetzen. Der damit verbundene, für die Hände wichtigster Sinn ist der Tastsinn, welcher dem Menschen Informationen über den berührten Gegenstand oder die Oberfläche gibt. Das sprichwörtliche Fingerspitzengefühl ermöglicht das 'blinde' Befestigen einer Schraube oder das Tippen auf einer Tastatur ohne dabei die Finger mit den Augen visuell zu koordinieren. Viele dieser Fähigkeiten werden dabei nur oberflächlich vom Gehirn kontrolliert und basieren lediglich auf dem Zusammenspiel zwischen Tastsinn und reflexartiger Rückkopplung. Vollständig ohne Eingreifen des Gehirns, agieren dabei die Schutzreflexe, welche das schlagartige Zurückziehen der Hand bei einem starken Schmerzreiz bewirken.

Um diese aus der Realität bekannten Erfahrungen in der virtuellen-Realität abzubilden, bedarf es der Nachahmung dieser Reize. Dies kann mechanisch, beispielsweise durch Vibration, oder mittels Muskel-/Nervenstimulation erfolgen. Dazu sind bereits seit vielen Jahren verschiedene Systeme auf dem Markt gebracht worden, die den unterschiedlichsten Ansätzen folgend, versuchen ein haptisches Erlebnis für den Nutzer abzubilden.

Ein haptisches Feedback ist aber nicht nur aus der Anwendungsergonomie wünschenswert, sondern dient auch der Verbesserung von Fehlerquote und Geschwindigkeit zum Beispiel bei der Eingabe in ein elektronisches Gerät, wie die virtuelle Tastatur eines Smartphones, welche durch ein vom Vibrationsalarm ausgelöstes fühlbares "Klicken" bei der Betätigung einer Taste fühlbar macht. Hier kann der Nutzer auf die visuelle Kontrolle der Eingabe zum Teil verzichten, da eine Taste scheinbar gedrückt wurde, und damit nur noch das Zielen auf jene Taste verbleibt.

Damit zeichnen sich zwei Ziele ab, die durch ein haptisches Feedback bei VR/AR-Anwendungen erreicht werden können: Zum einen die Immersion, also die Realitätsnähe bei der Erfahrung von Virtueller Realität. Auf der anderen Seite eine geringere Fehlerrate sowie eine gleichzeitig höhere Geschwindigkeit im Arbeiten mit VR/AR-Systemen.

### 1.2 Zielsetzung

In dieser Arbeit soll ein Multi-Device-Framework zur Anbindung von Geräten für haptisches Feedback entwickelt werden. Dazu wird ein Handmodell erarbeitet, mit dessen Hilfe die Feedbackdaten aus der Anwendung auf die anzusteuernde Hardware abgebildet werden kann. Mittels eines vereinheitlichenden Handmodells sollen sowohl unterschiedliche Geräte, ihren Möglichkeiten entsprechend, angebunden werden können, als auch eine einheitliche Schnittstelle für die VR-Anwendung geschaffen werden.

Die Entwicklung als Framework bietet dabei den Vorteil, dass die Wahl eines Geräts vollständig von der Einbindung des Frameworks in eine VR-Anwendung unabhängig ist. Abhängigkeiten auf beiden Seiten werden durch die jeweilige Schnittstelle abstrahiert und eine lose Kopplung von Anwendung und Hardware erreicht.

### 1.3 Gliederung

Zunächst wird einleitend eine Übersicht auf die Thematik des Haptischen Feedbacks gegeben. Hier wird der aktuelle Stand der Forschung und die derzeitigen Einsatzgebiete von künstlich erzeugter Haptik aufgezeigt. Auf die vor allem aus 2.1 und 2.3 abzuleitenden Anforderungen und Eigenschaften, die durch das Framework zu erfüllen sind, wird im Kapitel 3 eingegangen. Anhand dieser Vorgaben wird im Kapitel 4, der Entwurf des Frameworks und dessen Architektur behandelt. Darauffolgend zeigt ein Proof of Concept im Kapitel 5 die Ergebnisse der Entwicklungsarbeit an einem Demonstrationsaufbau.

## 2 Stand der Forschung und Entwicklung

Dieses Kapitel befasst sich mit den Grundlagen der Haptischen Wahrnehmung und den derzeitigen Einsatzgebieten von taktilem Feedback. Im ersten Teilkapitel 2.1 werden die physiologischen Grundlagen für Haptisches Feedback thematisiert. Am Beispiel von existierenden Anwendungen, die mittels haptischem Feedback eine verbesserte Kommunikation zwischen Mensch und Maschine ermöglichen, werden im Folgenden Abschnitt 2.2 die Vorteile von haptischem Feedback in VR/AR-Systemen gezeigt. Folgend wird in Abschnitt 2.4 die Bedeutung von Haptik, in der virtuellen Realität für die Immersion und den Einfluss auf die Fehlerrate und Arbeitsgeschwindigkeit der Nutzer, geklärt. Im Abschnitt 2.5 wird am auf die Vorzüge und Eigenschaften von Multi-Device-Frameworks eingegangen.

### 2.1 Physiologie

Haptik beziehungsweise die haptische Wahrnehmung leitet sich vom griechischen *haptós* = 'fühlbar' ab und bezeichnet die Wahrnehmung durch den Tastsinn. Die Haptik beschäftigt sich mit den Themen um den Tastsinn wie das Greifen, Berühren oder die Wahrnehmung über den motorischen Apparat des Menschen. Wenn diese Sensorik gezielt durch technische Systeme von außen manipuliert wird, um den Eindruck einer Berührung oder einer wirkenden Kraft zu erzeugen, spricht man von haptischem Feedback. Im Folgenden wird auf die Physiologischen Grundlagen der Sensorik eingegangen und die daraus abgeleitete Einordnung von haptischem Feedback.

#### 2.1.1 physiologische Grundlagen

Die haptische Wahrnehmung des Menschen ist in drei Organgruppen gegliedert: Mechanosensoren, Propriozeptoren und Vestibularorgane. Diese Organe liefern dem Körper fortlaufend Informationen über Berührungen, Schmerzen, Lage und Bewegung bzw. die Ausrichtung der Gelenke. Dabei können aus einem Reiz heraus direkte Reflexreaktionen ohne das Einwirken des Gehirns ablaufen, wie zum Beispiel das Gehen, welches primär auf Verarbeitung und Signalisierung durch das untere Rückenmark zurückgreift.

### **Mechanosensoren**

Die Oberflächensensibilität der Haut wird durch verschiedene Sinneszellen der Gruppe der Mechanosensoren, gebildet. Diese liefern Informationen über auftretende Kräfte durch Berührungen und die dadurch im Hautgewebe auftretenden Verformungen. In Kombination mit einer ausgeführten Bewegung liefert der Tastsinn eine hinreichend präzise Auflösung. Um Aktionen ohne die visuelle Rückmeldung der Augen durchzuführen. Die auf den Tastsinn spezialisierten Nervenzellen sind in einer hohen Anzahl in den Fingerspitzen zu finden, wodurch dort eine hohe Sensibilität erreicht wird.

### **Propriozeptoren**

Gelenkstellung sowie Muskel- und Skelettkräfte werden durch die Gruppe der Propriozeptoren aufgenommen. Die zwei wichtigsten Typen sind dabei die Golgie-Sehnenorgane und die Muskelspindeln. Das Golgie-Sehnenorgan, welches am Sehnenansatz des Muskels liegt, liefert über dessen Dehnung eine Information über die vom Muskel aufgebrachte Kraft. Die Längenänderung des Muskels wird hingegen von der Muskelspindel zurückgemeldet. (Meroth und Tolg, 2008, 116 f)

#### **2.1.2 Taktiles Feedback**

Das taktile Feedback wird eingesetzt um Berührungen von Oberflächen darzustellen. Diese Methode setzt gezielt bei der Reizung der Mechanosensoren an, indem direkt oder indirekt, zum Beispiel mittels Schallwellen, die Oberfläche der Haut gereizt wird. Diese auch später als physisches Feedback bezeichnete Haptik, kann sowohl die Geometrie von Objekten in VR-Anwendungen als auch deren Oberflächenstruktur erfassbar machen.

#### **Vibrotaktiler Feedback**

Eine Besonderheit bildet das vibrotaktile Feedback, welches nicht die eigentliche Haptik einer Berührung nachbildet, sondern mittels Vibration die Assoziation an ein aus der Realität bekanntes weckt. Die meist durch kleine Aktuatoren realisierbare haptische Wirkung, bietet eine weniger hohe Präzision in der Wahrnehmung. Im Gegenzug dafür sind die benötigten Mittel für die technische Umsetzung schon sehr weit miniaturisiert und sehr einfach anzusteuern. (Heinecke, 2012, 159 ff)

### **2.1.3 Force Feedback**

Force Feedback bezeichnet eine Methode des haptischen Feedbacks, welche eine direkte physische Gegenkraft dem Nutzer entgegenbringt. Dabei wird, zum Beispiel anhand eines Modells, die Kraft errechnet, die dem Nutzer durch das virtuelle Objekt entgegengesetzt wird. Durch das hohe Auflösungsvermögen Propriozeptoren für kleine Kraft- und Positionsänderungen, ist bei dem Einsatz von Force-Feedback eine hohe Aktualisierungsrate im Bereich ab 500 Hz notwendig. Dies ist für VR und AR-Anwendungen im Vergleich zur Bildwiederholfrequenz und den darüber angesprochenen Seh-Sinn eine über zehn mal höhere Aktualisierungsrate. (? , 318)

### **2.1.4 Semantisches Feedback**

Das semantische Feedback bildet für den Anwender Metainformationen aus der ausgeführten Interaktion ab. Als Beispiel kann hier die virtuelle Tastatur von mobilen Touchgeräten genannt werden. Durch das Betätigen einer angezeigten Schaltfläche wird dabei ein kurzer Impuls mittels des für den Vibrationsalarms zuständigen Motors erzeugt. Dies bietet dem Nutzer eine Assoziation, welche an die Anschlagdynamik einer Tastatur erinnert. Ebenso wird bei anderen GUI-Elementen versucht, eine aus der Realität bekannte Betätigungshaptik nachzustellen, wie etwa die Rasten eines Mausekkrads. Das semantische Feedback kann auch zur zusätzlichen Orientierung bei Interaktionen mit Reglern hinzugefügt werden, um ein scheinbares Gefühl für die Änderung der einzustellenden Werte zu erzeugen.

## **2.2 Beispielapplikationen**

Die bereits bestehenden Lösungen werden nachfolgend dargestellt, dazu werden sie ihrer Möglichkeit zum mobilen Einsatz nach in zwei Klassen unterteilt, stationäre und mobile Lösungen. Die stationären Lösungen zeichnen sich durch ihre lokale Bindung an einen Ort aus. Im Vergleich zu den mobilen Lösungen bieten diese Systeme eine höhere Standsicherheit und können potentiell mit höheren Kräften agieren als ein Handheld Device. Stationäre Feedback-Geräte eignen sich dadurch primär für Tangibles und Force-Feedback. Einige Geräte setzen in ihrer Funktionsweise auf die Berührung des Benutzers und binden damit die Interaktion lokal an das Gerät und dessen physikalische Grenzen. Ein weiterer Ansatz ist in der Mid-Air-Haptik von Ultrahaptics lässt dem Benutzer einen definierten Raum in der Nähe des Geräts zur Erfahrung von haptischem Feedback.

Im Gegensatz zu den Stationären Lösungen, bieten die Mobilen Geräte eine vom Arbeitsbereich ungebundene Verwendung. Die auch als Wearables bezeichneten Feedback-Devices erlauben dem Benutzer die freie Bewegung im Raum, sind aber dafür auf die Leistungsfähigkeit der mobilen Energieversorgung, meist Akkus, angewiesen. Neben der Stromversorgung muss auch die Realisierung des Feedbacks soweit miniaturisierbar sein, sodass ein Gerät auch wirklich tragbar ist und nicht die Bewegung des Nutzers einschränkt.

Das zu erreichende Feedback ist bei mobilen Systemen darüber hinaus meist auf die taktile Wahrnehmung durch die mechanosensorischen Zellen beschränkt. So kann die Haptische Empfindung nur passiv durch das Handtracking und über die Simulation erfolgen, im Vergleich dazu können die stationären Systeme ein aktives ertasten ermöglichen. [Miguel A. Otaduy u. a. \(2016\)](#)

### 2.2.1 Stationäre Lösungen

#### Particle Jamming Arrays

Dieses Gerät verbindet Force- und taktiles Feedback um dem Nutzer eine haptische Erfahrung von 3D-Objekten zu liefern. Durch einen erhöhten Druck bei der Berührung soll auch eine Eingabe durch das Gerät möglich sein. Die Oberfläche besteht hier aus einer in Zellen

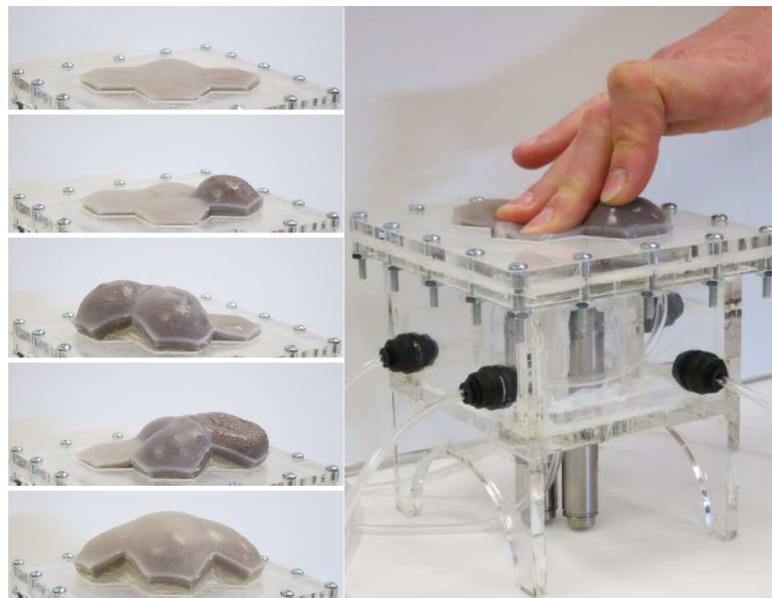


Abbildung 2.1: Modellaufbau mit vier Zellen, Quelle: Stanford University, <http://charm.stanford.edu/Main/DesignAndControlOfHapticSystems>

strukturierten weichen Membran. Diese wird durch den Einsatz von Über- oder Unterdruck und zusätzlichen Pneumatikzylindern deformiert. Durch diese Kombination ist es möglich die Oberfläche zu formen und die Steifigkeit der individuellen Zellen zu kontrollieren. In der Abbildung 2.1 ist ein Versuchsaufbau mit vier Zellen zu sehen. Die Darstellungen auf der linken Seite zeigen verschiedene darstellbare Oberflächen, die durch das Gerät abgebildet werden können. Durch die Größe des Aufbaus und die bisher geringe Reaktionsgeschwindigkeit, eignet sich dieser Ansatz so nicht für den mobilen Einsatz. Durch eine starke Miniaturisierung der Technik, könnte dieses System jedoch für mobile Geräte eine bessere Darstellung von Material- und Oberflächeneigenschaften bieten.

### Phantom

Dieses Gerät besteht aus einem Stift, der durch einen Arm gehalten wird und sich in 6 Freiheitsgraden bewegen lässt. In den Gelenken des Arms sind Motoren eingebaut, die der Bewegung des Nutzers eine Kraft entgegensetzen können. Mit dem Stift kann sowohl ein Forcefeedback gegeben, als auch 3D-Eingaben vom Benutzer ausgeführt werden. Dabei muss die Aktualisierung der Kräfte für das Feedback mit mindestens 1000 Hz geschehen, damit die Kräfte durch den Anwender als kontinuierlich und realistisch wahrgenommen werden. An dem in Abbildung 2.2



Abbildung 2.2: Phantom, Ein 3D Eingabestift mit haptischem Feedback, Quelle: <http://www.geomagic.com/de/products-landing-pages/haptic/>

gezeigten Gerät ist der Aufbau mit dem Eingabestift und Tragarm sowie dessen Handhabung gut zu erkennen. Der Vorteil dieser Umsetzung ist die direkte Kopplung der Stiftbewegung an

die resultierende Gegenkraft, durch die eine gute Darstellung von Oberflächenstruktur und Materialeigenschaften realisiert werden kann. Ein semantisches Feedback ist hierbei nicht vorgesehen. Dies ließe sich aber durch die vorhandenen Aktuatoren verwirklichen. Jedoch lässt sich dieses Prinzip nicht für die mobile Nutzung umsetzen. [McNeely u. a. \(1999\)](#)

### Ultrahaptics

Mittels Ultraschall wird bei dem Ansatz von Ultrahaptics ein taktiler Feedback erzeugt. Dazu wird von einem Array aus Ultraschalllautsprechern eine Druckänderung an der Handoberfläche des Benutzers erzeugt. Dies geschieht durch die Interferenz von Schallwellen, die von den Lautsprechern ausgesendet werden. Durch den Phasenunterschied, mit dem die Signale an die verschiedenen Lautsprecher übertragen und dort in Schallwellen umgewandelt werden, überlagern sich diese linienförmig oder an einzelnen Punkten.

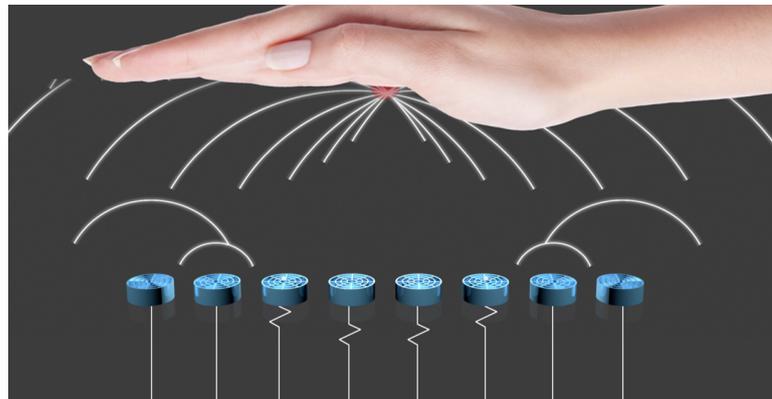


Abbildung 2.3: Schematische Darstellung des Funktionsprinzips durch die Überlagerung von Schallwellen, Quelle: [\(Miguel A. Otaduy u. a., 2016, 94\)](#)

Das Prinzip der zeitversetzten Ansteuerung und der aus Überlagerungen der Schallwellen resultierende Brennpunkt ist in [Abbildung 2.3](#) dargestellt. Dieses Gerät ist in der Auflösung durch die Schallgeschwindigkeit und der Arbeitsfrequenz von 40kHz auf eine minimale Auflösung von 8,5 mm pro Objekt beschränkt. Im Vergleich zu der in dieser Arbeit angestrebten vibrotaktilen Lösung kann dieses Gerät unabhängig von fest angebrachten Aktuatoren die Oberfläche der Haut mit vergleichsweise frei definierbaren Mustern und Formen stimulieren. Es wird zwar einen Arbeitsbereich im Abstand von 4cm bis 2m vor dem Gerät geboten, jedoch lässt sich der Aufbau nicht gut miniaturisieren und ist damit für den mobilen Einsatz ungeeignet. [\(Miguel A. Otaduy u. a., 2016, 92-123\)](#)

## 2.2.2 Mobile Lösungen

### Smartphone

Eines der am meisten verwendeten Geräte, die ein haptisches Feedback liefern, ist das Smartphone. Der in den Vorläufern, den einfachen Mobiltelefonen, eingebaute Vibrationsalarm diente zunächst dem stummen Signalisieren eines Anrufs. Dies wurden auch für Rückmeldungen der simplen Spiele auf den Geräten verwendet, um Ereignisse, wie Gewinn oder Spielende, haptisch zu untermalen.

Mit der Entwicklung des Smartphones begann die Bedeutung des haptischen Feedbacks zunehmend wichtiger zu werden. Bei einer, mittels Finger, auf der glatten Touchoberfläche getätigten Eingabe auf einer virtuellen Tastatur wird der Nutzer zwar eine Berührung wahrnehmen, jedoch fehlt die Haptik einer realen Taste. Das Durchbrechen einer Taste kann hier durch einen kurzen Impuls an den Vibrationsmotor für ein ausreichendes Feedback und die einhergehende Assoziation zum bekannten Tastendruck leisten.

Neben der Simulation von bekannten haptischen Erlebnissen aus der realen Welt, werden durch das semantische Feedback auch neue Metaphern für die Benutzeroberflächen geschaffen. Diese Funktionen sind zum Teil integriert oder lassen sich durch den Entwickler einsetzen. So können auf der Skala eines Sliders bestimmte Werte visuell markiert und gleichzeitig erföhl- bzw. begreifbar gemacht werden.

### 3-DoF Taktiles Rendering

Eine Möglichkeit Forcefeedback als mobile Lösung zu realisieren stammt von der URJC Madrid, die im Rahmen eines Talks auf der Siggraph 2016 gezeigt wurde. Das Ziel des Gerätes ist

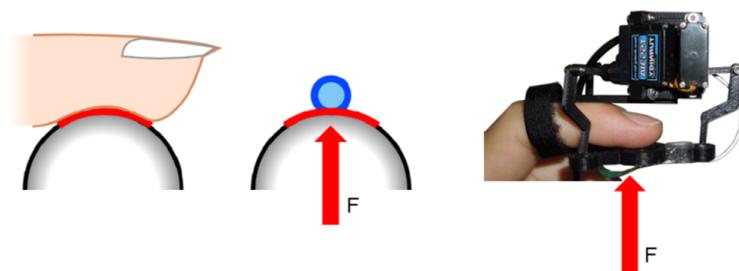


Abbildung 2.4: Darstellung eines Geräts zur Realisierung von gerichtetem taktilem Feedback, Quelle: URJC Madrid/Siggraph '16, (Miguel A. Otaduy u. a., 2016, 31)

die Abbildung der Oberflächenausrichtung des virtuellen Objektes am Finger. Dazu wird

mittels Servo-Motoren eine Kunststoffplatte der Ausrichtung der virtuell berührten Oberfläche entsprechend ausgerichtet und dann an die Fingerspitze gedrückt.

In der linken Darstellung von Abbildung 2.4 wird die Situation in der virtuellen Realität dargestellt. Die in der Mitte des Bildes dargestellte durch die VR berechnete Kraft wird im rechten Teil durch die am Finger montierten Motoren auf diesen übertragen. Dieses Konzept kann die Richtung der Kraft darstellen, Trotz der mobilen Einsetzbarkeit der bisherigen Bauweise, ist aufgrund der Größe des Gerätes die Bewegungsfreiheit stark eingeschränkt. Darüber hinaus ist mit einem optischen Handtracking, wie der Leap-Motion, die Erfassung der Hand nicht mehr möglich, da die Sicht auf diese optisch verbaut ist.

### Wolverine

Das an der Stanford University entwickelte Konzept mit dem Namen Wolverine soll das wortwörtliche Begreifen von Gegenständen in der virtuellen Realität ermöglichen. In der

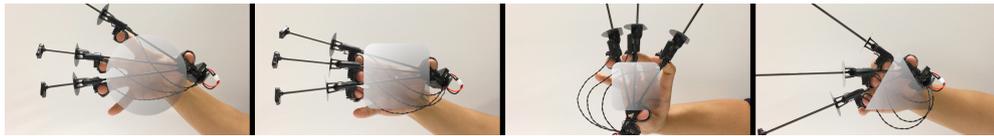


Abbildung 2.5: Wolverine, ein ein haptisches Interface zum Greifen in der VR, Quelle: [Choi und Follmer \(2016\)](#)

Abbildung 2.5 ist die Anwendung des Feedbacksystems mit eingeblendeten, virtuell gegriffenen Objekten zu sehen. Das Gerät besteht aus drei Stangen, die flexibel über ein Gelenk verbunden sind. Zeige-, Mittel- und Ringfinger werden durch Schleifen an Gleitern auf den Stangen flexibel gehalten. Der Daumen ist am zentralen Gelenk der Stangen fixiert. Zur Simulation einer Gegenkraft wird der Wagen auf seiner Bewegungsachse gebremst. Durch die Bremse baut die Kraft des Fingers eine Kraft auf die Stange des Gleiters und damit auf den Daumen aus. Aufgrund der einrastenden Bremse kann nur eine vollständige Arretierung erreicht werden. Daher ist es mit diesem Device nur möglich feste Oberflächen mit immer gleicher Elastizität darzustellen. Über diese Greif-Metapher hinaus ist keine weitere Feedbackform möglich.

## 2.3 Handmodell

In den vorgestellten Applikationen wird für einige Geräte ein Handmodell verwendet, welches die Abbildung zwischen der in der virtuellen Realität ablaufenden Berührungen auf die Hardware überträgt. Die Handmodelle werden hier in unterschiedliche Feedbackzonen gegliedert. Jede

Zone kann entsprechend der angeschlossenen Geräte verschieden haptische Informationen an den Nutzer übermitteln.

Zunächst fällt auf, dass die genannten stationären Lösungen ihr Feedback durch die Änderung ihrer physikalischen Eigenschaften realisieren. Durch Form oder Verhaltensänderung bieten sie dem Nutzer die Möglichkeit durch aktives Berühren und Begreifen die dargestellte Haptik zu erleben.

Das Smartphone welches als Handheld-Device durch seine Vibration als ganzes nur ein semantisches Feedback liefert, setzt nicht gezielt ein Handmodell ein, jedoch durch die Bedienung mittels der Fingerspitze kann hier ein Handmodell mit einer Feedbackzone vorrangig am Zeigefinger ausgegangen werden. Die beiden anderen mobilen Systeme setzen auch am Fingerspitzengefühl an und bilden die Hand in ihrem Model über diese ab. Bei der Greifhaptik von Wolverine sind diesen Fingerspitzen letztendlich nur Bits zugeordnet, welche das Aktivieren und Lösen der Bremse kontrollieren. Im Kontrast dazu zeigt die Lösung aus Madrid eine dreidimensionale Ausrichtung und Intensität auf.

## **2.4 Immersion**

Immersion wird der Grad des Eintauchens eines Nutzers in eine virtuelle Realität genannt. Neben einer hohen Auflösung bei der Darstellung und entsprechen hohen Bildwiederholungsrate, kann haptisches Feedback die virtuelle Realität begreifbar gemacht werden. Wichtig für eine hohe Immersion ist die Plausibilität der wahrgenommenen Reize. Dabei muss die Darstellung zu der Wahrnehmung in ihrer Intensität und sowohl logisch als auch zeitlich passen.

Negativ wirken sich zum Beispiel eine mit der VR-Brille ausgeführte Kopfbewegung aus, wenn die dargestellte Welt sich entgegen dieser Bewegung verschiebt. (Dörner u. a., 2013, 198)

### **2.4.1 Echtzeitaspekt**

Zur Immersion ähnlich verhält es sich beim haptischen Feedback. Wie schon bei dem Gerät Phantom angesprochen, muss die Interaktion des Nutzers bei Force-Feedback-Geräten mit einer hohen zeitlichen Auflösung zurückgemeldet werden, damit die Handlung und das Feedback als zusammengehörig erkannt und wahrgenommen wird.

Dabei gilt auch für die Verwendung von mobilen Geräten in Kombination mit VR-Brillen. Jedoch muss für das hier meist verwendete taktile Feedback besonders die visuelle und haptische Darstellung als zeitgleich empfunden werden. Da die Anzeige selten mehr als 100 Bilder/s darstellen kann, sind Latenzen von unter 10ms vollkommen ausreichend.

### 2.4.2 Geringere Fehlerrate

Nach [Yang u. a. \(2011\)](#) verbessert sich durch den gezielten Einsatz von haptischem Feedback die Genauigkeit und die Geschwindigkeit der ausgeführten Tätigkeit. In ihrer Untersuchung wurde ein Touchscreen mit Forcefeedback ausgerüstet um die Probanden bei der Benutzung einer grafischen Oberfläche mittels Rückmeldung zu unterstützen. Die Tests ergaben einem 50% geringere Fehlerrate und eine dabei leicht erhöhte Geschwindigkeit in der die Aufgaben absolviert wurden.

## 2.5 Frameworks mit Multideviceunterstützung

Da in den Abschnitten [2.2](#) und [2.3](#) verschiedene Lösungsansätze mit gleichen Zielen und ähnlichen Strukturen dargelegt werden konnten, ist eine vereinheitlichte Ansteuerung durch ein Multi-Device-Framework sinnvoll. Durch eine generalisierte Eingabe kann die Applikation unabhängig von existierenden und zukünftigen Lösungen realisiert werden.

Eine entsprechende Entwicklung für die Geräte unabhängige Eingabe von Gesten und deren Darstellung als generalisierte digitale Repräsentation wurde bereits in [Kobayashi u. a. \(2005\)](#) beschrieben. Auf der Ausgabeseite ist noch nicht ersichtlich, dass eine generalisierende Lösung kommt. In [Abschnitt 2.2](#) sind verschieden brauchbare Lösungen und Ansätze zu sehen, die durch ein vereinheitlichtes Interface profitieren können.

## 2.6 Zusammenfassung

Wie in diesem Kapitel zu sehen ist, existieren schon verschiedene Lösungen für haptisches Feedback, die sich je nach Anwendungsfall mehr oder weniger gut im Kontext von AR und VR-Anwendungen nutzen lassen. Auch Strategien zum generalisierten Abbilden von hardware-spezifischen Schnittstellen wurden bereits beschrieben und realisiert. Jedoch leistet weder die Hardware, noch existiert eine entsprechende Software, die den im [Kapitel 1](#) geforderten Ansätze vollständig umsetzt.

Die meisten Hardwaresysteme fokussieren sich auf ein gesondertes Ziel, welches meist nur indirekt mit einer Anwendung im Zusammenhang mit der virtuellen Realität steht. Andere sind auf eine spezielle Aufgabe und dem dafür benötigten Feedback ausgelegt und lassen einen offeneren Nutzungsrahmen nur schwer zu. Auf Seiten der Hardwareanbindung ist leider nur die für jedes Gerät passend zugeschnittene Schnittstelle zu finden. Ein hardwareunabhängiges Framework, welches eine generalisierte Ansteuerung von Feedbackhardware erlaubt, konnte nicht gefunden werden.

In dieser Arbeit sollen aus diesen Gründen eine Lösung aus zwei Systemen bestehend entwickelt werden. Zunächst ein Multi-Device-Framework, welches die Ansteuerung von Haptik-Feedback-Devices mittels einer generalisierten und hardwareunabhängigen Schnittstelle ermöglicht. Zudem wird zur Verifizierung, für ein Proof of Concept, ein Gerät für haptisches Feedback entwickelt. Dieses Feedback-Device soll mittels des zuvor entwickelten Frameworks dessen Funktion und Leistungsfähigkeit aufzeigen.

## 3 Anforderung

In dieser Arbeit soll eine Lösung für mobiles haptisches Feedback erarbeitet und umgesetzt werden, welche sowohl physiologisches als auch semantisches Feedback vermittelt. Unabhängig von der Hard- und Softwareplattform der VR-Anwendung, soll eine Anbindung an verschiedene existierende Geräte für die Vermittlung von haptischem Feedback ermöglicht werden. Zunächst wird die Struktur der zu entwickelnden Lösung in Abschnitt 3.1 vorgestellt. Die Anforderung an beide Projektteile folgen in 3.2 und 3.3.

### 3.1 Projektstruktur

Die Lösung soll dabei in zwei separat zu entwickelnden Teilen erstellt werden. Im ersten Teilprojekt im Unterkapitel 3.2 soll das zu entwickelnde Framework als Schnittstelle zwischen einer VR-Anwendung (Eingangsseite) und der einzusetzenden Feedback-Hardware (Ausgangsseite) dienen. Dabei soll das Framework die verwendete Hardware gegenüber der Eingangsseite abstrahieren und dadurch transparent werden lassen. Besonderheiten der Hardware sollen so durch das Framework gekapselt werden, dass diese für die Einbindung auf der Eingangsseite keine zusätzlichen Abhängigkeiten erfordern.

Im zweiten Teilprojekt im Abschnitt 3.3 werden die Anforderungen an einen Proof of Concept gezeigt, der mittels einer konkreten Hardware als prototypisches Beispiel für eine mobile Lösung realisiert werden soll. Das Gerät soll dabei preisgünstig und kompakt aufgebaut sein und sowohl das semantische, als auch ein physiologisches Feedback liefern. Die Hardware-schnittstelle ist dabei explizit unabhängig vom Framework zu wählen, um dessen Leistung darstellen zu können.

### 3.2 Framework

#### 3.2.1 Funktionale Anforderungen

Das Framework soll sich in die bestehende und zukünftige VR/AR-Landschaft einfügen lassen. Dazu ist es notwendig, dass die Schnittstellen zur Eingangsseite als auch zur Ausgangsseite

möglichst generisch gestaltet werden, um eine geringe Einschränkungen in der Verwendbarkeit zu erreichen. Unabhängig von der angeschlossenen Hardware muss die Eingangsschnittstelle gegenüber der verwendenden Applikation immer die gleiche Gestalt haben. Daraus ergibt sich, dass die hardwareseitigen Anforderungen bei der Verwendung des Frameworks durch dieses gekapselt und abstrahiert werden müssen.

#### **3.2.2 Plattformunabhängigkeit**

Viele Anwendungen werden heute nicht mehr nur gezielt für eine Rechnerplattform entwickelt. Dies gilt auch für VR-Anwendungen, welche zunehmend auf Multiplattform-tauglichen Engines wie Unity oder Xenko basierend entwickelt werden. Um dieser Entwicklung Rechnung zu tragen soll das Framework unabhängig von der unterliegenden Plattform nutzbar sein.

#### **3.2.3 Kommunikation**

Um die Kommunikation mit dem Framework für die VR-Anwendung einfach zu gestalten, soll die Kommunikation über eine Netzwerkschnittstelle erfolgen. Ebenso wie die Plattformunabhängigkeit ist die Anforderung an Geschwindigkeit und Stabilität der Lösung entscheidend. Dazu muss die Kommunikation einfach und robust aufgebaut werden. Auf die Sicherheit, gegen Manipulation von außen, wird in der Entwicklung dabei explizit nicht eingegangen, da diese Schnittstelle lediglich der Demonstration dient. Da nicht von Anfang an bekannt ist, wie die Ansteuerung der Hardware von verschiedenen Herstellern anzusteuern ist, soll das Framework sowohl die synchrone als auch asynchrone Datenübermittlung ermöglichen.

#### **3.2.4 Echtzeitanforderung**

Die Anbindung von Geräten, die haptisches Feedback erzeugen, bindet weitere Sinne in das Erleben der virtuellen Realität ein. Das Ziel ist dabei eine erhöhte Immersion, also das Ein- oder Abtauchen der Wahrnehmung in die Simulation. Damit dieser Effekt erreicht wird, muss das haptische Feedback für den Nutzer als synchron mit der ausgeführten Interaktion wahrgenommen werden. Als ein Richtwert sollte die Gesamtlatenz nicht größer als 10 ms sein.

#### **3.2.5 Handmodell abstraktion**

Das Handmodell definiert eine abstrahierte Sichtweise auf die Hand und soll für die verschiedenen existierenden Geräte eine Abbildung ermöglichen. Wie im vorherigen Kapitel unter [2.3](#) beschrieben nutzen verschiedene Feedback-Devices unterschiedlich detaillierte Handmodelle. Ein generalisierendes Handmodell muss daher hinreichend detailliert sein, um eine generische

Schnittstelle für möglichst viele, auch unbekannte, Geräte realisieren zu können. Von diesem Modell aus muss für Geräte, welche weniger detailliertes Feedback erzeugen können, eine entsprechende Übersetzung vor der Ansteuerung erfolgen. Zur Strukturierung der Daten wird das Handmodell in Feedbackzonen mit einheitlichen Parametern für das Feedback aufgeteilt.

## 3.3 Hardware

Die für den Proof of Concept entwickelte Hardware soll in der Lage sein, taktilen und semantisches Feedback zu erzeugen. Das für die Darstellung einer Berührung genutzte physiologische bzw. physische Feedback, soll die Umgebung der virtuellen Realität für den Benutzer berühr- bzw. begreifbar machen. Verschiedene Intensitäten des Feedbacks sollen dabei als Reaktion auf verschieden starke Berührungen und für Simulation unterschiedlicher Materialien dienen.

Aus der Verwendung eines Touchscreens wie bei einem Smartphone ist hingegen die unterstützende Wirkung von semantischem Feedbacks bekannt. Die von der Tastatureingabe bekannte Haptik einer gedrückten Taste, beziehungsweise das 'Klicken' bei deren Betätigung, fehlt bei einem Touchscreen, wegen der verwendeten Glasoberfläche. Dieser fehlenden Reiz wird durch einen kurzen, fühlbaren Impuls des Vibrationsalarms künstlich erzeugt. Auch dieses semantische Feedback, welches dem Nutzer die Rückmeldung einer Interaktion im Gegensatz zu einer Berührung vorspielt, soll die Hardware realisieren können. Im VR-Kontext wird das semantische Feedback derzeit vor allem durch Spiele-Controller wie die, der HTC-Vive realisiert. Der Einsatz des haptischen Feedbacks ist dabei durch die VR-Anwendung anzusteuern. (Dörner u. a., 2013, 159 ff)

## 3.4 Zusammenfassung

Die wesentlichen Anforderungen an beide Teilprojekte ist eine sehr niedrige Latenz, damit das haptische Feedback die Immersion für den Anwender erhöhen kann. Für das Framework bedeutet diese Vorgabe, dass die Verarbeitung der eingehenden Informationen, über das haptische Feedback, möglichst linear für die Weiterverarbeitung, an die Hardwaresteuerung, bereitgestellt werden muss. Die zu entwickelnde Hardware muss über die Anforderungen an geringe Latenzen auch noch den Aspekt des mobilen Einsatzes erfüllen.

Daneben ist das generalisierte Handmodell für die Umsetzung des Frameworks von entscheidender Bedeutung. Wird das Handmodell zu ausführlich gestaltet, wirkt sich dies auf die Performance und damit auch auf die Latenz aus. Ein zu gering detailliertes Modell hingegen liefert nicht genug Daten für Geräte mit hohem Darstellungsvermögen.

## 4 Entwurf

Zur Realisierung der in Kapitel 3 gestellten Anforderungen, werden zwei Teilprojekte definiert. Das erste Projekt in Kapitel 4.1 beschäftigt sich mit der Entwicklung des Multi-Device-Frameworks. Das Framework stellt die Primäre Softwarekomponente dar, welche die abstrakte Schnittstelle für eine VR-Anwendung bereitstellt.

Als zweites Projekt wird in Kapitel 4.2 ein prototypisches Gerät zur Erzeugung von physischem und haptischem Feedback entwickelt. Dieses besteht aus der Hardware und dazugehöriger Firmware, sowie einer Steuersoftware, welche durch das Framework anzusteuern ist.

### 4.1 Framework

In diesem Abschnitt wird die Konzeption und Architektur des Frameworks beschrieben. Das Ziel der Software ist es, eine nach außen einheitliche und hinreichend detaillierte Schnittstelle zu definieren, welche eine Ansteuerung von Geräten für haptisches Feedback ermöglicht. Abschnitt 4.1.1 zeigt einen Überblick über die Architektur einer VR-Anwendung mit einem über das Framework angebindenem Feedback-Device. Nachfolgend wird in 4.1.2 auf das Handmodell und dessen Eigenschaften eingegangen. Die Schnittstelle der Eingangsseite und das darüber ausgetauschte Protokoll werden in den darauffolgenden Unterkapiteln 4.1.3 behandelt. Zuletzt zeigt Abschnitt 4.1.4 die Klassenstruktur des Framework.

#### 4.1.1 Software-Architektur

Das Framework wird in Javascript (JS), auf NodeJS basierend, als Modul entwickelt. Kern der Implementation ist dabei das Handmodell, welches zusammen mit der Eingangsschnittstelle das eigentliche Framework bildet. Eine Software, die ein konkretes Feedback-Gerät ansteuern soll bindet das Framework ein und bildet damit die generische Schnittstelle für die VR-Anwendung ab. Die Struktur und der Einsatz des Frameworks wird in der Abbildung 4.1 deutlich. Als Bindeglied zwischen VR-Anwendung und der Hardwaresteuerung des Feedback-Gerätes bildet

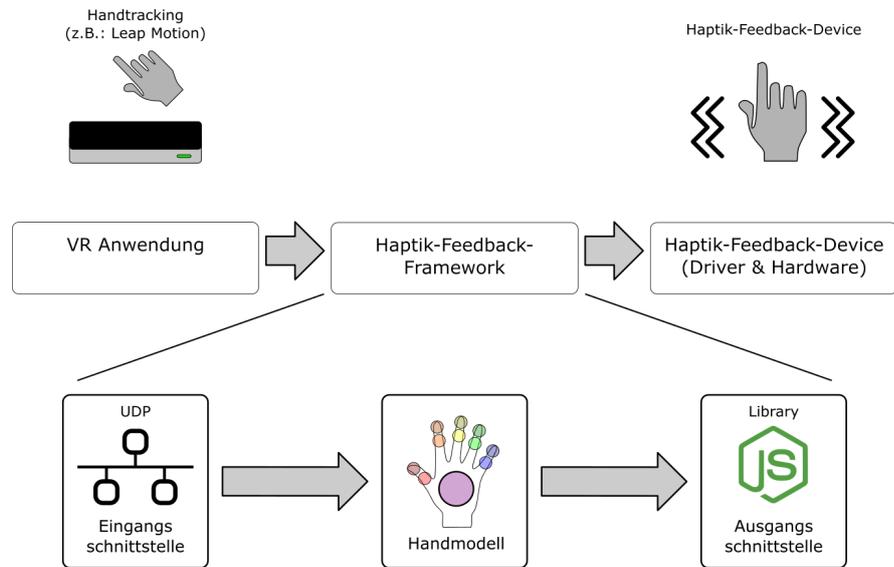


Abbildung 4.1: Übersicht über die eingesetzte Lösung, Quelle: eigenes Werk

das Framework in Richtung der Anwendung die generische Schnittstelle ab und liefert diese Daten in Form des Handmodells als eingebundenes Modul der Steuerungssoftware.

#### 4.1.2 Handmodell

Der Kern des Frameworks besteht aus einem generischen Handmodell, welches verschiedene Feedbackzonen für das physische Feedback definiert. Dieses besteht aus fünf Fingermodellen

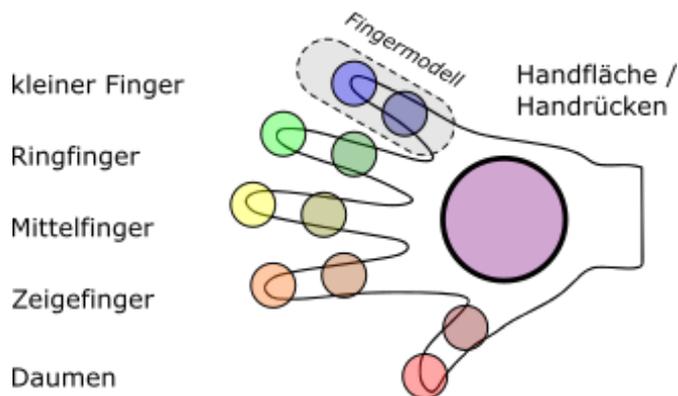


Abbildung 4.2: Darstellung des intern verwendeten Handmodells, Quelle: Eigenes Werk

und einem Modell für die Vorder- und Rückseite der Mittelhand. Letztere dient vor allem der

Rückmeldung einer Kollision, während die anderen primär für das Greifen und Ertasten von Virtuellen Objekten oder die Interaktion mit diesen dienen.

Das verwendete Handmodell aus Abbildung 4.2 orientiert sich an der Physiologie des Tastsinnes. Die Empfindlichen Fingerspitzen, sowie je eine weitere der Fingermitte zugeordnete Feedbackzone, sollen ein der Realität nachempfundenes Begreifen und ertasten ermöglichen. Die Zonen auf der Vorder- sowie Rückseite der Mittelhand sind für die Darstellung von Kollisionen mit Objekten in diesem Bereich vorgesehen. Im Gegensatz zum physiologischen Feedback, welches die Berührung der Hand an bestimmten Stellen simuliert, ist für den Einsatz des semantischen Feedbacks keine Lokalisierung notwendig. Eine Ausnahme stellt das Zuordnen der Hand im Falle des beidhändigen oder kollaborativen Arbeitens dar. Diese Zuordnung muss jedoch von der VR-Anwendung realisiert werden.

### 4.1.3 Schnittstellen

Um die Anforderungen der Plattformunabhängigkeit gegenüber der VR-Anwendung zu erfüllen wird die Kommunikation mittels einer Netzwerkschnittstelle erfolgen. Dies ermöglicht, ein Auslagern der Haptiksteuerung an einen anderen Computer, wodurch der Einsatz in Kombination mit einer VR-Anwendung auf einem Smartphone möglich ist. Um eine geringe Latenz zu erzielen, wird als Netzwerkübertragungsprotokoll UDP ausgewählt. Die dabei entfallende Übertragungssicherheit wird dabei zugunsten der höheren Nutzdurchsätze in Kauf genommen.

Ausgangseitig wird das Framework durch die Verwendung als Bibliothek in eine entsprechende Software zur Steuerung der Hardware eingebunden. Die Steuerungssoftware kann zwei verschiedene Callback-Methoden des Frameworks nutzen, einen Timer, der Zeitgesteuert eine Funktion der Steuerungssoftware aufruft, sowie eine Methode, welche nach jeder Aktualisierung durch die VR-Anwendung ausgelöst wird. Das Framework bildet also ein Messaging-System ab, das sowohl Asynchrone als auch Synchrone Verarbeitung ermöglicht. Damit wird die Hardwaresteuerung vom eingehenden Datenstrom entkoppelt. Als weiterer positiver Nebeneffekt ist es möglich, die Hardwareansteuerung durch einen anderen Computer im Netzwerk durchführen zu lassen.

### Protokoll

Das Protokoll ist als UDP-Basierter Stream entworfen, über den die Daten ausschließlich von der VR-Anwendung zum Framework transportiert werden. Dabei ist jeder Frame ein zustandsloses Paket im JSON-Format. Die Struktur der Daten ist durch das Handmodell implizit vorgegeben



Abbildung 4.3: Beispielpaket für das entworfene Protokoll, Quelle: eigenes Werk

und lässt sich durch die Implementierung in Node-JS direkt als JSON-Objekt umsetzen. Die Intensität des physischen Feedbacks wird als Gleitkomma-Wert zwischen 0.0 (keine Intensität) und 1.0 (starke bis schmerzhaft Berührung) zu übermitteln. Die Übertragung aus der VR-Anwendung erfolgt hier nach dem Fire and Forget-Prinzip, da einzelne entfallende Pakete im Vergleich zu der Latenz einer übertragungsgesicherten Kommunikation weniger schwer wiegen. Das Semantische Feedback wird durch ein symbolisches Schlüsselwort übertragen, welches die Art des Feedbacks bezeichnet. Die konkrete Umsetzung der Semantik ist von der Hardware bzw. ansteuernden Software zu Implementieren.

Die Abbildung 4.3 den Aufbau eines Datenpakets, welches mittels UDP übermittelt wird. In Gruppen, die den Fingern zugeordnet sind, werden die zwei Werte für die Intensität des taktilen Feedbacks übermittelt. Beispielhaft werden in der Grafik für die Fingerspitzen und Handinnenfläche wird jeweils der Wert 0,7 und für die Fingermitte bzw. den Handrücken ein Wert von 0,3 übermittelt.

Anders als bei dem taktilen, wird für die Übertragung des semantischen Feedbacks ein feststehender Begriff übermittelt, der einen logischen Namen für das Feedback beschreibt. Des weiteren muss sichergestellt werden, dass trotz der unsicheren Übertragung mittels UDP, das semantische Feedback ausgeführt wird. Dazu wird die Information in mehreren aufeinanderfolgenden Paketen übermittelt. Damit das angeforderte Feedback nicht durch verspätet eintreffende Pakete wiederholt von der Hardware ausgeführt wird, ist in der Struktur neben dem Typen-Feld auch die Übermittlung eines Tokens notwendig. Das semantische Feedback wird dadurch pro Token genau einmal dargestellt.

#### 4.1.4 Komponenten

Aufgrund der hohen Anforderungen an die Verarbeitungsgeschwindigkeit ist das Framework schlank entworfen worden. Die Eingangsseite ist als Netzwerkschnittstelle mit den in der Kapitel 4.1.3 beschriebenen Protokoll implementiert. Die zweite Komponente bildet die Imple-

mentierung des Handmodells, welches durch die Eingangsschnittstelle von der verwendenden VR-Anwendung die Feedbackdaten erhält. Das Framework selbst wird als Komponente in die Umgebende Steuerungssoftware für die angeschlossenen Feedback-Hardware eingebunden. Das Framework besteht aus einer Klasse, welche die Kommunikation auf der Eingangsseite

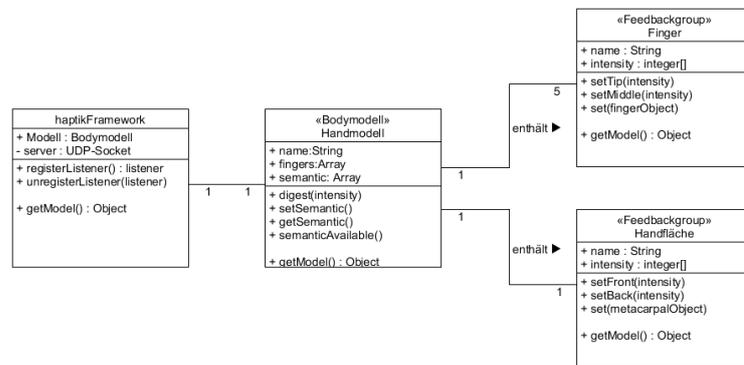


Abbildung 4.4: Komponentendiagramm des Frameworks, Quelle: eigenes Werk / UMLet

steuert und gleichzeitig das Handmodell für die verwendende Software bereitstellt. In der Abbildung 4.4 ist ferner die Struktur des Handmodells als Zusammensetzung aus Unterklassen zu erkennen. Durch die Austauschbarkeit des Handmodells, kann das Framework auch für andere Körperteile oder die Bereitstellung eines Full-Body-Modells genutzt werden. Das gewünschte Modell kann bei der Instanziierung des Frameworks mittels dependency injection integriert werden.

In der Abbildung 4.5 wird der zeitliche Verlauf bei einer Aktualisierung von Feedbackdaten durch die VR-Anwendung dargestellt. Dabei wird der Inhalt des UDP-Pakets von der Netzwerkschnittstelle an das Handmodell zur Verarbeitung weitergereicht. Nach Abschluss der Aktualisierung kann ein von der Hardwaresteuerung zuvor registriertes Event ausgelöst werden.

## 4.2 Hardware

Da die in den Anforderungen beschriebene Hardware zum Zeitpunkt der Arbeit nicht beschaffbar war, wurde diese nach den Prinzipien des Rapid-Prototyping für den Proof of Concept entwickelt. Die Entwicklung soll dabei so einfach und preiswert wie möglich gestaltet werden und trotzdem den Anforderungen an die Funktion gerecht werden.

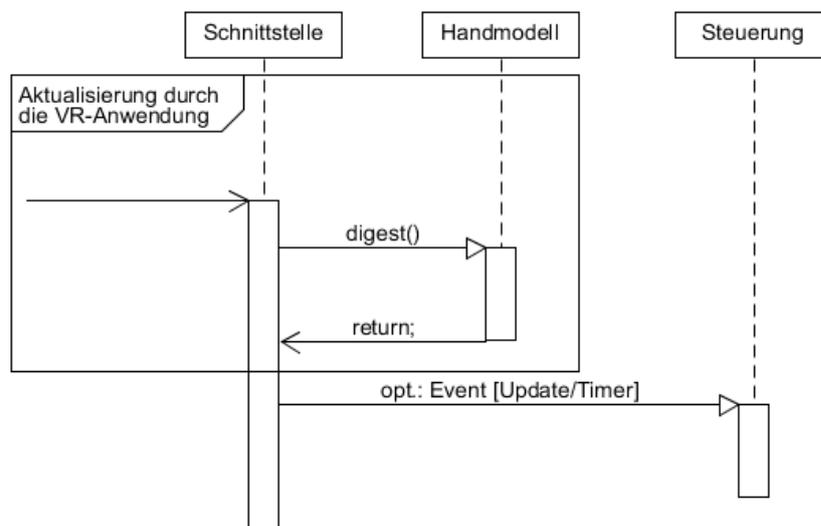


Abbildung 4.5: Übersicht über die eingesetzte Lösung, Quelle: eigenes Werk

#### 4.2.1 Plattform

Zum Einsatz kommt das auf dem AT-MEGA328 basierende Arduino-Entwicklungsboard. Dieses beinhaltet neben der frei zur Verfügung stehenden Entwicklungsumgebung auch alle Betriebskomponenten zum Programmieren und den Betrieb des Mikrocontrollers. Die Programmierung des Controllers erfolgt in C da diese Sprache es erlaubt, auf dem Mikrocontroller abstrakt aber dennoch hardwarenah zu programmieren.

#### 4.2.2 Aktuatoren

Um eine günstige und bekannte Methode für das Haptische Feedback zu verwenden, kommen als Aktuatoren Vibrationsmotoren zum Einsatz, wie sie auch für den Vibrationsalarm eines Mobiltelefons zum Einsatz kommen. Um die Motoren vor Umwelteinflüssen und gegen Blockieren zu schützen, sind diese am fertigen Gerät in ein Gehäuse aus Kunststoff eingelassen. Die Elektronik zur Ansteuerung dieser Feedbackmodule wird auf dem Handrücken eines Handschuhs befestigt. Ebenso werden die Module mit den Aktuatoren an entsprechend definierten Feedbackzonen am Handschuh angebracht. Der Prototyp hat fünf Feedbackzonen, die durch je einen Vibrationsmotor realisiert werden. An jedem Finger befindet sich an der Fingerspitze je eine Feedbackzone. Jede dieser Zonen kann ein physisches Feedback liefern. Am Zeigefinger wird zusätzlich das semantische Feedback realisiert, da dieser meist in der

Interaktion mit Tasten oder Schaltern verwendet wird. Der Schaltplan 4.6 zeigt den Aufbau der

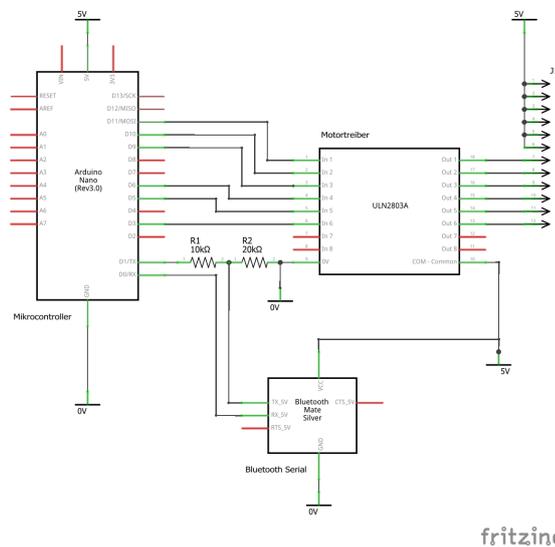


Abbildung 4.6: Der Schaltplan für den Prototyp, Quelle: eigenes Werk / Fritzing

Elektronik. Mikrocontroller des Arduino-Boards steuert mittels Pulsweitenmodulation (PWM) die Leistung der Vibrationsmotoren. Ein Darlington-Array (ULN2803) in der Bildmitte ist hier als Motortreiber eingesetzt. Da die benötigten Ströme zum Betrieb der Motoren bei direktem Anschluss die Ausgänge des Mikrocontrollers zerstören könnten. Am rechten Bildrand sind die Ausgänge des Motortreibers auf einen Steckverbinder geführt, um die an der Schaltung angeschlossenen Aktuatoren tauschen zu können.

## Haptik

Das physische Feedback wird von dem Prototypen durch verschieden intensives dauerhaftes Vibrieren angezeigt. Im Gegensatz dazu soll bei der Interaktion mit Elementen, wie einem Schalter oder Regler, ein semantisches Feedback dem Nutzer nicht die eigentliche Berührung sondern eine Metainformationen über die Handlung oder Eingabe rückmelden. Der Aktuator des Zeigefingers wird dazu, für die Dauer des semantischen Feedbacks, nach Vorgabe der gewünschten Semantik angesteuert. Danach wird die Intensität wieder durch das taktile Feedback gesteuert.

### 4.2.3 Kommunikation

Das verwendete Arduino-Board verfügt über eine Serielle Schnittstelle, die über USB als virtueller serieller Port zur Verfügung steht. Das darüber ausgetauschte binäre Protokoll zur Steuerung des Feedbacks ist auf eine hohe Übertragungsgeschwindigkeit und geringe Verarbeitungszeit ausgelegt. Darüber hinaus hat ein kompaktes Binärprotokoll den Vorteil, dass die Dekodierung im Mikrocontroller nur eine geringe Verarbeitungszeit benötigt und damit nur wenig zur Latenz beiträgt.

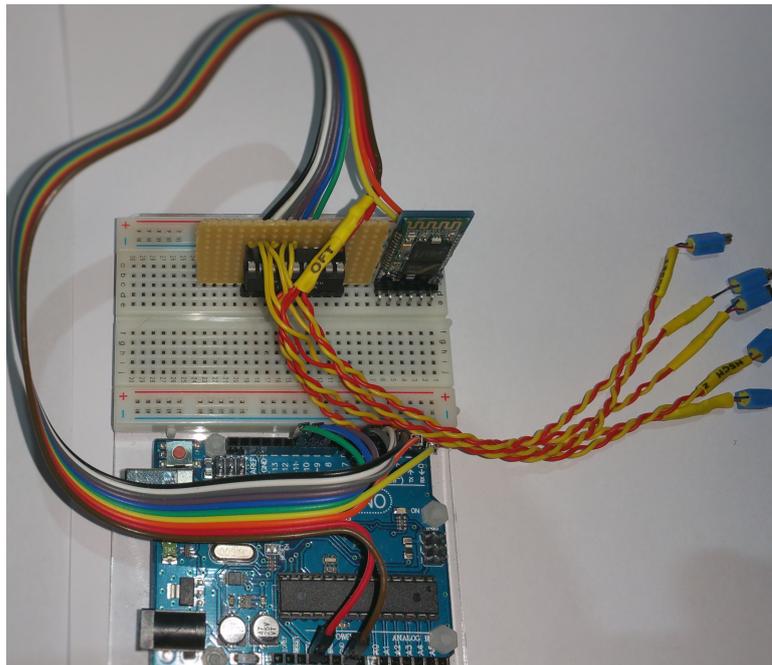


Abbildung 4.7: Testaufbau der Hardware, Quelle: eigenes Werk

Am Testaufbau in der Abbildung 4.7 sind die drei Komponenten für die Realisierung des Feedback zu erkennen. Die Steuerleitungen des Arduino sind an einem Motortreiber angeschlossen, durch den dann die Vibrationsmotoren angesteuert werden. Ferner ist ein Bluetooth-Modul für die drahtlose Übertragung der Steuerbefehle von der Treibersoftware an den Arduino vorhanden.

#### Protokoll

Als Steuerbefehle für die Hardware stehen zwei Befehle zur Verfügung, um das physische und semantische Feedback zu steuern. Ein Befehl besteht aus einem Typenbyte, zu übermittelnde

Nutzdaten und den ASCII-Steuerbefehlen Return und Newline, diese Steuerzeichen dürfen nur als Befehlsende eingesetzt werden. Um die das physiologische Feedback zu steuern wird

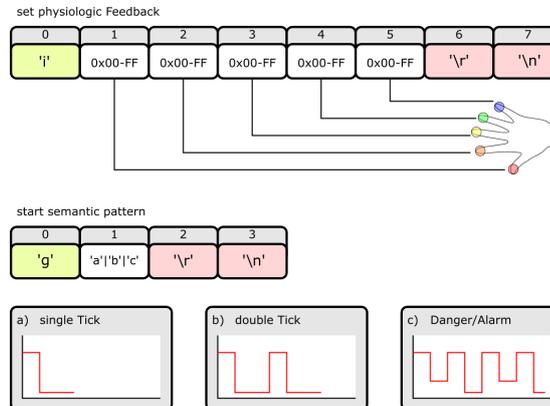


Abbildung 4.8: Protokoll für das Haptik-Feedback-Gerät, Quelle: eigenes Werk

ein 'i' als Kenner für die Intensität gesendet. Danach folgen fünf Bytes, die der Reihe nach die Intensität für das Feedback beginnend am Daumen übertragen. Der erlaubte Wertebereich liegt bei 0 (Aus), zwischen 20 (niedrig) und 255 (hoch).

Semantisches Feedback wird durch ein 'g' gefolgt von 'a', 'b' oder 'c' als Parameter aufgerufen. Die von der Hardware unterstützten Semantiken sind dabei ein einfacher Tick, dieser zeichnet sich wie in der Abbildung 4.8 zu sehen ist durch einen kurzen und starken Impuls, gefolgt von einer Pause, aus. Nach dem gleichen Prinzip wird ein doppeltes Ticken realisiert. Dieses kann zum Beispiel als Bestätigung für eine getätigte Eingabe Verwendung finden. Als letztes stellt ein hektisches in verschiedenen Stärken verlaufendes Muster eine Warnung oder Alarm da. Die Zuordnung der Parameter ist hier willkürlich gewählt, da die Aufrufe für semantischen Feedback an dem anderen Befehl in seiner Struktur angelehnt sein soll.

### 4.3 Zusammenfassung

In diesem Kapitel wurde der Entwurf der umzusetzenden Hardware dargelegt. Zunächst wurden die für das Multi-Device-Framework erforderlichen Design- und Architekturentscheidungen festgelegt. Die daraus entstandene Struktur und Protokolldefinition wird im folgenden Kapitel implementiert. Ebenso sind die Grundlagen für das Feedback-Device

# 5 Prototypische Umsetzung

In diesem Abschnitt wird die Implementation der Lösung für Haptisches Feedback anhand eines Proof of Concept gezeigt. Die Verwendung des Framework ist Thema im Unterkapitel 5.1. Nachfolgend wird im Abschnitt ?? auf die Realisierung der Hardware eingegangen. Zuletzt wird der Aufbau einer Demoapplikation unter 5.3 gezeigt, die zum realen Testen der hier beschriebenen theoretischen Überlegungen dient.

## 5.1 Framework

Wie im vorherigen Kapitel entworfen, wird das Framework als Bibliothek in die Steuerungssoftware eingebunden. Wie in der Abbildung 5.1 zu dargestellt ist, wird durch das Framework

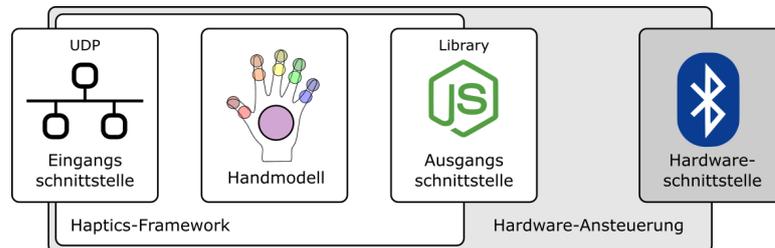


Abbildung 5.1: Illustration des Framework im Einsatz, Quelle: eigenes Werk

die Netzwerkkommunikation sowie deren Aufbereitung und Bereitstellung der Daten als Handmodell übernommen. Die Steuerungssoftware verarbeitet dann die angebotenen Daten und übermittelt dann die Befehle an die Hardware.

### 5.1.1 Steuerungssoftware

Durch das eingebundene Framework und dessen Benachrichtigung nach einem Update, wird hier die Abbildung des Handmodells auf die anzusteuern Hardware und die Kommunikation über die Schnittstelle mit dem Feedback-Device implementiert.

Die Hauptaufgabe der Steuerungs-Software liegt in der Übersetzung. Feedbackdaten aus dem Handmodell werden auf die Fähigkeiten der Hardware umgerechnet und dann an diese übermittelt.

### 5.2 Hardware

Das entwickelte Feedback-Gerät besteht aus einem dünnen Handschuh, der als Träger für die Elektronik und zur Positionierung der Aktuatoren an den Fingerspitzen dient.

#### 5.2.1 Elektronik

Die Hardware-Plattform für den Proof of Concept basiert auf einem Arduino-Board.

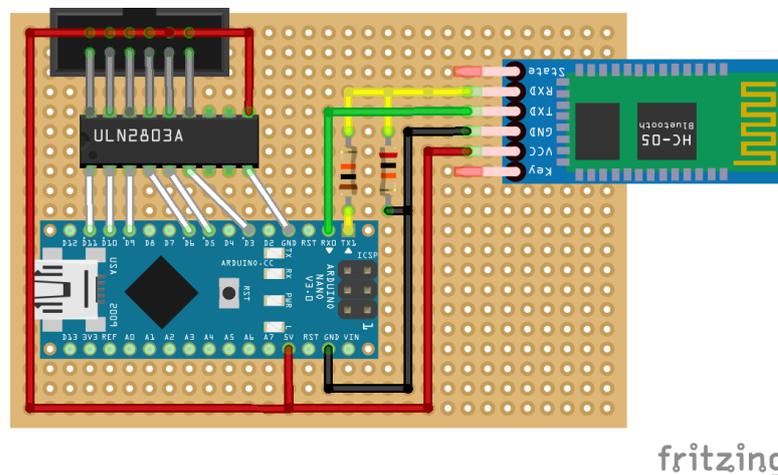


Abbildung 5.2: Hardware des Prototypen als Plan für die Umsetzung auf Lochraster-Platinen, Quelle: eigenes Werk / Fritzing

Die Schaltung wurde zunächst als Testaufbau auf einem Steckbrett realisiert und danach getestet. Im Schaltplan ist zusätzlich ein Bluetooth-Modul vorgesehen, welches optional die Datenübertragung der seriellen Schnittstelle des Arduinos übernehmen kann. Der finale elektronische Aufbau ist in der Abbildung 5.2 zu sehen. Dabei wurden wie im Entwurf in Abschnitt 4.2 vorgesehen die Motoren nicht direkt auf der Leiterplatte sondern mittels Steckverbinder angeschlossen.

### 5.2.2 Firmware

Die Firmware des Mikrocontrollers basiert auf einer Eventloop. Bei jedem Durchlauf der Schleife werden drei Funktionen aufgerufen, welche die Steuerung der Motoren für die beiden Feedbackmethoden und das Dekodieren des Steuerprotokolls übernehmen.

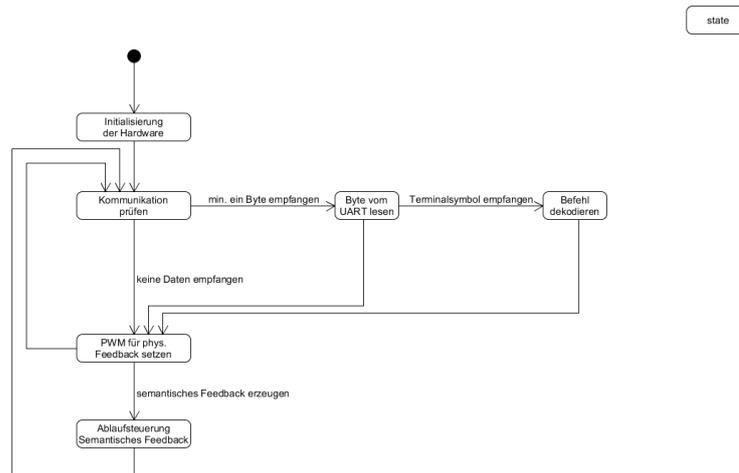


Abbildung 5.3: Struktur der Eventloop, Quelle: eigenes Werk / UMLet

Dazu muss jede Funktion der Firmware in nicht blockierender Weise programmiert werden. Dadurch ist eine hohe Umlaufgeschwindigkeit der Eventloop möglich, jedoch muss beim Warten auf einen Zustand oder einen Wert in jedem Umlauf gepollt werden. Der Zustand der verschiedenen Feedbackmethoden wird in entsprechenden Zustandsvariablen gespeichert.

Wie in der Grafik 5.3 zu sehen ist, wird in jedem Schleifendurchlauf zunächst geprüft ob neue Daten an der seriellen Schnittstelle eingetroffen sind. Ist dies der Fall, wird das erste Byte gelesen und in den Kommunikationspuffer abgelegt. Wenn nun das empfangene Byte das Terminalsymbol ist, so wird die bisher gespeicherte Nachricht dekodiert. Nach der Dekodierung werden die entsprechenden Werte in die dafür vorgesehenen Zustandsvariablen geschrieben. Nachfolgend werden die Ausgänge für das taktile Feedback gestellt sowie bei Bedarf der Ablauf des semantischen Feedbacks gesteuert.

#### Motorsteuerung

Zwei Funktionen steuern die Motoren um das physiologische bzw. semantische Feedback zu erzeugen. Für das Physiologische werden die Motoren entsprechend der übermittelten Werte angesteuert.

Der Aktuator am Zeigefinger weicht im dem Fall, dass ein semantisches Feedback gegeben werden soll von dieser Steuerung ab. Wenn das Semantische Feedback abgelaufen ist, wird die Steuerung durch den physiologischen Modus übernommen.

## 5.3 Demoapplikation

Um die Funktion der hier entwickelten Lösung zu verifizieren wurden im Verlauf verschiedene Test-Applikationen realisiert, diese werden in 5.3.1 kurz dargestellt. Danach wird die Einarbeitung in die Arbeit mit der Leap-Motion in Kombination mit dem Feedback-Device in Abschnitt 5.3.2. Im letzten Teilkapitel 5.3.3 sollen auch die Erfahrungen mit der eingesetzten Lösung verdeutlicht werden.

### 5.3.1 Testapplikationen

Während der Entwicklung des Framework stellte sich der Test durch externe Hilfsprogramme als sehr vorteilhaft heraus. Um von vornherein die Unabhängigkeit der Netzwerkschnittstelle zu testen, wurde dazu ein Tool in Processing 3 mit Python Erweiterung geschrieben. Ebenso wurde der Einsatz des des Frameworks als NodeJS-Modul getestet und damit die Vorarbeit zum Einsatz der Steuerungssoftware geleistet.

Darüber hinaus wurden die Latenzen des Frameworks in NodeJS gemessen. Dazu wurden an das Framework Testdaten gesendet und die benötigte Zeit zwischen dem Absenden der Nachricht und dem Aufruf der Callbackfunktion zur Signalisierung aktualisierter Daten. Die für die Verarbeitung benötigte Zeit schwankte jedoch auch unter hoher CPU-Last durch andere Programme nur selten über 1 ms.

Auch der Mikrocontroller kommt bei der Messung der Roundtrip-Zeit auf eine ähnlich niedrige Zeit. Dabei darf jedoch die Anlaufzeit der Motoren nicht vernachlässigt werden. Dennoch ist die gesamte Latenz bei unter 5 ms anzusetzen. Damit liegt die Gesamtlösung unterhalb der 11,1 ms, die als maximale Framerate der HTC-Vive angegeben wird.

Weitere Experimente mit der Hardware ergaben, dass die Hardware hinreichend schnell angesteuert werden kann, um so das semantische Feedback mittels Software zu realisieren. Durch die Ansteuerung der Feedbackzonen für das physische Feedback, wird dann das semantische Feedback erzeugt.

### 5.3.2 Tracking

Das Tracking der Hand wird in der DemoApp durch die Leap-Motion realisiert. Dieser Sensor nimmt mittels zwei eingebauten Infrarotkameras die Hand des Nutzers aus zwei Perspektiven

auf und errechnet daraus die Position und Haltung der erkannten Hand. Diese Daten sollen in der Demonstrationsapplikation zur Positionierung der virtuellen Hand dienen. An Grenzen stößt das Gerät in dem Moment, in dem die Hand als solche für die Software nicht mehr erkennbar ist. In diesen Momenten kann es zu fehlerhaften Darstellungen der Hand durch die Leap-Motion kommen. Daraus ergab sich für die entwickelte Feedback-Hardware die folgende Problematik. Als Material für den Handschuh kommt nur Material in Frage, welches ausreichend reflektiv für Infrarotes Licht ist. Der zweite wesentliche Faktor besteht in der Größe und Form der Aktuatorbefestigung, denn diese kann zu Fehlinterpretationen bezüglich der Fingerhaltung durch die Leap-Motion führen.

### 5.3.3 DemoApp

Die Applikation wurde zur Demonstration der Eigenschaften des entwickelten Frameworks erstellt. Sie zeigt die Möglichkeiten des kombinierten Feedbacks anhand von Kollisionsobjekten, die ein taktiles Feedback erzeugen, wie auch die semantische Rückmeldung bei der Interaktion mit Gui-Elementen.

Die Störungen im Feedback bei der Nutzung der DemoApp, lagen im Abriss der Trackingdaten, was zum einen auf die teilweise hohe CPU-Last der Leapmotion zum anderen auf die nicht optimal handförmige äußere Form des Feedback-Devices.

### 5.3.4 Zusammenfassung

Die Realisierung des Frameworks mit Hinblick auf die gestellten Anforderungen hat diese in vollem Umfang erfüllen können. Auch die Hardware entspricht den gestellten Anforderungen an die technische Funktion. Mehr noch konnte das Gerät zeigen, dass ein semantisches Feedback durch die Ansteuerung der Aktuatoren über die Befehle für die Steuerung der taktilen Feedbacks möglich ist.

Die ersten Versuche mit der Leap-Motion zeigten jedoch die Grenzen des Handtrackings bezüglich der räumlichen Ausdehnung der Aktuatoren an den Fingerspitzen. Auch hier wirkt sich eine Miniaturisierung der Aktuatorik positiv auf die Benutzbarkeit aus.

## 6 Zusammenfassung und Ausblick

In Abschnitt 2 wurden zunächst die Grundlagen von haptischem Feedback anhand der Physiologie des Tastsinnes und den dazu bereits existierenden Lösungen untersucht. Dabei fiel die Diversität der verschiedenen Ansätze zur Erzeugung von haptischem Feedback auf. Die Systeme wurden dabei in stationäre, also Bauart oder technisch bedingt ortsgebundene, sowie mobile Geräte unterschieden. Obwohl der Fokus dieser Arbeit auf den mobilen Geräten liegt, konnten interessante Technologien gefunden werden, die durch Miniaturisierung auch den Einsatz in mobilen Lösungen finden und damit diese bereichern können.

Im Rahmen dieser Arbeit wurde eine Lösung für mobiles Feedback umgesetzt. Ausgangslage für die Entwicklung war die im Vergleich der bestehenden mobilen Geräte aufgefallenen Ähnlichkeiten in der Modellierung des haptischen Feedbacks, sowie der Bedarf einer generalisierenden Schnittstelle zur Ansteuerung. Als Resultat der Ähnlichkeiten wurde in 4.1.2 ein Handmodell als Grundlage für eine generalisierende Schnittstelle erarbeitet.

Mangels eines Feedback-Devices welches die beiden im mobilen Kontext verbreitetsten Feedback-Methoden realisiert und das zu entwickelnde Framework testen könnte, wurde die Entwicklung eines Prototypen zum zweiten Teil der hier umzusetzenden Lösung. Die an diese Gesamtlösung gestellten Anforderungen lassen sich auf zwei wichtige Kernpunkte reduzieren: Ein generalisierendes Handmodell soll die Arbeitsgrundlage des Frameworks bilden und die Latenzen sollten kleiner als 10 ms sein.

Die grundlegenden Architektur-Entscheidungen und Protokoll-Definitionen wurden im Entwurf erstellt. Im Nachhinein stellte sich heraus, dass zur Erzeugung eines semantischen Feedbacks keine spezielle Implementation notwendig ist. Bei hinreichend hoher Geschwindigkeit in der Ansteuerung, ist das semantische Feedback durch die Verwendung des physiologischen Feedback möglich.

### 6.1 Ausblick

Die hier eingesetzte Framework-Architektur für die generalisierte Ansteuerung von Hardware lässt sich sowohl auf weitere Körperteile in Richtung eines Fullbody Feedbackmodells. Darüber hinaus bietet der Ansatz, einer generalisierten Netzwerkschnittstelle, die Möglichkeit einer

Verarbeitung des Feedbacks als verteiltes System. Dazu muss die Netzwerkkommunikation auch unter dem Aspekt der Sicherheit betrachtet werden.

Zukünftig könnte auch, durch einen Rückkanal, Informationen vom Haptik-Feedback-Device zurück an die VR-Anwendung übermitteln. Dazu müssten dann ähnliche Überlegungen bezüglich der Generalisierung wie in dieser Arbeit angestrebt werden.

Das größte Potential liegt derzeit jedoch bei der Miniaturisierung und Integration verschiedener Methoden um Haptik für VR-Anwendungen zu verbessern. Beispielsweise durch den Einsatz kleiner Lufzellen wie in Abschnitt 2.2.1 vorgestellten Particle Jamming Array, kann ein realer Druck auf die Finger des Nutzers ausgeübt werden.

Ein weiteres, hier nicht weiter betrachtetes, Potentielles Entwicklungsfeld für haptisches Feedback liegt in der direkten elektrischen Stimulation der Nervenenden, um die Gefühle und Empfindungen ohne mechanische Umwege in das Bewusstsein einzubringen.

# Literaturverzeichnis

- [Choi und Follmer 2016] CHOI, Inrak ; FOLLMER, Sean: *Wolverine: A Wearable Haptic Interface for Grasping in VR*. ACM, 2016
- [Dörner u. a. 2013] DÖRNER, Ralf ; BROLL, Wolfgang ; GRIMM, Paul ; JUNG, Bernhard: *Virtual und Augmented Reality (VR/AR)*. Springer-Verlag Berlin Heidelberg, 2013
- [Heinecke 2012] HEINECKE, Andreas M.: *Mensch-Computer-Interaktion*. Springer-Verlag Berlin Heidelberg, 2012
- [Kobayashi u. a. 2005] KOBAYASHI, Nobuyuki ; TOKUNAGA, Eiji ; KIMURA, Hiroaki ; HIRAKAWA, Yasufumi ; AYABE, Masaaki ; NAKAJIMA, Tatsuo: An Input Widget Framework for Multi-modal and Multi-device Environments. (2005)
- [McNeely u. a. 1999] MCNEELY, William A. ; PUTERBAUGH, Kevin D. ; TROY, James J.: Six Degree-of-Freedom Haptic Rendering Using Voxel Sampling. (1999)
- [Meroth und Tolg 2008] MEROTH, Ansgar ; TOLG, Boris: *Infotainmentsysteme im Kraftfahrzeug*. Friedr. Vieweg und Sohn Verlag, GWV Fachverlage GmbH, Wiesbaden, 2008
- [Miguel A. Otaduy u. a. 2016] MIGUEL A. OTADUY, URJC M. ; ALLISON OKAMURA, Stanford U. ; SRIRAM SUBRAMANIAN, University of S.: *Haptic Technologies for Direct Touch in Virtual Reality*. ACM, 2016
- [Yang u. a. 2011] YANG, Yi ; ZHANG, Yuru ; HOU, Zhu ; LEMAIRE-SEMAIL, Betty: Adding Haptic Feedback to Touch Screens at the Right Time. (2011)

# Abbildungsverzeichnis

2.1	Modellaufbau mit vier Zellen, Quelle: Stanford University, <a href="http://charm.stanford.edu/Main/DesignAndControlOfHapticSystems">http://charm.stanford.edu/Main/DesignAndControlOfHapticSystems</a> . . . . .	7
2.2	Phantom, Ein 3D Einagbestift mit haptischem Feedback, Quelle: <a href="http://www.geomagic.com/de/products-landing-pages/haptic/">http://www.geomagic.com/de/products-landing-pages/haptic/</a> . . . . .	8
2.3	Schematische darstellung des Funktionsprinzips durch die Überlagerung von Schallwellen, Quelle: (Miguel A. Otaduy u. a., 2016, 94) . . . . .	9
2.4	Darstellung eines Geräts zur Realisierung von gerichtetem taktilem Feedback, Quelle: URJC Madrid/Siggraph '16, (Miguel A. Otaduy u. a., 2016, 31) . . . . .	10
2.5	Wolverine, ein ein haptisches Interface zum Greifen in der VR, Quelle: Choi und Follmer (2016) . . . . .	11
4.1	Übersicht über die eingesetzte Lösung, Quelle: eigenes Werk . . . . .	19
4.2	Darstellung des intern verwendeten Handmodells, Quelle: Eigenes Werk . . . . .	19
4.3	Beispielpaket für das entworfene Protokoll, Quelle: eigenes Werk . . . . .	21
4.4	Komponentendiagramm des Frameworks, Quelle: eigenes Werk / UMLet . . . . .	22
4.5	Übersicht über die eingesetzte Lösung, Quelle: eigenes Werk . . . . .	23
4.6	Der Schaltplan für den Prototyp, Quelle: eigenes Werk / Fritzing . . . . .	24
4.7	Testaufbau der Hardware, Quelle: eigenes Werk . . . . .	25
4.8	Protokoll für das Haptik-Feedback-Gerät, Quelle: eigenes Werk . . . . .	26
5.1	Illustration des Framework im Einsatz, Quelle: eigenes Werk . . . . .	27
5.2	Hardware des Prototypen als Plan für die Umsetzung auf Lochraster-Platinen, Quelle: eigenes Werk / Fritzing . . . . .	28
5.3	Struktur der Eventloop, Quelle: eigenes Werk / UMLet . . . . .	29

*Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.*

Hamburg, 17. Mai 2017 

---

 Jan Stieglitz