



Hochschule der Medien Stuttgart
Fakultät Electronic Media
Studiengang Audiovisuelle Medien

Entwicklung einer technischen Umgebung für eine Rauminstallation mit Binauraltechnik, Head-Tracking und Indoor-Lokalisierung

Masterarbeit

zur Erlangung des akademischen Grades
Master of Engineering

vorgelegt von

Florian Beck

Matrikelnummer 31764
Bärenseestraße 20, 70569 Stuttgart
mail@florianbeck.de

Erstprüfer: Prof. Oliver Curdt
Zweitprüfer: Prof. Dr. Simon Wiest
Bearbeitungszeitraum: 1. Juni 2017 – 30. November 2017

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen wurden, sind in jedem Fall unter Angabe der Quelle kenntlich gemacht. Die Arbeit ist noch nicht veröffentlicht oder in anderer Form als Prüfungsleistung vorgelegt worden.

Stuttgart, am 30. November 2017

Florian Beck

Inhaltsverzeichnis

Erklärung	ii
Kurzfassung	vi
Abstract	vii
Darstellungsverzeichnis	viii
Abkürzungsverzeichnis	x
1 Einführung	1
1.1 Ausgangssituation	1
1.2 Motivation und Ziele	2
1.3 Vorgehen	3
2 Grundlagen und Definitionen	4
2.1 Räumliches Hören	4
2.1.1 Richtungshören	4
2.1.2 Entfernungshören	7
2.2 Klanglandschaften	7
2.3 Positions- und Lagebestimmung	9
2.3.1 Freiheitsgrade	9
2.3.2 Position und Lage	10
3 Räumliche Tonübertragung	11
3.1 Objektbasierte Tonübertragung	12
3.2 Lautsprecherbasierte Tonübertragung	14
3.2.1 Mehrkanal-Stereofonie	14
3.2.2 Vector Base Amplitude Panning	15
3.2.3 Ambisonics	16
3.2.4 Wellenfeldtsynthese	18
3.2.5 Einordnung konkreter 3D-Audio-Systeme	20
3.3 Binaurale Tonübertragung	21

3.3.1 Außenohrübertragungsfunktion	22
3.3.2 Akustische Raumsimulation	22
3.3.3 Einschränkungen	23
4 Head-Tracking	25
4.1 Mathematische Darstellung	25
4.1.1 Eulersche Winkel	25
4.1.2 Achsenwinkel und Quaternionen	28
4.2 Tracking-Verfahren	29
4.3 Inertialsensoren	31
4.3.1 Beschleunigungssensor	31
4.3.2 Drehratensensor	33
4.4 Inertiale Messeinheit	35
4.4.1 Informationsfusion	35
4.4.2 Lagebestimmung mit Inertialsensoren	36
4.4.2.1 Integration von Drehraten	36
4.4.2.2 Schätzung mittels Beschleunigung	37
4.4.2.3 Fusion beider Sensortypen	37
4.4.3 Bauformen	38
5 Indoor-Lokalisierung	41
5.1 Messverfahren	41
5.1.1 Laufzeitmessung	41
5.1.2 Laufzeitdifferenzmessung	44
5.1.3 Messung des Einfallswinkels	45
5.2 Lokalisierungsmethoden	45
5.2.1 Trilateration	46
5.2.2 Triangulation	47
5.2.3 Umgebungsanalyse	48
5.3 Technologien	48
5.3.1 Geeignete Signale	49
5.3.2 Vergleich verschiedener Systeme	50
5.3.3 Ultra-Breitband	51
6 Umsetzung des Projekts	53
6.1 Funktionsprinzip	53
6.2 Sensorsystem	55
6.2.1 Mikrocontroller	55
6.2.2 Head-Tracking	57
6.2.3 Indoor-Lokalisierung	59
6.2.4 Datenübermittlung	61

Inhaltsverzeichnis	v
6.3 Datenverarbeitung	64
6.3.1 Visuelle Programmierumgebung	64
6.3.2 Head-Tracking	65
6.3.3 Indoor-Lokalisierung	66
6.3.3.1 Glättung	66
6.3.3.2 Kalibrierung	68
6.3.3.3 Trilateration	69
6.3.4 Benutzeroberfläche	72
6.4 Audio-Rendering	74
7 Schlussbetrachtungen	78
7.1 Ergebnisse	78
7.2 Diskussion und Ausblick	79
7.3 Fazit	80
Literaturverzeichnis	81
Anhang	85
A Inhalt des beiliegenden Datenträgers	85

Kurzfassung

Die vorliegende Arbeit kombiniert drei Themenbereiche: Binauraltechnik, Head-Tracking und Indoor-Lokalisierung. Mit diesen Technologien wird eine technische Umgebung für eine immersive Rauminstallation entwickelt. Dieses System soll es ermöglichen, in eine virtuelle, auditive Umgebung einzutauchen. Für die räumliche Tonübertragung werden binaurale Audiosignale erzeugt und über Kopfhörer wiedergegeben. Zum einen werden dabei Bewegungen der Person im Raum, zum anderen eine ortsstabile Lokalisierung der virtuellen Soundquellen berücksichtigt. Das Head-Tracking wird mit verschiedenen Inertialsensoren (Gyroskop, Beschleunigungssensor, Magnetometer) implementiert. Die Indoor-Lokalisierung wird mittels Trilateration auf Basis von Ultra-Breitband realisiert. Neben einer umfangreichen Erläuterung und Einordnung der verwendeten Technologien wird auch die konkrete Umsetzung beschrieben und diskutiert. Die Umsetzung beinhaltet hauptsächlich die Programmierung von Mikrocontrollern zum Auslesen der Messwerte und die Aufarbeitung dieser rohen Messdaten auf einem Rechner.

Schlagworte Binauraltechnik, Immersive Audio, 3D-Audio, Head-Tracking, Indoor-Lokalisierung, Inertialmessung, Trilateration

Abstract

The present paper deals with three major topics: Binaural rendering, head-tracking and indoor-positioning. With this three technologies a immersive room-based installation is engineered. This system should provide the possibility to immerse in an virtual auditive environment. Binaural signals are generated and played via headphones for a spatial audio reproduction. Therefore the movings of the person inside the room, as well as a stable localisation of the sound sources are considered. The head-tracking uses differen inertial sensors (gyroscope, accelerometer, magnetometer). The indoor-positioning is done by a ultra wide band based trilateration. Besides an extensive explanation of the used technologies, the concrete realisation of the project will be explained and discussed. The implementation mainly contains the programming of microcontrollers to read the sensor values and the processing of the raw measurement values inside a computer.

keywords binaural audio, immersive audio, spatial audio, head-tracking, indoor-positioning, inertial meassurement, trilateration

Darstellungsverzeichnis

Abbildungen

2.1	Kopfbezogenes Koordinatensystem	5
2.2	Richtungsbestimmende Bänder	6
3.1	Konzept von kanalbasierter und objektbasierter Tonübertragung .	13
3.2	Lautsprecher-Tripel bei dreidim. Vector Base Amplitude Panning	15
3.3	Kugelflächenfunktionen verschiedener Ordnung	17
3.4	Huygens-Fresnel-Prinzip als Grundlage der Wellenfeldsynthese . .	19
3.5	Binauraltechnik	23
4.1	Gieren, Nicken und Rollen eines menschlichen Kopfes	26
4.2	Achsenwinkel-Darstellung einer Rotation	28
4.3	Optisches Head-Tracking mittels Marker	30
4.4	Bildbasiertes Head-Tracking	30
4.5	Funktionsprinzip eines Beschleunigungssensors	32
4.6	Corioliskraft bei einem Vibrations-Drehratensensor	34
4.7	Komplementärfilter zur Sensorfusion	38
4.8	Plot von Roll-Winkeln (gemessen und angenähert)	39
4.9	Ruhendes und rotierendes Bezugssystem	40
4.10	IMU mit stabiler Plattform	40
5.1	Two Way Ranging	43
5.2	Two Way Ranging mit doppelter Antwort	44
5.3	Messung des Einfallswinkels	45
5.4	Ideale Trilateration	46
5.5	Trilateration mit ungenauen Messwerten	46
5.6	Triangulation	47
5.7	Prinzip von Ultra-Breitband	51
6.1	Schematische Darstellung des Gesamtsystems	54
6.2	Kopfhörer mit angebaute Sensoreinheit	55

Darstellungsverzeichnis	ix
6.3 Localino RTLS Hardware	59
6.4 Signalplan der Sensoren am Kopfhörer	62
6.5 Beispieldatenpakete des Übertragungsprotokolls	63
6.6 Plot einer Entfernungsmessung mit unterschiedlicher Einstellung des Glättungsfilters	68
6.7 Entfernungsmessungen zur Bestimmung einer Korrekturfunktion	68
6.8 Geometrische Darstellung des Ungenauigkeitsausgleichs durch den Trilaterationsalgorithmus	70
6.9 Berechnung der Koordinaten der Basisstationen über deren Ab- stände zueinander	71
6.10 Benutzeroberfläche der Datenverarbeitungsanwendung	73
6.11 Benutzeroberfläche des SoundScape Renderers	76

Tabellen

5.1 Vergleich von Positionierungssystemen in Gebäuden	50
---	----

Hinweis: Für alle Darstellungen werden die Quellen direkt innerhalb der Arbeit in der Darstellungsbeschriftung angegeben. Ist dort keine Quelle angegeben ist der Autor Urheber der Darstellung.

Abkürzungsverzeichnis

A-GPS	Assisted Global Positioning System
AoA	Angle of Arrival (Messung des Einfallswinkels)
ASDF	Audio Scene Description Format
ATSC	Advanced Television Systems Committee
BRIR	Binaural Room Impulse Response (binaurale Raumimpulsantwort)
BRTF	Binaural Room Transfer Function (binaurale Raumübertragungsfunktion)
DMP	Digital Motion Processor
DoF	Degrees of Freedom (Freiheitsgrade)
EEPROM	Electrically Erasable Programmable Read-Only Memory
EPROM	Erasable Programmable Read-Only Memory
FTDI	Future Technology Devices International (Deonym für USB-RS232-Interface-Chips)
GSM	Global System for Mobile Communications
HOA	Higher Order Ambisonics
HRIR	Head Related Impulse Response (kopfbezogene Impulsantwort)
HRTF	Head Related Transfer Function (kopfbezogene Übertragungsfunktion)
ICSP	In-Circuit Serial Programming
IDE	Integrated Development Environment (Entwicklungsumgebung)
ILD	Interaural Level Difference (interaurale Pegelunterschiede)

IMU	Inertial Measurement Unit
IR-UWB	Impulse Radio Ultra Wide Band
ISP	In-System-Programmierung
ITD	Interaural Time Difference (interaurale Zeitunterschiede)
LiPo	Lithium Polymer
MEMS	Microelectromechanical Systems
MPEG	Moving Picture Experts Group
RFID	Radio Frequency Identification
ROM	Read-Only Memory
RTLS	Real-Time Locating System (Echtzeit-Lokalisierung)
RTT	Round Trip Time (Umlaufzeit)
SMD	Surface-Mount Device
SSR	SoundScape Renderer
TCP	Transmission Control Protocol
TDoA	Time Difference of Arrival (Laufzeitdifferenzmessung)
ToA	Time of Arrival (Laufzeitmessung)
TWR	Two Way Ranging
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
UWB	Ultra Wide Band
VBAP	Vector Base Amplitude Panning
WFS	Wellenfeldsynthese
WIPS	Wireless Indoor Positioning System
WLAN	Wireless Local Area Network
XML	Extensible Markup Language

Kapitel 1

Einführung

1.1 Ausgangssituation

Schon seit längerem zählt 3D-Audio als der nächste große Entwicklungsschritt im Audiobereich. In jüngster Zeit schreitet zum einen die Entwicklung, zum anderen aber auch die Kommerzialisierung stark voran.

Im Kino bzw. im Film hat sich Dolby Atmos mittlerweile stark verbreitet. Ende 2016 waren weltweit bereits über 2000 Kinos mit dem System ausgestattet und über 500 Filme entsprechend produziert (vgl. Dolby Laboratories, 2016). Auch die Standardisierung in diesem Bereich schreitet voran. Mit MPEG-H ist mittlerweile ein Audiocodec standardisiert worden (ISO/IEC 23008-3), der unter anderem auch dreidimensionale Audioübertragung beinhaltet. Im März 2017 wurde MPEG-H durch Standardisierung im ATSC 3.0 Digital Television System (A/342 PART 3:2017) Teil des zukünftigen Fernsehstandards in den Vereinigten Staaten.

Eine Technik, die dreidimensionale Audioübertragung ermöglicht, ist die Binauraltechnik. Im diesem Bereich werden zunehmend kommerzielle Produkte verfügbar. Als Beispiele seien hier der Spatial Audio Designer von New Audio Technology¹ oder das WAVES Nx System² genannt. Ersteres ermöglicht die Produktion von dreidimensionalem Audiomaterial. Zweiteres simuliert virtuelle Abhörräume für die Musikproduktion über Kopfhörer mit Head-Tracking.

¹Produktseite des Herstellers New Audio Technology: <http://www.newaudiotechnology.com/de/spatial-audio-designer>

²Produktseite des Herstellers WAVES: <https://www.waves.com/plugins/nx>

Auf internationalen Konferenzen hält das Thema ebenfalls Einzug. Auf der Tonmeistertagung 2016 in Köln widmete sich einer von fünf parallel bespielten Tagungsräumen ausschließlich dem Thema 3D-Audio. Mit der International Conference on Spatial Audio (ICSA) in Graz fand im September 2017 zum vierten Mal eine Konferenz statt, die sich ausschließlich räumlicher Tonübertragung widmet, wobei einen großen Teil auch Vorträge im Bereich Binauraltechnik darstellten. Man kann also durchaus von einer gewissen Relevanz der dreidimensionalen räumlichen Tonübertragung in der aktuellen Entwicklung der Audiotechnik ausgehen.

1.2 Motivation und Ziele

Ziel dieser Arbeit ist es, eine technische Umgebung für eine Rauminstallation mittels Binauraltechnik, Head-Tracking und Indoor-Lokalisierung zu entwickeln. Mit einem solchen System soll es für eine Person möglich werden, in eine virtuelle, auditive Umgebung einzutauchen. Dabei soll zum einen eine Bewegung im Raum und zum anderen eine ortsstabile Lokalisierung einzelner Audioereignisse möglich sein. Inwieweit ein solches System mit den drei Technologien Binauraltechnik, Head-Tracking und Indoor-Lokalisierung umsetzbar ist, wird in dieser Arbeit untersucht. Neben der Realisierung eines solchen Systems, spielt die Auseinandersetzung mit den nötigen Technologien eine große Rolle.

Die Umsetzung soll dabei Bauteile und Technologien nutzen die als Open-Source verfügbar sind. Das schlägt sich zum einen meist in einem deutlich niedrigeren Preis nieder, zum anderen – und das ist für diese Arbeit das ausschlaggebende Kriterium – sind die Systeme offener und die Quelltexte können eingesehen, nachvollzogen und angepasst werden.

Das stellt vor allem eine interessante Entwicklung dar, weil Rauminstallationen zwar oftmals eine dreidimensionale, räumliche Tonübertragung nutzen, diese bislang aber zumeist mit Lautsprechern realisiert wird. Anders als z. B. in einer klassischen Konzertsituation, sollen die Rezipienten einer Rauminstallation explizit dazu angeregt werden, sich in diesem zu bewegen und so die Klänge (und evtl. auch den Raum) zu „erforschen“ und zu erleben. Da Rauminstallation eigentlich immer so angelegt sind, dass sie von mehreren Personen gleichzeitig begangen werden können, macht eine Wiedergabe über

Lautsprecher durchaus Sinn. Durch eine Kopfhörerwiedergabe, wie in dieser Arbeit angestrebt, werden jedoch noch ganz andere Herangehensweisen ermöglicht, da nur die Person mit dem Kopfhörer die virtuelle, auditive Umgebung hören kann. Dadurch sind auch unterschiedliche auditive Umgebungen für unterschiedliche Personen möglich.

1.3 Vorgehen

Die vorliegende Arbeit behandelt sowohl die technologischen Grundlagen, als auch die konkrete Realisierung eines technischen Systems. Nachdem die Ziele dieser Arbeit im vorangegangenen Abschnitt definiert wurden, werden in Kapitel 2 zuerst grundlegende Sachverhalte und Begriffe erläutert. Kapitel 3 widmet sich der räumlichen Tonübertragung. Verschiedene Verfahren werden erläutert und verglichen, sowohl Systeme zur Lautsprecherwiedergabe, als auch Systeme für eine Wiedergabe über Kopfhörer. Dem Thema Head-Tracking nähert sich Kapitel 4 an, indem es – beginnend bei der nötigen Mathematik, bis hin zu konkreten Bauformen – verschiedene Technologien betrachtet. In ähnlichem Aufbau betrachtet Kapitel 5 das Thema Indoor-Lokalisierung. Hier werden Messverfahren, die mathematische Berechnung und konkrete Technologien erläutert. Die ausführliche Auseinandersetzung mit den Grundlagen der verwendeten Technologien stellt eine gute und wichtige Ausgangssituation dar, um schließlich die konkrete Umsetzung des Projekts in Kapitel 6 anzugehen. Dieses Kapitel beschreibt dabei die in einem iterativen Prozess entstandene Realisierung eines Systems für binaurale Rauminstallation. Eine Zusammenfassung der Ergebnisse, ein Fazit und ein kurzer Ausblick schließen die Arbeit ab.

Kapitel 2

Grundlagen und Definitionen

In diesem Kapitel werden sowohl relevante Grundlagen erläutert, als auch spezielle Sachverhalte für diese Arbeit definiert. Dabei unterteilt sich das Kapitel in die Abschnitte *Räumliches Hören*, *Klanglandschaften* und *Positions- und Lagebestimmung*.

2.1 Räumliches Hören

Die Fähigkeit die Richtung und die Entfernung eines akustischen Ereignisses zu erkennen, wird als räumliches Hören bezeichnet. Blauert (1974, S. 2, 30) unterscheidet dabei zwischen dem wahrgenommenen *Hörereignis* und dem physikalischen *Schallereignis*. Die Zuordnung von Hörereignisort und Schallereignisort nennt er *Lokalisierung*.

Räumliches Hören berücksichtigt neben der reinen Zuordnung von Hör- und Schallereignisorten auch Beziehungen zwischen Hörereignissen, deren räumliche Ausdehnung, sowie Wechselwirkungen mit anderen Ereignissen (z. B. anderen Sinnesgebieten). (vgl. Blauert & Braasch, 2008, S. 87)

Im Folgenden werden vor allem die physikalischen Grundlagen von Richtungshören und Entfernungshören erläutert.

2.1.1 Richtungshören

Für die Überlegungen zum Richtungshören wird das kopfbezogene Koordinatensystem nach Blauert (1974) in Abbildung 2.1 zugrunde gelegt. Eine

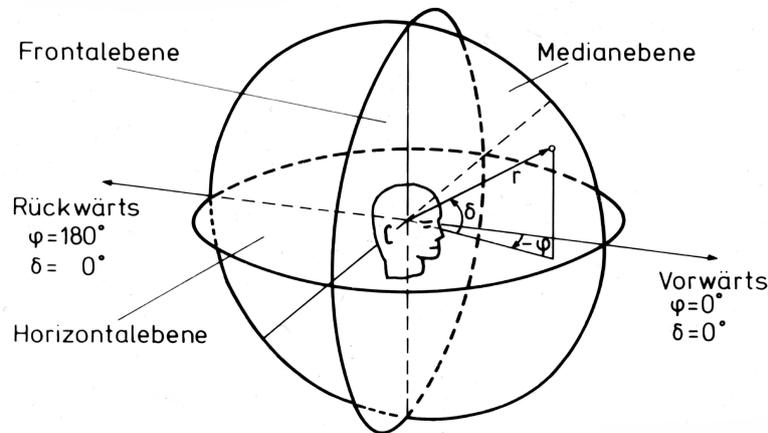


Abbildung 2.1: Kopfbezogenes Koordinatensystem. Quelle: Blauert (1974, S. 11).

Schallquelle wird dabei durch die Kugelkoordinaten¹ Azimut φ , Elevation δ und Abstand r dargestellt.

Bei Schalleinfall aus seitlicher Richtung ($\varphi \neq 0^\circ \neq 180^\circ$), erfolgt die Richtungswahrnehmung hauptsächlich durch Unterschiede der Signale an den beiden Ohren (interaurale Unterschiede). Das sind hauptsächlich Laufzeitunterschiede und Pegelunterschiede. Bei Schalleinfall aus der Medianebene treten praktisch keine interauralen Unterschiede auf. Die Lokalisierung erfolgt in diesem Fall über Spektralunterschiede, hervorgerufen hauptsächlich durch die individuelle Form des Außenohrs. Beim natürlichen Hören treten diese drei Phänomene zwingend gemeinsam auf.

Spektralunterschiede

Befindet sich ein Schallereignis in der Medianebene, treten praktisch keine interauralen Laufzeit- bzw. Pegelunterschiede auf. Durch die Form des Kopfes und vor allem des Außenohrs treten aber richtungsabhängige spektrale Veränderungen auf. Diese sind nicht direkt hörbar, werden aber für die Lokalisierung vom Gehirn ausgewertet. Blauert (1974, S. 90ff) hat diese Spektralunterschiede untersucht und beschreibt die *richtungsbestimmenden Bänder*, welche in

¹räumliche Polarkoordinaten

Abbildung 2.2 dargestellt sind. Das sind bestimmte Frequenzbereiche, deren Präsenz eine bestimmte Lokalisierung zur Folge hat.

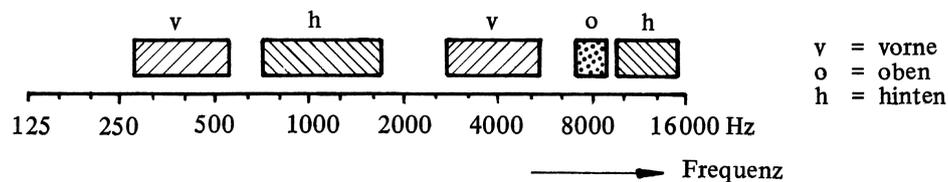


Abbildung 2.2: Richtungsbestimmende Bänder. Quelle: Dickreiter und Goeres-Petri (2014, S. 131).

Einen interessanten Aspekt führt in diesem Zusammenhang auch Sengpiel (1992) an. Demnach stimmen die richtungsbestimmenden Bänder in etwa mit den Kurven gleicher Lautstärke überein.

Wichtig bei Spektralunterschieden ist, dass diese sehr stark von der Anatomie der Außenohren abhängen. Diese ist bei jedem Menschen individuell. Daher kann hier keine allgemeingültige Aussage zu Frequenzbereichen getroffen werden. Die richtungsbestimmenden Bänder sind demnach nur Näherungen und keinesfalls allgemeingültig.

Laufzeit- und Pegelunterschiede

Befindet sich ein Schallereignis nicht auf der Medianebene, treten zusätzlich Laufzeit- und Pegelunterschiede zwischen dem linken und dem Rechten Ohr auf. Diese interauralen Unterschiede steigern die Präzision sehr wesentlich.

Bedingt durch den Ohrabstand können interaurale Laufzeitdifferenzen (engl. Interaural Time Difference, ITD) nur etwa 0,63 Millisekunden betragen. Die Laufzeitdifferenzen kann das Gehör dabei sowohl anhand der Trägerfrequenz, als auch anhand der Hüllkurve beurteilen. Reine Sinustöne können daher nur unterhalb von 800–1600 Hertz lokalisiert werden. (Dickreiter & Goeres-Petri, 2014, S. 129)

Vor allem aufgrund von Abschattung durch den Kopf und das Außenohr entstehen interaurale Pegeldifferenzen (engl. Interaural Level Difference, ILD). Grundsätzlich ist das Gehör in der Lage in allen Frequenzbereichen Pegelunterschiede wahrzunehmen. Unterhalb von etwa 300 Hertz entsteht – aufgrund von Beugung des Schalls um den Kopf herum – jedoch praktisch keine Pegel-

differenzen mehr. Darüber nehmen sie zwar zu, aber – zumindest für reine Sinussignale – nicht proportional mit dem Winkel. Es treten Spektralunterschiede wie bei Schallereignissen in der Medianebene auf. (vgl. Dickreiter & Goeres-Petri, 2014, S. 130)

Unterhalb von etwa 800–1600 Hertz wertet das Gehör also Laufzeitunterschiede von Trägerschwingungen aus. Oberhalb dieser Frequenz werden hauptsächlich Pegeldifferenzen und Laufzeitunterschiede der Hüllkurve ausgewertet. (vgl. Dickreiter & Goeres-Petri, 2014, S. 130)

2.1.2 Entfernungshören

Neben der Wahrnehmung der Richtung gehört auch die Wahrnehmung der Entfernung zum räumliches Hören. Zum Entfernungshören tragen vor allem Lautstärke und Frequenzspektrum des Hörereignisses bei, welche sich auf dem Weg zum Ohr abhängig von der Entfernung verändern. Sehr bedeutend für eine Beurteilung anhand dieser komplexen Merkmale, ist die Erfahrung des Gehörs und die Bekanntheit des Signals. (vgl. Dickreiter & Goeres-Petri, 2014, S. 132)

Mit der Verdoppelung der Entfernung halbiert sich der Schallpegel bei gleichmäßiger Abstrahlung. Erfolgt die Schallemission gerichtet, ist der Pegelabfall geringer. Da tiefe Frequenzen eher ungerichtet abgestrahlt werden, kommt es mit zunehmender Entfernung auch zu Klangfarbenänderungen. Das Gehör wird zudem bei geringen Pegeln für tiefe Frequenzen unempfindlicher. Dies verstärkt den Effekt zusätzlich. (vgl. Dickreiter & Goeres-Petri, 2014, S. 132)

In geschlossenen Räumen trägt zudem das Verhältnis von direktem und diffusem Schall (Nachhall) zum Entfernungshören bei. Auch das setzt eine gewisse Hörerfahrung voraus, da das Verhältnis von Direktschall und Nachhall vom Raum abhängt. (vgl. Dickreiter & Goeres-Petri, 2014, S. 133)

2.2 Klanglandschaften

Der Begriff *Soundscape* (von engl. *sound* und engl. *landscape*) wurde Ende der Sechziger Jahre von Raymond Murray Schafer und Michael Southworth geprägt. Im Deutschen hat sich zu Beginn der neunziger Jahre die Bezeichnung

Klanglandschaft gegenüber anderen Bezeichnungen wie *Schallwelt*, *Lautsphäre*, *Lautlandschaft* oder *akustischer Landschaft* durchgesetzt. (vgl. Winkler, 1995, S. 10)

Das Konzept „Klanglandschaft“ schließt jedes akustische Studienfeld mit ein, es kann sich also beispielsweise auf musikalische Kompositionen, Radioprogramme oder die klangliche Umwelt beziehen (vgl. Hannoschöck, 2009, S. 39).

Im technischen Sinne besteht eine Klanglandschaft aus „aktiven und passiven akustischen Elementen, aus Schallquellen und Schall reflektierenden Objekten“ (Görne, 2017, S. 13). Diese Klänge können in drei Kategorien aufgeteilt werden (vgl. Schafer, 1994 [1977], S. 60, 9f; Hannoschöck, 2009, S. 40):

- *keynote sounds*: Der Hintergrund einer akustischen Umgebung. Sie werden meistens nicht bewusst wahrgenommen, häufig erst bei Veränderung.
- *signals*: Vordergründige Klänge, welche bewusst wahrgenommen werden.
- *soundmarks*: Für eine Region typische Klänge.

Dabei schließen Klanglandschaften sowohl die reine Präsenz von uns umgebenden Klangobjekte ein, als auch explizit die Wahrnehmung eben dieser (vgl. Winkler, 1995, S. 10). Im Bereich des Film-Sounddesigns „kann die Gesamtheit der Tongestaltung einer Szene als Soundscape des im Film gezeigten fiktionalen Ortes aufgefasst werden“ (Görne, 2017, S. 13).

Neben der reinen technischen Auseinandersetzung mit Klanglandschaften, spielt also dessen Wahrnehmung eine wichtige Rolle. Für die Wahrnehmung von Klängen folgert Hannoschöck (2009, S. 51):

Aus dem Verständnis der Sound Studies wird ein Klang nicht einfach nur als physikalisches Phänomen betrachtet, sondern das Gehörte stellt auch immer ein Produkt der menschlichen Wahrnehmung dar, so dass durch die kulturelle Sozialisation die Hörwahrnehmung beeinflusst und das Gehörte mit Bewertungen und Deutungen belegt wird, welche bei verschiedenen Menschen oder Bevölkerungsgruppen außerdem unterschiedlich ausfallen können. Eine Bewertung eines Klanges als ‚Lärm‘ kann auf die Furcht vor einer Störung der sozialen Ordnung hinweisen und hintergründige gesellschaftliche Konflikte offenbaren.

Im künstlerischen Kontext bedeuten Soundscapes meist die Beschreibung

einer Klangumgebung. Zu finden sind sie daher häufig in der *Musique Concrète* oder in Klanginstallationen. Durch die Aufzeichnung und Archivierung von Klanglandschaften soll zudem eine Veränderung der Umwelt dokumentiert werden. Im Film spricht man meist nicht von Soundscapes. Die Funktionsweise und Gestaltung von Atmos bedient sich aber weitgehend Prinzipien von Klanglandschaften.

2.3 Positions- und Lagebestimmung

Bei der Positions- und Lagebestimmung wird häufig nicht genau differenziert. Position und Lage unterscheiden sich jedoch sehr grundlegend. Vor einer Abgrenzung dieser beiden Begriffe werden jedoch zunächst die Freiheitsgrade im Sinne der Mechanik erläutert.

2.3.1 Freiheitsgrade

Die unabhängig voneinander möglichen Bewegungen eines starren Körpers im dreidimensionalen Raum werden durch sechs Freiheitsgrade beschrieben. Dabei kann in *translatorische Bewegung* (alle Punkte erfahren die gleiche Verschiebung) und *rotatorische Bewegung* (alle Punkte bewegen sich mit gleicher Winkelgeschwindigkeit um eine gemeinsame Achse) unterteilt werden (vgl. Zwiers, 2009, S. 4):

Translation

- vor/zurück entlang der X-Achse (Längsachse)
- links/rechts entlang der Y-Achse (Querachse)
- auf/ab entlang der Z-Achse (Hochachse)

Rotation

- von Seite zu Seite kippen (Rollen) an der X-Achse
- vor und zurück kippen (Nicken) an der Y-Achse
- links und rechts drehen (Gieren) an der Z-Achse

2.3.2 Position und Lage

Die Begriffe Position und Lage werden meist recht unscharf verwendet, für diese Arbeit sollen jedoch die folgenden Begriffsbedeutungen gelten:

- Die *Position* beschreibt dabei den Ort im Raum, der durch Translation geändert werden kann. Meist wird sie durch Koordinaten dargestellt. Den Vorgang der Positionsbestimmung nennt man auch Lokalisierung.
- Die *Lage* beschreibt die Orientierung im Raum, sie kann durch Rotation geändert werden. Meist wird sie durch Angabe von Winkeln dargestellt. Den Vorgang der Lagebestimmung nennt man auch Orientierung.

An dieser Stelle ein kurzer Vergleich zum Richtungshören, denn auch dort wird der Begriff Lokalisierung verwendet. Dabei wird einem Hörereignisort im Grunde eine Position im Raum (Richtung und Entfernung) zugewiesen.

Im Verlauf der Arbeit werden sowohl Head-Tracking (Kapitel 4) als auch Indoor-Lokalisierung (Kapitel 5) erläutert. Head-Tracking bestimmt dabei die Lage des Kopfes und die Indoor-Lokalisierung die Position eines Objekts im Raum.

Kapitel 3

Räumliche Tonübertragung

Für die Wiedergabe räumlichen Audioinhalten – wie etwa Klanglandschaften – sind grundsätzlich unterschiedliche Verfahren denkbar. In diesem Kapitel werden dazu verschiedene Kategorisierungen vorgestellt und anschließend konkrete Verfahren und Systeme eingeordnet und erläutert.

Kategorisierung

Tonübertragungsverfahren kann man laut Theile, Wittek und Reisinger (2002, S. 1) in drei grundsätzlich unterschiedliche Methoden einteilen:

- Lautsprecher-Stereofonie
- Synthese des umgebenden Schallfeldes
- Reproduktion der Ohrsignale

Lautsprecher-Stereofonie und die Synthese des umgebenden Schallfeldes nutzen dabei Lautsprecher zur Wiedergabe. Die Reproduktion der Ohrsignale (Binaurale Übertragung) verwendet hingegen Kopfhörer.

Betrachtet man die Tonübertragung eher von der Produktions- und Distributionsseite, kann man in kanalbasierte und objektbasierte Verfahren unterscheiden.

Diese beiden Kategorisierungen werden in den folgenden Abschnitten verwendet um konkrete Systeme und Verfahren einzuordnen.

3.1 Objektbasierte Tonübertragung

Objektbasierte Tonübertragung ist ein vielversprechendes neues Konzept, welches nicht nur die Erstellung von Medieninhalten, sondern vor allem auch die Wiedergabe revolutioniert (Silzle et al., 2016, S. 140).

Kanalbasiert vs. Objektbasiert

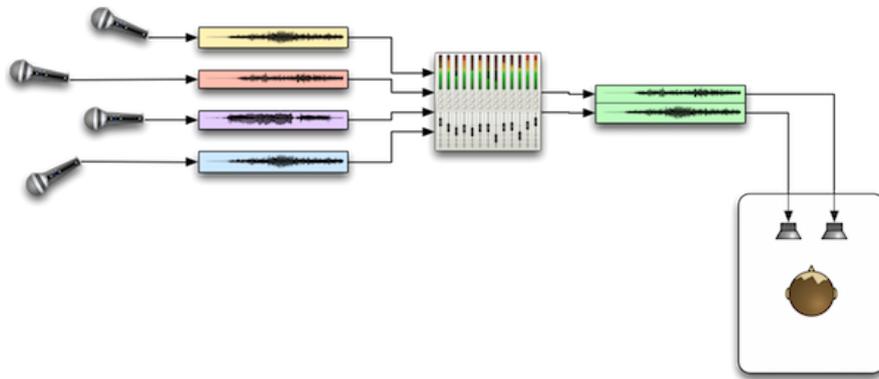
Bei kanalbasierter Übertragung ist jedem Lautsprecher ein expliziter Audiokanal zugewiesen. Sowohl die Anzahl der Audiokanäle, als auch die Anordnung der Lautsprecher ist dabei festgelegt. Jeder Audiokanal enthält genau die Audioinformationen, die später über den entsprechenden Lautsprecher wiedergegeben werden sollen. Abbildung 3.1a zeigt dieses Konzept exemplarisch. (Silzle et al., 2016, S. 141)

Die objektbasierte Übertragung bricht mit diesem Konzept. Hier werden einzelne Audioobjekte (Audioquellen) mit Metadaten (z. B. Position und Lautstärke) verknüpft und dadurch eine Audioszene beschrieben. Erst bei der Wiedergabe werden mit diesen Daten von einem Renderer die Audiosignale für eine bestimmte Lautsprecheranordnung generiert. Abbildung 3.1b zeigt das Konzept der objektbasierten Audioproduktion und der Wiedergabe von objektbasiertem Audiomaterial. (Silzle et al., 2016, S. 141f)

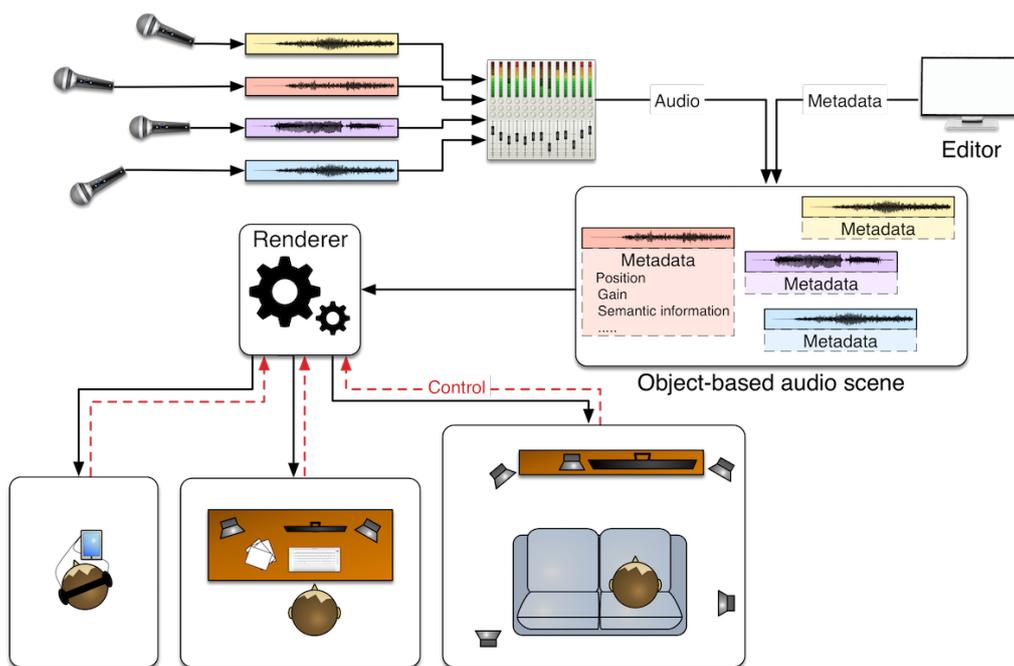
Datenbasiert vs. Modellbasiert

Des Weiteren kann auch zwischen datenbasierter und modellbasierter Abbildung unterschieden werden. Die datenbasierte Abbildung bezeichnet dabei Audiomaterial, welches nicht nur ein Audioereignis beinhaltet, sondern auch die individuellen Parametern des Audioereignisses, die Eigenschaften des Aufnahmebereichs und die Wechselwirkungen zwischen Audioereignissen. Als Beispiel seien hier Aufnahmen mit einem Hauptmikrofon angeführt. (Geier & Spors, 2012, S. 647f)

Die modellbasierte Abbildung beinhaltet hingegen trockene Aufnahmen, welche dann als virtuelle Quellen in einem raumzeitlichen Modell benutzt werden. Das Modell könnte z. B. eine Punktschallquelle sein. Aber auch komplexere Quellen mit spezieller Richtcharakteristik sind denkbar. (Geier & Spors, 2012, S. 648)



a) Kanalbasierte Tonübertragung.



b) Objektbasierte Tonübertragung.

Abbildung 3.1: Konzept von kanalbasierter und objektbasierter Tonübertragung. Quelle: Silzle et al. (2016, S. 142).

Objektbasierte Produktionen sind dabei nicht auf modellbasiertes Audiomaterial beschränkt, sie können ebenso datenbasiertes Audiomaterial enthalten. Auch kanalbasierte Quellen (z. B. Stereofile) können in objektbasierte Produktionen integriert werden. Dann werden Audioobjekte in die Szene integriert, die virtuelle Lautsprecher simulieren. (Geier & Spors, 2012, S. 648)

3.2 Lautsprecherbasierte Tonübertragung

Zu Beginn des Kapitels wurde die Kategorisierung nach Theile vorgestellt. Demnach kann bei Lautsprecherbasierten Übertragungsverfahren in Lautsprecher-Stereofonie und die Reproduktion des Schallfeldes unterschieden werden. „Alle bekannten räumlichen Tonübertragungsverfahren lassen sich darauf zurückführen oder stellen Mischformen dar“ (Theile et al., 2002, S. 1). In diesem Abschnitt werden nun konkrete Verfahren vorgestellt, welche eine Wiedergabe über Lautsprecher realisieren.

3.2.1 Mehrkanal-Stereofonie

Unter Mehrkanal-Stereofonie fallen alle Verfahren, die ein Klangfeld über Phantomschallquellen – also durch Überlagerung von Schallfeldern von zwei oder mehr Lautsprechern – reproduzieren: Stereo, Quadro, Surround, Auro 3D, et cetera. Diese Verfahren sind schon sehr etabliert, daher sind entsprechende Aufnahmetechniken, Produktions-Workflows, entsprechende Wiedergabesysteme usw. sehr verbreitet. Dennoch gibt es zwei entscheidende Nachteile: Zum einen ist es nicht möglich Hörereignisse zwischen den Lautsprechern und dem Hörer zu positionieren. Zum anderen ist die Hörzone (Sweetspot) – bedingt durch die geringe Richtungsstabilität der Phantomschallquellen – sehr begrenzt. (Theile et al., 2002, S. 2f).

Da die Grundsätze der Stereofonie bekannt sein sollten und die Funktionsweisen bei Mehrkanal-Stereofonie den gleichen Prinzipien folgen, wird an dieser Stelle darauf nicht weiter eingegangen. Eine gute Übersicht findet sich z. B. bei Dickreiter, Dittel, Hoeg und Wöhr (2014, Kapitel 5.2, Bd. 1, S. 221–246).

3.2.2 Vector Base Amplitude Panning

Vector Base Amplitude Panning (VBAP) ist im Grunde auch eine Form von Mehrkanal-Stereofonie. Für die Erzeugung der Phantomschallquellen werden jedoch nur Lautstärkeunterschiede verwendet. Dazu wurde das klassische Amplitudenpanning neu formuliert und mittels Vektoren und Basisvektoren ausgedrückt (Pulkki, 1997, S. 457). So wird eine Positionierung von virtuellen Soundquellen ermöglicht, die unabhängig vom verwendeten Lautsprecher-setup ist (Pulkki, 1997, S. 456). Im Grunde handelt es sich also um einen objektbasierten Ansatz.

Theoretisch ist mit VBAP jede Lautsprecheranordnung möglich. Da das Verfahren aber für das Amplitudenpanning auf der Bildung von Lautsprecher-Tripeln basiert (das ist auch exemplarisch in Abbildung 3.2 zu sehen), sind gute Hörerergebnisse nur mit einer ausreichenden Anzahl an Lautsprechern und durch entsprechende Optimierung der Anordnung möglich (vgl. Romanov, Frank, Zotter & Nixon, 2016).

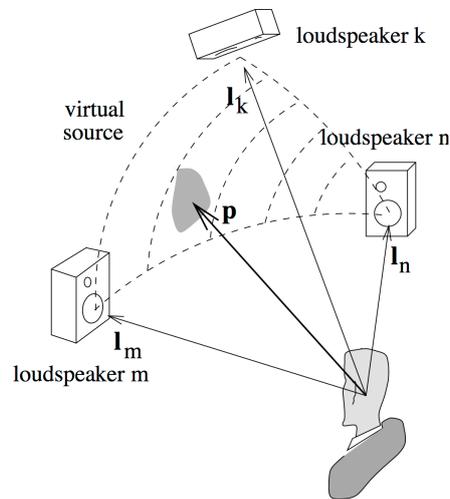


Abbildung 3.2: Lautsprecher-Tripel bei dreidimensionalem Vector Base Amplitude Panning. Quelle: Pulkki (2001, S. 17).

3.2.3 Ambisonics

Mit Ambisonics können dreidimensionale Schallfelder übertragen bzw. virtuelle Klangquellen synthetisiert werden. Die Genauigkeit ist von der Anzahl der verwendeten Aufnahmekanäle bzw. der Anzahl der Lautsprecher bei der Wiedergabe abhängig. (vgl. Slavik & Weinzierl, 2008, S. 659)

Dreidimensionale Schallfelder lassen sich mathematisch als Überlagerung von Kugelflächenfunktionen, den sogenannten sphärischen Harmonischen beschreiben. Diese Funktionen lassen sich als Fourier-Reihe entwickeln, daher steigt die Genauigkeit der Abbildung auch mit der Anzahl der übertragenen Komponenten. (vgl. Slavik & Weinzierl, 2008, S. 660f)

Abbildung 3.3 zeigt Kugelflächenfunktionen verschiedener Ordnung. Werden höhere Ordnungen mit einbezogen spricht man auch von *Higher Order Ambisonics* (HOA). Für das grundsätzliche Verständnis von Ambisonics reicht jedoch die Betrachtung der 1. Ordnung. Eine allgemeine mathematische Beschreibung von Ambisonics (auch höherer Ordnung) findet sich z. B. bei Weinzierl (2008, Kapitel 11.8.2, S. 659–664).

B-Format

Mit den folgenden Gleichungen können Monoquellen s_i im B-Format kodiert (positioniert) werden, wobei eine hörerorientierte Perspektive durch Verwendung von Kugelkoordinaten (Azimut φ und Elevation δ , siehe Abbildung 2.1) berücksichtigt wird (vgl. Hollweger, 2008, S. 1):

$$W = \frac{1}{k} \sum_{i=1}^k s_i \left(\frac{1}{\sqrt{2}} \right) \quad \text{Omnidirektionaler Anteil} \quad (3.1a)$$

$$X = \frac{1}{k} \sum_{i=1}^k s_i (\cos \varphi_i \cos \delta_i) \quad \text{nach X gerichteter Anteil} \quad (3.1b)$$

$$Y = \frac{1}{k} \sum_{i=1}^k s_i (\sin \varphi_i \cos \delta_i) \quad \text{nach Y gerichteter Anteil} \quad (3.1c)$$

$$Z = \frac{1}{k} \sum_{i=1}^k s_i (\sin \delta_i) \quad \text{nach Z gerichteter Anteil} \quad (3.1d)$$

Der omnidirektionaler Anteil (W-Kanal) wird dabei mit einem Mikro-

fon mit Kugelcharakteristik aufgenommen, der W-Kanal repräsentiert also Schalldruck. Der gerichteter Anteil (X-, Y-, und Z-Kanal) wird von drei Mikrofonen mit der Richtcharakteristik Acht aufgenommen, der X-, Y- und Z-Kanal repräsentieren also Schallschnellen. Die Mikrofone sind dabei, wie man Abbildung 3.3 entnehmen kann, orthogonal ausgerichtet. Die Mikrofone müssen zudem koinzident im Ursprung des Koordinatensystems platziert sein. Praktisch lässt sich das jedoch nicht realisieren. (vgl. Hollweger, 2008, S. 1f)

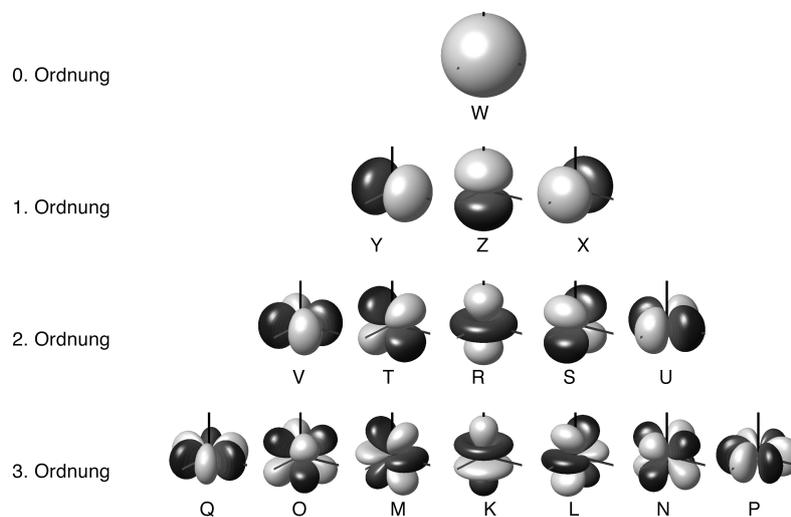


Abbildung 3.3: Kugelflächenfunktionen verschiedener Ordnung. Basiert auf: Frank Zotter (https://commons.wikimedia.org/wiki/File:Spherical_Harmonics_deg3.png <2017-11-05>).

A-Format

Ambisonics wird daher in der Praxis eher im A-Format aufgenommen. Das ist z. B. mit vier tetraedisch angeordneten Mikrofonen mit Nierencharakteristik möglich. Anschließend kann das A-Format mittels Matrizierung (ähnlich der M/S-Stereofonie) in das B-Format überführt werden. Für diese Mikrofonanordnung wird meist der Begriff *Soundfield Mikrofon* verwendet. (vgl. Hollweger, 2008, S. 2)

Dekodierung

Bei der Dekodierung besteht jedes Lautsprecher-signal aus einer gewichteten Summe aller Ambisonics Kanäle. Dabei entspricht die Gewichtung eines Ambisonics Kanals für einen Lautsprecher dem Wert der sphärischen Harmonischen für die Position des Lautsprechers. Das Lautsprecher-signal p_j für den j -ten Lautsprecher wird wie folgt dekodiert (vgl. Hollweber, 2008, S. 3):

$$p_j = \frac{1}{N} \left[W \left(\frac{1}{\sqrt{2}} \right) + X(\cos \varphi_j \cos \delta_j) + Y(\sin \varphi_j \cos \delta_j) + Z(\sin \delta_j) \right] \quad (3.2)$$

Dabei wird die Position des j -ten Lautsprechers über die Kugelkoordinaten φ_j und δ_j definiert. N ist die Anzahl der Ambisonics Kanäle.

Ambisonics hat gegenüber den anderen vorgestellten Verfahren den Vorteil, dass ein unabhängiges Zwischenformat (B-Format) eingeführt wird. Sowohl die Aufnahme, als auch die Wiedergabe sind davon unabhängig (Überführung mittels Matrixierung). Die Qualität hängt dabei sowohl von der Ordnung als auch der Lautsprecheranordnung ab.

3.2.4 Wellenfeldtsynthese

Auch die Wellenfeldtsynthese (WFS) versucht ein Schallfeld nachzubilden. Dabei erzeugen Interferenzen einer Vielzahl an eng zueinander Positionierten Lautsprecher-signalen ein weitgehend physikalisch korrektes Abbild des Schallfeldes.

Physikalische und mathematische Grundlage

Die physikalische Grundlage für die Wellenfeldtsynthese bildet das Huygens-Feresnel-Prinzip. Demnach „kann jeder Punkt einer Wellenfront als Ausgangspunkt einer neuen, sich kugelförmig ausbreitenden Elementarwelle oder Sekundärwelle betrachtet werden. Die weitere Ausbreitung der Wellenfront ergibt sich dann als äußere Einhüllende dieser sich überlagernden Elementarwellen“ (Slavik & Weinzierl, 2008, S. 665). Abbildung 3.4a veranschaulicht dieses Prinzip.

Bei der Wellenfeldtsynthese in Abbildung 3.4b erzeugen die Lautsprecher-signale diese Huygensschen Sekundärwellen. Diese Wellen synthetisiert das

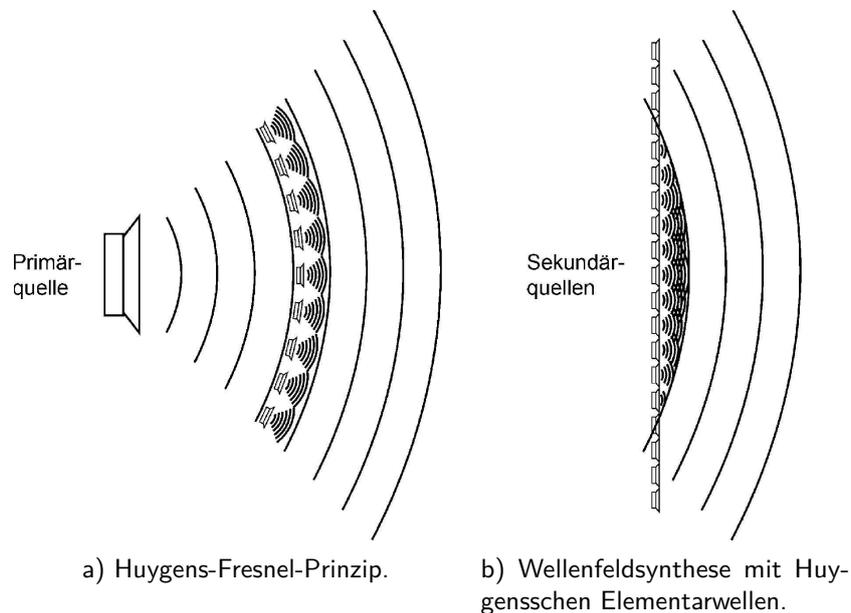


Abbildung 3.4: Huygens-Fresnel-Prinzip als Grundlage der Wellenfeldsynthese. Quelle: Slavik und Weinzierl (2008, S. 666)

Schallfeld einer virtuellen, real nicht vorhandenen Primärquelle. (vgl. Slavik & Weinzierl, 2008, S. 666)

Eine mathematische Formulierung der Wellenfeldsynthese liefert das Kirchhoff-Helmholtz-Integral. Dieses „gibt einen Zusammenhang an zwischen dem Schallfeld in einem Raumvolumen V einerseits und dem Schalldruck und der Schallschnelle auf einer das Volumen V umschließenden Fläche S andererseits“ (Slavik & Weinzierl, 2008, S. 666). Die Aussage des Kirchhoff-Helmholtz-Integrals ist also, dass ein Wellenfeld im Inneren eines Raumvolumens eindeutig durch den Schalldruck an der Oberfläche definiert ist. Eine unmittelbare Umsetzung dieses Prinzips würde in einer komplett geschlossenen, unendlich dichten Lautsprecheranordnung um den Hörer resultieren. (vgl. Slavik & Weinzierl, 2008, S. 666f)

Praktische Umsetzung

Daher verzichtet man in der Praxis sowohl auf „die Anordnung von Schallquellen auf einer geschlossenen Oberfläche zugunsten [sic] eines nur in einer

Ebene angeordneten Arrays von Lautsprechern“ (Slavik & Weinzierl, 2008, S. 667), als auch „auf die unendlich dichte Anordnung der Quellen zugunsten eines Arrays mit diskreten Lautsprecherabständen“ (Slavik & Weinzierl, 2008, S. 667).

Die großen Vorteile der WFS sind eine stabile Lokalisierung in einem großen Hörbereich und die Möglichkeit der Reproduktion von virtuelle Schallquellen, die zwischen Hörer und Lautsprecher-Array positioniert sind. Durch die ortsfesten Punktschallquellen entsteht eine realistische Perspektive. (vgl. Theile et al., 2002, S. 7; Theile, 2014, S. 355)

In der Praxis gibt es jedoch einige Einschränkungen. Durch den Abstand von realen Lautsprechern entsteht ein *Spatial Aliasing* genannter Effekt. Durch Interferenz von Lautsprechersignal mit diskretem Abstand kann nur ein Wellenfeld mit eingeschränktem räumlichen Auflösungsvermögen reproduziert werden. Oberhalb einer bestimmten Grenzfrequenz entstehen Alias Effekte, die sich als falsch orientierte Wellenfronten zeigen. Außerdem entstehen bei begrenzten Lautsprecher-Arrays an den Rändern Beugungswellen. Dieser *Spatial Truncation* genannte Effekt kann sich als Vor- oder Nachecho bei fokussierten Schallquellen bemerkbar machen. Weiterhin können Reflexionen im Wiedergaberaum die Synthese des Schallfelds beeinträchtigen. Darunter leidet vor allem der räumliche Eindruck. Und zuletzt ist durch das Konzept der getrennten Übertragung von Klanggestalt und Raum die Übertragung ausgehnter Schallquellen (z. B. Atmo oder ein Chor) bei der WFS problematisch. (vgl. Theile, 2014, S. 357; Slavik und Weinzierl, 2008, S. 669)

3.2.5 Einordnung konkreter 3D-Audio-Systeme

In diesem Abschnitt sollen die gängigsten 3D-Audio-Systeme kurz in die eben vorgenommene Kategorisierung eingeordnet werden.

Auro 3D – im Jahr 2005 von der belgischen Firma Auro Technologies entwickelt und nun zusammen mit Barco vertrieben – ist eine dreidimensionale Erweiterung von klassischem 5.1 Surround mit einer weiteren Ebene Lautsprecher, dem sogenannten Höhenlayer. In einigen Konfigurationen enthält Auro 3D zudem einem Toplayer mit Lautsprechern über den Zuhörern. Die Produktion und die Wiedergabe erfolgen kanalbasiert mit datenbasiertem Audiomaterial. Die Lautsprecherkonfigurationen sind bei Auro 3D fest vorge-

geben. Die Audiodaten werden im sogenannten Octopus Codec kodiert. (vgl. Auro Technologies & Barco, 2012)

Bei *Dolby Atmos* – von den Dolby Laboratories im Jahr 2012 vorgestellt – wird dagegen objektbasiert gearbeitet. Einzelne Soundquellen werden dabei modellbasiert behandelt. Dolby Atmos berücksichtigt aber z. B. mit sogenannten Beds auch datenbasierte Elemente, die über virtuelle Lautsprecher realisiert werden. Bei Atmos gibt es neben Lautsprechern, die um die Zuhörer in Ohrhöhe angebracht sind, auch Deckenlautsprecher, welche für die Wiedergabe der Höheninformationen des Audiosignals verantwortlich sind. Durch die objektbasierte Architektur ist für die Wiedergabe ein Dolby Controller notwendig, der die Lautsprechersignale je nach Lautsprecherkonfiguration aus den Audio- und Metadaten berechnet. Dolby Atmos ist vor allem in Kinos verbreitet. (vgl. Dolby Laboratories, 2014)

Iosono wurde vom Fraunhofer Institut entwickelt, gehört aber seit 2014 zu Barco Audio Technologies. Es basiert – anders als die beiden bisher vorgestellten Systeme – auf der Wellenfeldsynthese. Die Produktion und Wiedergabe erfolgt objektorientiert. Durch die Eingliederung bei Barco ist seit 2015 auch *AuroMax* verfügbar, eine Fusion aus Auro 3D und Iosono. Es kombiniert Mehrkanal-Stereophonie mit objektbasierter Wellenfeldsynthese. (vgl. Auro Technologies & Barco, 2015)

Des Weiteren sind einige Systeme auf dem Markt, die eine bestimmte Raumakustik simulieren. Dabei werden Schallereignisse im Raum mit Mikrofonen aufgenommen, entsprechende Raumsignale (Reflexionen, Hall) berechnet und mittels Lautsprechern wiedergegeben. Als Beispiele seien hier *Constellation* von Meyer Sound oder *AFC3* von Yamaha genannt.

3.3 Binaurale Tonübertragung

Die bisher vorgestellten Verfahren nutzen alle eine Wiedergabe über Lautsprecher. Im Gegensatz dazu nutzt die Binauraltechnik (binaural, zweiohrig) eine Wiedergabe über Kopfhörer. Sie versucht die beiden Ohrsignale korrekt zu synthetisieren und dadurch eine im Idealfall komplette Rekonstruktion einer akustischen Umgebung zu schaffen. Der Schalldruckverlauf von den Trommelfellen enthält alle dazu nötigen Informationen. Binaurale Signale können im Prinzip mit Mikrofonen im Gehörgang aufgenommen werden. Von

Binauraltechnik spricht man hingegen erst, wenn die Ohrsignale durch digitale Signalverarbeitung generiert werden. (vgl. Slavik & Weinzierl, 2008, S. 671)

Die Produktion erfolgt dabei meist objektbasiert. Die Distribution und die Wiedergabe hingegen erfolgt mittels eines herkömmlichen Stereofilms, ist somit also kanalbasiert.

Die Synthese von Ohrsignalen „basiert auf der Tatsache, dass die Schallübertragung von einem Punkt im Raum bis vor das menschliche Trommelfell ein weitgehend lineares, zeitinvariantes System ist und somit im Zeitbereich durch eine Impulsantwort bzw. im Frequenzbereich durch eine Übertragungsfunktion beschrieben werden kann“ (Slavik & Weinzierl, 2008, S. 671).

3.3.1 Außenohrübertragungsfunktion

Die Außenohrübertragungsfunktion (engl. *Head Related Transfer Function*, HRTF) beschreibt die durch Zusammenwirken der individuellen anatomischen Merkmale entstehenden linearen Verzerrungen des Schallsignals auf dem Weg vom Schallereignisort zum Trommelfell. (vgl. Blauert, 1974, S. 62) Die HRTF „ist das Ergebnis von akustischer Abschattung, Beugung, Verzögerung, Resonanzen und Reflexionen durch Torso, Schulter, Kopf, Ohrmuscheln (pinnae), den Eingang in den Ohrkanal (cavum conchae) und den Ohrkanal selbst“ (Slavik & Weinzierl, 2008, S. 586).

Die Darstellung im Zeitbereich geschieht mittels kopfbezogener Impulsantwort (engl. *Head Related Impulse Response*, HRIR). HRTF (Frequenzbereich) und HRIR (Zeitbereich) können mittels Fourier-Transformation umgerechnet werden. „Das binaurale Signalpaar lässt sich somit durch Faltung des am Senderort abgegebenen Signals mit den beiden kopfbezogenen Raumimpulsantworten [...] erzeugen“ (Slavik & Weinzierl, 2008, S. 671).

3.3.2 Akustische Raumsimulation

Bezieht man bei der Messung nicht nur den Freifeldanteil, also den Einfluss von Außenohr, Kopf und Torso mit ein, sondern auch die raumspezifischen Reflexionen, spricht man von einer binauralen Raumübertragungsfunktion (engl. *Binaural Room Transfer Function*, BRTF) bzw. binauralen Raumimpulsantwort (engl. *Binaural Room Impulse Response*, BRIR).

Abbildung 3.5 zeigt das Prinzip der Binauraltechnik. Aus einem mög-

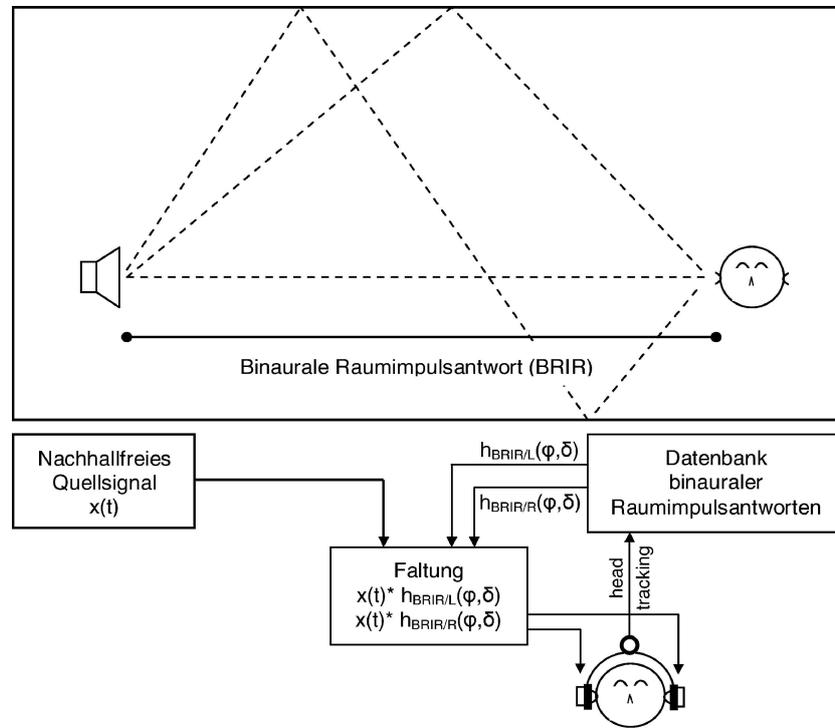


Abbildung 3.5: Binauraltechnik. Quelle: (Slavik & Weinzierl, 2008, S. 672).

lichst nachhallfreien Signal werden die beiden Binauralsignale mittels zuvor gemessenen oder aus einer Datenbank entnommenen BRIRs synthetisiert. Durch Head-Tracking kann die Orientierung des Kopfes bestimmt, somit das passende BRIR-Paar gewählt und das Signal entsprechend nachgeführt werden.

3.3.3 Einschränkungen

Laut vgl., Slavik und Weinzierl (2008, S. 271f) gibt es bei der Binauraltechnik folgende Schwierigkeiten:

- *Individuelle Anatomie:* Die Impulsantworten sind sehr stark von der individuellen Anatomie des Außenohrs und des Kopfes abhängig. Daher sind sie für unterschiedliche Personen sehr individuell. Der Effekt der binauralen Audiowiedergabe kann also – abhängig von der Übereinstimmung der Impulsantworten bei der Aufnahme und der tatsächlichen Impulsantwort

des Rezipienten – stark variieren.

- *Kopforientierung*: Wenn sich die Orientierung des Kopfes ändert, muss auch die Impulsantwort möglichst verzögerungsfrei in die zugehörige Impulsantwort überführt werden.
- *Entzerrung*: Die Impulsantworten müssen von allen Verzerrungen, die z. B. durch das Messmikrofon oder die Wiedergabekopfhörer entstehen, befreit werden.
- *Faltung in Echtzeit*: Die Faltung der Audiosignale mit den Impulsantworten sollte mit möglichst geringer Latenz ablaufen. Während HRIRs relativ kurz sind, sind BRIRs wesentlich länger (mindestens so lang wie die Nachhallzeit) und somit auch wesentlich aufwendiger in der Berechnung.
- *Lautsprecherwiedergabe*: Die Wiedergabe binauraler Signale über Lautsprecher führt zu Klangverfärbungen und dem Verlust der Räumlichkeit – ist also nicht zu empfehlen.

Kapitel 4

Head-Tracking

Head-Tracking ist die Erfassung der Lage (Orientierung) des Kopfs mittels technischer Einrichtungen. Dieses Kapitel beschreibt die Lagebestimmung von Objekten im Allgemeinen, geht dabei aber teilweise auch auf Besonderheiten bei der Lagebestimmung des Kopf im Speziellen ein. Sowohl mathematische und technische Grundlagen, als auch konkrete Realisierungsansätze und technische Umsetzungen werden behandelt.

4.1 Mathematische Darstellung

Um die Lage eines Objekts in einem räumlichen Bezugssystem (Koordinatensystem) zu definieren, gibt es in der Mathematik verschiedene Modelle. Die im Kontext dieser Arbeit Wichtigsten sind *Eulersche Winkel* und *Quaternionen*. Diese beiden Modelle werden in diesem Abschnitt kurz beschrieben und ihre Vor- und Nachteile aufgezeigt.

4.1.1 Eulersche Winkel

Die Eulerschen Winkel beschreiben die Lage eines Objekts durch die Winkelangaben der nacheinander nötigen Einzeldrehungen an den momentanen Koordinatenachsen, die benötigt werden, um das Objekt von seiner Ursprungslage in die zu beschreibende Lage zu bringen (vgl. Eberhard & Fleißner, 2017, S. 2):

— **Gier-Winkel** ψ (engl. *yaw*): Drehung um die z -Achse (Vertikalachse)

- **Nick-Winkel** ϑ (engl. *pitch*): Drehung um die y -Achse (Querachse)
- **Roll-Winkel** φ (engl. *roll*): Drehung um die x -Achse (Längsachse)

Zu beachten ist dabei zwingend die Reihenfolge der Einzeldrehungen, weil nur die erste Drehung an einer raumfesten Achse stattfindet. Anschließend wird an den Achsen des jeweils zuvor entstandenen Systems gedreht. Für diese Arbeit wird die Reihenfolge der Einzeldrehungen und die Bezeichnung der Winkel an die übliche Konvention in der Luft- und Raumfahrttechnik (DIN9300) angelehnt.

Wie die Eulerschen Winkel an einem menschlichen Kopf angewendet werden können, zeigt Abbildung 4.1. Die Eulerschen Winkel repräsentieren dabei exakt den drei rotatorischen Freiheitsgrade (diese werden in Abschnitt 2.3.1 erläutert). Mit Gier-, Nick- und Roll-Winkel kann die Lage eindeutig beschrieben werden. Die Winkelangaben lassen sehr einfach einen Übertrag aus der Realität in die mathematische Darstellung zu – umgekehrt gilt das gleiche. Im Fall der Kopfposition ist das besonders deutlich, da hier die Eulerschen Winkel exakt die menschliche Anatomie reproduzieren. Sie sind somit die weitaus weniger Abstrakte Form der beiden mathematischen Modelle.

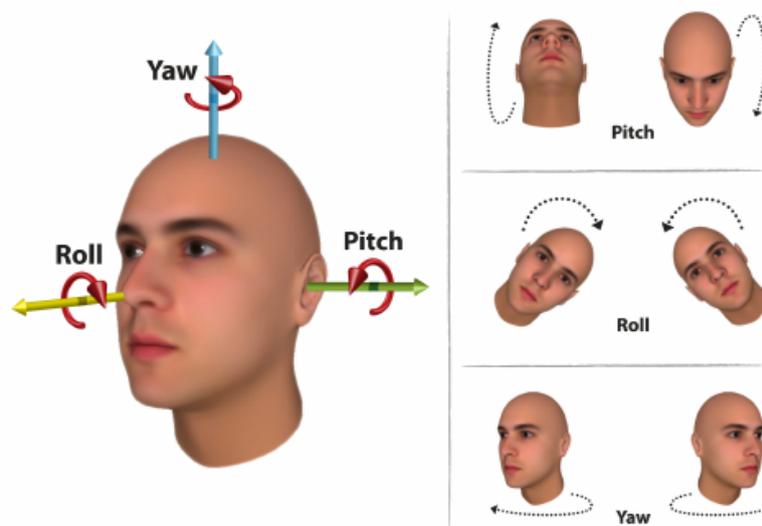


Abbildung 4.1: Gieren (engl. *yaw*), Nicken (engl. *pitch*) und Rollen (engl. *roll*) eines menschlichen Kopfes. Quelle: Arcoverde et al. (2014, S. 281).

Um Drehungen mathematisch zu berechnen, werden Drehmatrizen (4.1)

verwendet, welche in eine Rotationsmatrix (4.2) überführt werden können, die die Gesamtrotation beschreibt (Eberhard & Fleißner, 2017, S. 2):

$$D_x(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & -\sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{bmatrix} \quad (4.1a)$$

$$D_y(\vartheta) = \begin{bmatrix} \cos(\vartheta) & 0 & \sin(\vartheta) \\ 0 & 1 & 0 \\ -\sin(\vartheta) & 0 & \cos(\vartheta) \end{bmatrix} \quad (4.1b)$$

$$D_z(\varphi) = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.1c)$$

$$R = D_z(\psi) \cdot D_y(\vartheta) \cdot D_x(\varphi) \quad (4.2)$$

Computerprogramme können diese Berechnungen zwar theoretisch gut durchführen, da sie aber nicht exakt rechnen – aufgrund der begrenzten Genauigkeit bei der Darstellung von Gleitkommazahlen kommt es zu Rundungsfehlern – summieren sich die Fehler bei wiederholter Matrix-Matrix-Multiplikation (vgl. Hoffmann, 2009, S. 5).

Einen weiteren Nachteil stellt die *kardanische Blockade* (engl. Gimbal Lock) dar: Fallen durch Rotation der zweiten Achse die erste und die dritte Achsen zusammen, verliert das System einen Freiheitsgrad. Ein mögliches Beispiel ist ein Nicken von 90° . Anschließend hätte Gieren und Rollen den gleichen Effekt, die dritte Dimension ist blockiert¹.

Zudem besitzen die Eulerschen Winkel eine Mehrdeutigkeit: Eine Endposition kann durch unterschiedliche Rotationen erreicht werden. Ein Rollen um 180° wäre z. B. auch durch ein Gieren um 180° in Verbindung mit einem Nicken um 180° darstellbar².

¹Das ist anatomisch beim menschlichen Kopf nicht möglich, in anderen Anwendungsbe-
reichen – z. B. bei einem Flugzeug – aber durchaus denkbar.

²Vgl. die Hinweise in der vorangegangenen Fußnote.

4.1.2 Achsenwinkel und Quaternionen

In der Informatik werden Rotationen häufig durch Quaternionen beschrieben und berechnet. Die Berechnung ist genauer als die Berechnung mit Matrizen. Bei ihnen existiert weder die kardanische Blockade, noch sind sie mehrdeutig wie die Eulerschen Winkel. Die Darstellung einer Drehung mittels Quaternionen ist also mathematisch um ein vielfaches genauer als die Darstellung mittels Eulerscher Winkel. Sie ist aber auch wesentlich abstrakter.

Quaternionen sind den reellen Zahlen ähnlich, sie besitzen aber vier Dimensionen (vgl. Hoffmann, 2009, S. 5):

$$q = \begin{pmatrix} q_\omega \\ q_x \cdot i \\ q_y \cdot j \\ q_z \cdot k \end{pmatrix} \quad (4.3)$$

q_ω ist der skalare Anteil, q_x , q_y und q_z beschreiben den vektoriellen Anteil. Für die Beschreibung von Drehungen werden Quaternionen normiert ($q = \sqrt{q_\omega^2 + q_x^2 + q_y^2 + q_z^2} = 1$). i , j , und k ist der imaginäre Teil des Quaternionen (für ihn gilt: $i^2 = j^2 = k^2 = ijk = -1$).

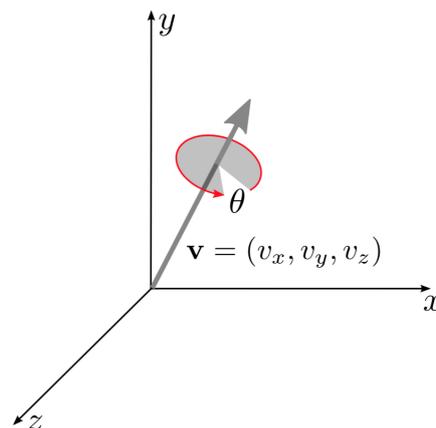


Abbildung 4.2: Achsenwinkel-Darstellung einer Rotation. Quelle: Wetzstein (2017, S. 7).

Eine mögliche Darstellungsform von Quaternionen ist die Achsenwinkel-Darstellung wie in Abbildung 4.2 gezeigt. Im Gegensatz zur Darstellung durch

Einzelrotationen wie bei den Eulerschen Winkeln, wird hier die Lage durch den Vektor v und eine Drehung θ dargestellt.

Ein Beispiel für die Darstellung eines Quaternions im Speicher eines Computerprogramms gibt Hoffmann (2009, S. 7).

4.2 Tracking-Verfahren

Head-Tracking-Verfahren können grundsätzlich in zwei verschiedene Gruppen eingeteilt werden. Zum einen sind dies *optische Verfahren*, zum anderen *sensorbasierte Messverfahren*. Die Einteilung wird in diesem Abschnitt genauer erläutert und die verschiedenen Verfahren werden verglichen. Im weiteren Verlauf der Arbeit werden aber – aufgrund des speziellen Anwendungsfalls – hauptsächlich sensorbasierte Verfahren behandelt.

Optisch

Optische Verfahren nutzen eine oder mehrere Kameras um anhand des Kamerabildes die Lage eines Objektes zu berechnen. Es gibt eine Vielzahl an Tracking-Systemen, die dieses System nutzen (vgl. Ribo, 2001, S. 2–8). Als weitere Klassifizierung kann man zwischen markerbasierten und bildbasierten Systemen unterscheiden.

Markerbasierte Systeme verwenden aktive oder passive Marker, die am Objekt angebracht werden. Durch die Wahl der Marker kann sichergestellt werden, dass diese leicht im Kamerabild analysiert werden können. Durch die Veränderung der Position der Marker zueinander kann die Lage berechnet werden. Abbildung 4.3 zeigt ein optisches Tracking mit Marker. (vgl. Mulder, Jansen & van Rhijn, 2003, S. 4)

Durch aufwändige Analyse des Kamerabildes, kann bei bildbasierten Verfahren die Lage des Objekts anhand des Objekts selbst berechnet werden. Dazu werden z. B. Ecken oder Kanten des Objekts als eine Art Marker analysiert. Bei Head-Tracking werden z. B. häufig die Augen und andere Gesichtsmerkmale verwendet. Abbildung 4.4 zeigt solch ein bildbasiertes optisches Tracking-Verfahren. (vgl. Mulder et al., 2003, S. 4)

Durch die Verwendung einer Kamera schränken optische Verfahren die Nutzungsmöglichkeiten stark ein: Das Objekt muss sich zwingend im Aufnahmebereich der Kamera befinden, dadurch ist keine große Bewegung des

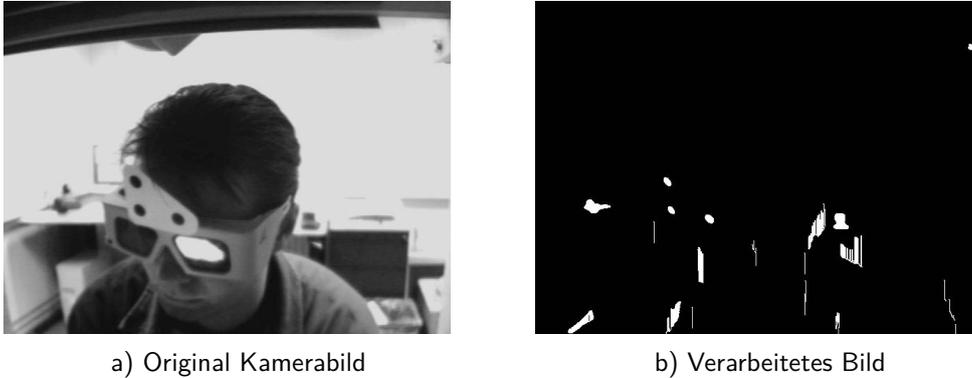


Abbildung 4.3: Optisches Head-Tracking mittels Marker. Quelle: Mulder, Jansen und van Rhijn (2003, S. 5).

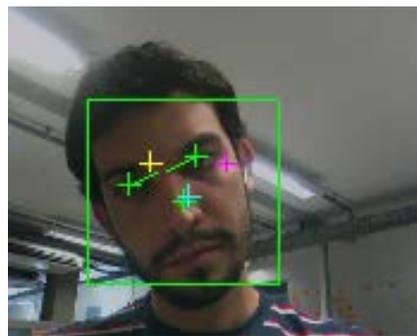


Abbildung 4.4: Bildbasiertes Head-Tracking. Quelle: Arcoverde et al. (2014, S. 289).

Objekts im Raum möglich. Da außerdem oftmals der Bereich, in dem die Lage bestimmt werden kann, beschränkt ist, ist eine 360°-Drehung oft nicht möglich. Für viele Head-Tracking Anwendungen ist das kein Ausschlusskriterium, oft sitzt der Nutzer vor einem Bildschirm, sieht in dessen Richtung und bewegt sich nicht von diesem weg. Eine Lagebestimmung von Objekten im Raum, wie für dieses Projekt nötig, ist also mit optischen Verfahren nicht möglich³.

³Durch Zusammenschaltung mehrerer solcher Systeme könnte das zwar theoretisch trotzdem realisiert werden, ist technisch dann aber sehr aufwändig.

Sensorbasiert

Nicht betroffen von den eben beschriebenen Einschränkungen im Anwendungsbereich sind Verfahren, die Sensoren zur Lagebestimmung nutzen. Das funktioniert z. B. durch Messung der Beschleunigung oder der Winkelgeschwindigkeit. Oft werden zum Tracking verschiedene Sensoren in einer *inertialen Messeinheit* kombiniert. Diese wird im Abschnitt 4.4 genauer betrachtet.

Sensorbasierte Tracking-Systeme können auch die Lage von sich bewegenden Objekten und in allen Richtungen messen. Damit sind sie sehr gut für ein Head-Tracking geeignet, wenn dieses losgelöst von einem fixen Bildschirm stattfinden soll. Daher wird in den folgenden Abschnitten die Technologie hinter solchen Tracking-Verfahren genauer betrachtet.

4.3 Inertialsensoren

Für ein gut funktionierendes Head-Tracking-System sind verschiedenen Sensoren nötig, welche im Folgenden kurz erklärt werden. Anhand der Trägheit der Masse können Inertialsensoren (lat. *inertia* für Trägheit) die Beschleunigung und die Winkelgeschwindigkeit messen. Zur den Inertialsensoren zählen Beschleunigungssensoren (engl. *accelerometer*) und Drehratensensoren (engl. *gyroscope*).

4.3.1 Beschleunigungssensor

Ein Beschleunigungssensor (engl. *accelerometer*) misst die durch die Beschleunigung a in Richtung einer definierten Achse (translatorisch) hervorgerufene Kraft F_a auf eine Masse m in m/s^2 .

„Das Grundprinzip aller Sensoren zur absoluten Messung der Beschleunigung besteht darin, die Wirkung der Beschleunigung auf ein gedämpftes Feder-Masse-System zu messen. Diese Wirkung der Beschleunigung besteht darin, dass die elastisch an das Gehäuse angekoppelte seismische Masse sich bei Beschleunigung des Gehäuses gegenüber dem Gehäuse verschiebt.“ (Horn & Höflinger, 2014, S. 544)

Der prinzipiellen Funktionsaufbau eines Beschleunigungssensors ist in Abbildung 4.5 dargestellt. Die seismische Masse m wird durch die Beschleuni-

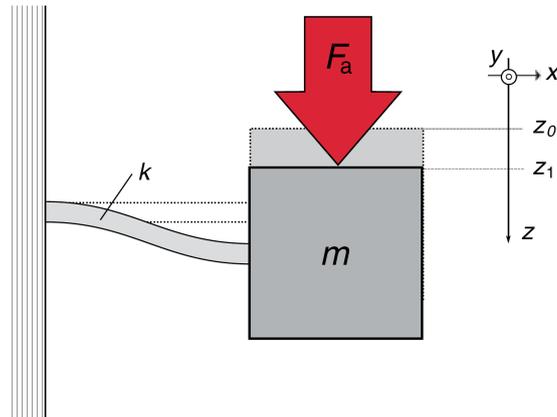


Abbildung 4.5: Funktionsprinzip eines Beschleunigungssensors. Quelle: Horn und Höflinger (2014, S. 545).

gungskraft F_a in z -Richtung ausgelenkt. Die Masse m ist durch die Federaufhängung k mit der Gehäusemasse elastisch verbunden.

Das in jedem Beschleunigungssensor vorhandene Masse-Feder-System ist in diesem Zusammenhang als das Grundmodell für einen schwingungsfähigen Prozess zu verstehen. Damit ist nicht gemeint, dass in jedem Beschleunigungssensor tatsächlich eine Blattfeder verbaut ist. Horn und Höflinger (2014, S. 554–559) benennen z. B. folgende konkrete Bauweisen:

- Kapazitive Beschleunigungssensoren
- Magnetische und induktive Beschleunigungssensoren
- Piezoresistive Beschleunigungssensoren
- Piezoelektrische Beschleunigungssensoren
- Thermische Beschleunigungssensoren
- Optische Beschleunigungssensoren

Alle diese Bauweisen messen die Beschleunigungskraft F_a in Richtung der definierten Achse. Die Beschleunigung a wird dann anhand des 2. Newtonschen Gesetzes berechnet:

$$F_a = m \cdot a \quad (4.4)$$

Laut Wetzstein (2017, S. 1) setzen sich Messwerte von Beschleunigungssensoren aus folgenden Bestandteilen zusammen:

$$\tilde{a} = a_g + a + \eta, \quad \eta \sim \mathcal{N}(0, \sigma^2) \quad (4.5)$$

Der gemessene Wert \tilde{a} setzt sich dabei aus der Gravitation a_g , Beschleunigung durch andere externe Kräfte a und einem Rauschen η (folgt einer Gaußverteilung) zusammen. Ursächlich für die durch den Beschleunigungssensor gemessene Kraft kann somit sowohl eine Beschleunigung (durch Bewegung des Sensors hervorgerufen), als auch die Gravitation sein, falls diese so gerichtet ist, dass ein Anteil auf die definierte Achse entfällt. Das Rauschen ist als Messfehler zu werten.

Um alle Richtungsanteile einer Beschleunigung messen zu können, werden drei einzelne Beschleunigungssensoren benötigt, deren definierte Achsen orthogonal sind (x -, y -, z -Achse). Man spricht in diesem Fall von einem 3-Achsen-Beschleunigungssensor.

4.3.2 Drehratensensor

Ein Drehratensensor (engl. *gyroscope*) misst die Winkelgeschwindigkeit ω um eine definierte Achse in rad/s.

Dazu existieren verschiedene Messprinzipien. Im Folgenden soll die Messung über die Corioliskraft vorgestellt werden, wie sie in den kostengünstigen – und somit stark verbreiteten – Drehratensensoren Anwendung findet. Teurere Sensoren nutzen den Sagnac-Effekt⁴ und sind dadurch etwas präziser.

Die Corioliskraft ist eine Trägheitskraft, die in rotierenden Systemen vorkommt. In einem mit der Winkelgeschwindigkeit ω rotierenden System wirkt auf eine sich mit Geschwindigkeit v bewegende Masse m eine Corioliskraft F_C :

$$\vec{F}_C = 2m(\vec{v} \times \vec{\omega}) \quad (4.6)$$

Die Corioliskraft \vec{F}_C wirkt somit orthogonal zur Richtung der Vektoren \vec{v} und $\vec{\omega}$ (Rechte-Hand-Regel). Mit den im vorangegangenen Abschnitt beschriebenen Beschleunigungssensoren kann dann die Auslenkung der Masse und somit die Corioliskraft gemessen und entsprechend die Winkelgeschwindigkeit berechnet werden.

Die einfachste Ausführungsform eines Drehratensensors stellt dabei ein

⁴siehe Horn und Höflinger (2014) Kapitel 9.4.2.4

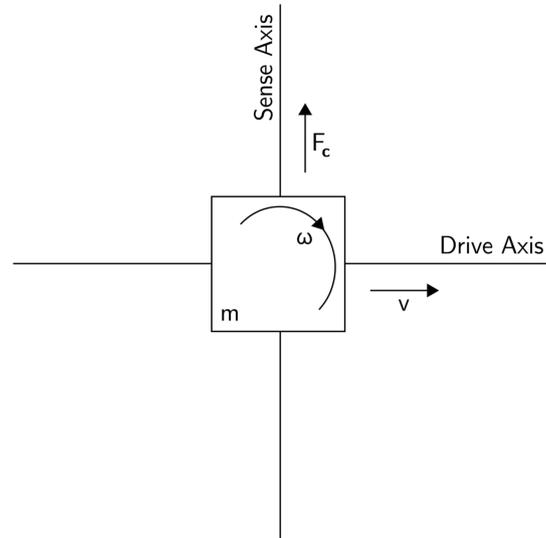


Abbildung 4.6: Corioliskraft bei einem Vibrations-Drehratensensor. Quelle: Woodman (2007, S. 10).

Einmassenschwinger dar, wie er in Abbildung 4.6 schematisch dargestellt ist. Die Masse m wird entlang der Bewegungsachse mit der Geschwindigkeit v zum vibrieren gebracht. Wenn der Drehratensensor nun rotiert wird, entsteht aufgrund der Corioliskraft F_C eine sekundäre Vibration entlang der definierten Sensorachse. Diese wird gemessen, nach Gleichung 4.6 wird die Winkelgeschwindigkeit berechnet. (vgl. Woodman, 2007, S. 9)

Messwerte von Drehratensensoren setzen sich nach Wetzstein (2017, S. 1) zusammen aus:

$$\tilde{\omega} = \omega + b + \eta, \quad \eta \sim \mathcal{N}(0, \sigma^2) \quad (4.7)$$

$\tilde{\omega}$ ist dabei der gemessene Wert, ω ist der tatsächliche Wert, b ist ein Bias und η ist Weißes-Rauschen, das einer Gaußverteilung folgt. Im Gegensatz zu einem Beschleunigungssensor, der wie zuvor beschrieben auch von der Gravitation abhängt, ist ein Drehratensensor an keinerlei äußere Kräfte gebunden.

Entsprechend dem Vorgehen bei den Beschleunigungssensoren müssen auch bei den Drehratensensoren drei einzelne Sensoren verwendet werden,

deren definierte Achsen orthogonal sind, um alle richtungsanteiligen Winkelgeschwindigkeiten einer Rotation zu messen. Man spricht dann von einem 3-Achsen-Drehratensensor. Abweichend von dem hier beschriebenen, sehr vereinfachten Grundprinzip gibt es auch komplexere Bauformen, die ohne diese Kombination von drei Einzelsensoren alle richtungsanteiligen Winkelgeschwindigkeiten messen können.

4.4 Inertiale Messeinheit

Um eine noch genauere Informationsverarbeitung als mit einem einzelnen Sensor möglich zu machen, können unterschiedliche Sensorinformationen kombiniert werden. Das geschieht z. B. in einer inertialen Messeinheit (engl. *Inertial Measurement Unit*, IMU). In diesem Abschnitt wird sowohl Informationsfusion, als auch IMUs im konkreten erklärt.

4.4.1 Informationsfusion

Die am häufigsten verwendeten Sensoren sind sogenannte Mikrosysteme (engl. *Microelectromechanical Systems*, MEMS), kleinste Bauteile (oft nur wenige Mikrometer groß) auf Halbleiter- oder Kunststoffbasis. Solche Bauteile können kostengünstig in Masse hergestellt werden, haben aber deutlich schlechtere Messeigenschaften als aufwändigere und teurere Systeme (vgl. Woodman, 2007, S. 10). Ziel der Sensorfusion ist es, diese Ungenauigkeiten auszugleichen.

Allgemein beschreiben Heinzmann et al. (2014, S. 1209f) den Prozess der Fusion verschiedener Sensorwerte wie folgt:

Informationsfusion lässt sich als Vorgehensweise definieren, um verfügbare Information [...] aus unterschiedlichen Informationsquellen mit dem Ziel zu verknüpfen, neues und/oder hochwertigeres Wissen [...] zu gewinnen. Das Ziel der Informationsfusion ist es, ein Fusionsresultat zu erzeugen, das den interessierenden Sachverhalt (z. B. eine physikalische oder andere Eigenschaft, eine zu treffende Entscheidung über das Vorhandensein eines Objekts oder das Vorliegen eines Ereignisses) „besser“ beschreibt als die von irgendeiner einzelnen Quelle zu erhaltenden Informationen.

Durch Kombination unterschiedlicher Sensortypen können Vor- und Nachteile der einzelnen Systeme ausgeglichen werden und somit ein effizientes

Gesamtsystem erzeugt werden. Aber auch die im vorangegangenen Abschnitt genannte Kombination von drei orthogonal angeordneten Sensoren – man erhält dann einen 3-Achsen-Sensor – stellt eine Form von Sensor- bzw. Informationsfusion dar. Auch wenn ein einzelner Sensor die gewünschte Information nicht liefern kann, kann unter Umständen die Fusion mit einem anderen Sensortyp die gewünschte Information verfügbar machen.

Beispiele für konkrete Algorithmen bzw. Umsetzungen zur Informationsfusion sind *Occupancy Grids* (räumliche Wahrnehmung, Navigation, Bayes'sche Schätzung), *Kalman Filter* (Korrektur von Sensordrift, Fusion von Sensorrohdaten, rekursive Schätzung), *Fuzzy Logic* (qualitative Modellierung und Problemstellung) oder *Neuronale Netze* (Lernen anhand von Beispielen, Klassifikation, Objekterkennung). (vgl. Heinzmann et al., 2014, S. 1207)

Damit Sensordaten kombiniert werden können müssen sie zwei Bedingungen erfüllen: Die Informationen müssen sich auf den gleichen Sachverhalt beziehen, was sowohl einen gleichen Kontextrahmen, als auch zeitliche Integrität mit einbezieht. Gleichzeitig müssen die Informationen auch mit Unsicherheiten behaftet sein. Eine Fusion wäre nicht sinnvoll, wenn die Informationen absolut sichere Fakten wären. (vgl. Heinzmann et al., 2014, S. 1212f)

4.4.2 Lagebestimmung mit Inertialsensoren

Wie alle Sensoren unterliegen sowohl Beschleunigungssensoren, als auch Drehratensensoren Messfehlern, die sich jedoch unterschiedlich auf die Messergebnisse auswirken.

Grundsätzlich messen Beschleunigungssensoren Kräfte auf Massen (siehe Gleichung 4.5) und Drehratensensoren Winkelgeschwindigkeiten (siehe Gleichung 4.7); beide Sensortypen sind jedoch in der Lage die Orientierung zu erfassen.

4.4.2.1 Integration von Drehraten

Da die Lagebestimmung im dreidimensionalen Raum stattfindet, benötigt man drei orthogonal stehende Drehratensensoren. Die drei Sensoren messen also die Winkelgeschwindigkeit in drei Dimensionen:

$$\tilde{\omega} = (\tilde{\omega}_x, \tilde{\omega}_y, \tilde{\omega}_z) \quad (4.8)$$

Integriert man die Winkelgeschwindigkeit $\tilde{\omega}$ nach der Zeit, erhält man den Winkel, der die Lage des Sensors repräsentiert. Die genaue Berechnung mittels Quaternionen ist z. B. bei Wetzstein (2017, S. 8) zu sehen.

Vor allem mechanische Unvollkommenheit bei günstigen Sensoren führt zu kleinen Messfehlern. Durch die Integration wächst dieser Fehler aber über die Zeit. Man spricht dann von *Drift* in den Messergebnissen.

4.4.2.2 Schätzung mittels Beschleunigung

Die gemessenen Werte von Beschleunigungssensoren enthalten sowohl die Beschleunigung durch Gravitation, als auch die Beschleunigung durch externe Kräfte (siehe Gleichung 4.5 in Abschnitt 4.3.1). Erstere macht man sich zu Nutze, um die Lage eines Objekts zu bestimmen. Drei orthogonal stehende Beschleunigungssensoren messen die Beschleunigung:

$$\tilde{a} = (\tilde{a}_x, \tilde{a}_y, \tilde{a}_z) \quad (4.9)$$

Zur Lagebestimmung nutzt man nun nur den Anteil der Gravitation. Dazu nehmen wir an, dass der Beschleunigungssensor keinen anderen Kräften ausgesetzt wird, also nicht bewegt wird. Die Gravitation ist dabei nach oben gerichtet:

$$a_g = (0, 0, 1) \quad (4.10)$$

Die Rotation von \tilde{a} nach a_g repräsentiert die Lage des Sensors. (vgl. Wetzstein, 2017, S. 8f)

Da Beschleunigungssensoren nicht in der Lage sind, die Gravitation losgelöst von anderen äußeren Kräften auf den Sensor zu messen, spricht man nur von einer Schätzung der Lage. Die Messung ist sehr anfällig für Bewegungen z. B. durch Erschütterungen des Sensors.

4.4.2.3 Fusion beider Sensortypen

Wie beschrieben können grundsätzlich beide Sensortypen die Lage eines Objekts erfassen, wenn sie an diesem angebracht werden. Die Nachteile von Drehratensensoren (vor allem Drift) und Beschleunigungssensoren (vor allem Rauschen) sind dabei komplementär und werden durch Sensorfusion sehr gut ausgeglichen (vgl. Wetzstein, 2017, S. 5).

Das kann z. B. sehr einfach mit einem Komplementärfilter realisiert werden, wie es in Abbildung 4.7 schematisch dargestellt ist. Die Messwerte des Beschleunigungssensors werden dabei mittels eines Tiefpassfilters vorverarbeitet, dadurch werden hochfrequente Anteile wie Rauschen entfernt. Die Messwerte des Drehratensensors werden nach dem Integrieren mittels eines Hochpassfilters vorverarbeitet, dadurch wird der Drift verringert. Werden diese Messdaten nun fusioniert, erhält man eine Messkurve, die sehr stabil ist und nahe am tatsächlichen Wert liegt.

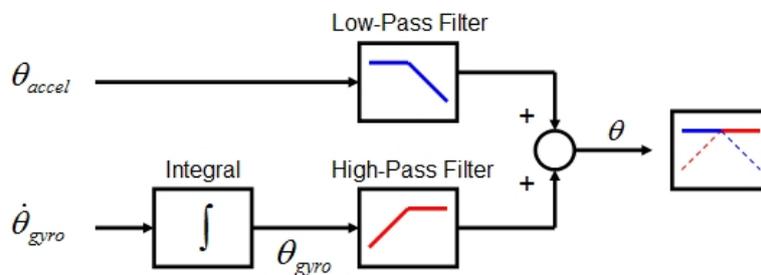


Abbildung 4.7: Komplementärfilter zur Sensorfusion. Quelle: instructables.com (© <http://www.instructables.com/file/F7X6ULMGI0TCNG6/> <13-10-2017>).

Die Messkurven von Beschleunigungssensor (Beschriftung „accelerometer“), Drehratensensor (Beschriftung „gyro“) und den fusionierten Daten mittels Komplementärfilter (Beschriftung „complementary filter“) können in Abbildung 4.8 verglichen werden. Die Drehraten-Messung ist sehr stabil, weicht aber von den anderen Messungen ab (Bias). Die Abweichung ändert sich über die Zeit zudem leicht (Drift). Die Messergebnisse des Beschleunigungssensors sind zwar global betrachtet korrekt, schwanken aber lokal sehr stark (Rauschen).

4.4.3 Bauformen

Inertiale Messeinheiten werden heute – ebenso wie einzelne Inertialsensoren – als Microelectromechanical Systems (MEMS) kostengünstig in hoher Stückzahl in Kleinstbauweise gefertigt. Dadurch sind sie in zahlreichen Elektronikprodukten zu finden: In Smartphones werden sie unter anderem eingesetzt,



Abbildung 4.8: Plot von Roll-Winkeln (gemessen und angenähert). Quelle: Wetzstein (2017, S. 4).

um die Orientierung der Kamera zu erkennen. In modernen Notebooks sind IMUs verbaut, um die Festplatte rechtzeitig abzuschalten, sollte das Gerät versehentlich fallen gelassen werden – so soll Datenverlust verhindert werden.

IMUs bei denen die Inertialsensoren direkt mit dem Rahmen (Platine) verbunden sind, nennt man *Strapdown-IMU*. Die zweite Bauform, bei der die Sensoren auf einer speziellen im Raum stabilisierten Plattform (kardanische Aufhängung, engl. Gimbal) platziert sind, nennt man *IMU mit stabiler Plattform*. Ein solches System ist schematisch in Abbildung 4.10 dargestellt. Die beiden Bauformen unterscheiden sich hauptsächlich durch unterschiedliche Bezugssysteme (siehe Abbildung 4.9): Die Inertialsensoren in Strapdown-IMUs beziehen sich auf das System selbst. In IMUs mit stabiler Plattform, messen die Sensoren im globalen Bezugssystem.

Üblicherweise werden für IMUs Sensoren verwendet, die jeweils drei orthogonal angeordnete Einzelsensoren des selben Typs vereinen, denn nur so sind Messungen im dreidimensionalen Raum möglich. Solch einen 3-Achsen-Sensor bezeichnet man häufig mit 3DoF (Degrees of Freedom). Kombiniert eine IMU z. B. einen 3-Achsen-Beschleunigungssensor mit einem 3-Achsen-Drehratensensor spricht man von 6DoF. Kommt zusätzlich noch ein 3-Achsen-Magnetometer (Kompass) hinzu, wird die IMU als 9DoF (engl. *Degrees of*

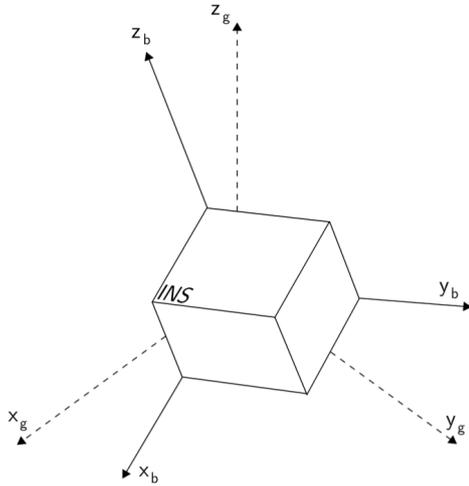


Abbildung 4.9: Ruhendes und rotierendes Bezugssystem. Quelle: Woodman (2007, S. 5).

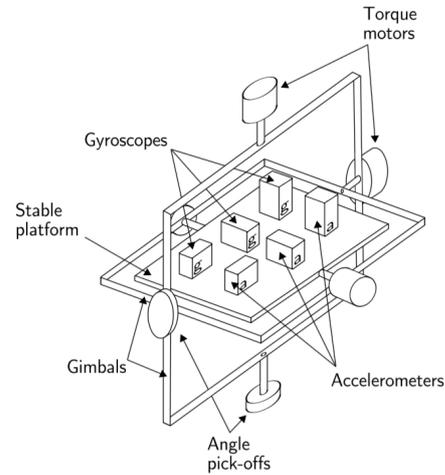


Abbildung 4.10: IMU mit stabiler Plattform. Quelle: Woodman (2007, S. 6).

Freedom, DoF) bezeichnet.

Diese Bezeichnungen mögen vermutlich etwas verwirren, da der Begriff Freiheitsgrad hier nicht in dem in der Mechanik üblichen Sinn verwendet wird (dieser wird in Abschnitt 2.3.1 erläutert). Die Anzahl der Freiheitsgrade, die bei Sensoren angegeben ist, setzt sich aus allen Freiheitsgraden zusammen, die die einzelnen Sensoren messen können, auch wenn der entsprechende mechanische Freiheitsgrad bereits von einem anderen Sensor erfasst wird.

Häufig ist in inertialen Messeinheiten zusätzlich ein Mikrocontroller verbaut. Dieser übernimmt die Vorverarbeitung und die Fusion der Messdaten.

Kapitel 5

Indoor-Lokalisierung

Indoor-Lokalisierung ist die Erfassung der Position eines Objekts im Raum mittels technischer Einrichtungen innerhalb von Gebäuden. Ein Komplettsystem zur Erfassung der Position in Echtzeit nennt man auch Real-Time Locating System (RTLS). Dieses Kapitel behandelt sowohl Messverfahren und Methoden zur Lokalisierung, als auch konkrete Technologien.

5.1 Messverfahren

In diesem Abschnitt werden die wichtigsten Messverfahren vorgestellt: *Laufzeitmessung*, *Laufzeitdifferenzmessung* und *Messung des Einfallswinkels*. Die vorgestellten Verfahren beziehen sich auf die Messung zwischen einer festen Basisstation (Anchor) und einer mobilen Einheit (Tag). Sie bilden die Grundlage für die verschiedenen Lokalisierungsmethoden. Dazu werden später mehrere Messungen mit unterschiedlichen Basisstationen kombiniert (siehe Abschnitt 5.2).

5.1.1 Laufzeitmessung

Bei der Laufzeitmessung wird zwischen *einfacher Laufzeitmessung* (Übertragung vom Sender zum Empfänger) und *Messung der Umlaufzeit* (Übertragung vom Sender zum Empfänger und zurück) unterschieden.

Einfache Laufzeitmessung

Die einfache Laufzeitmessung (engl. *Time of Arrival*, ToA) misst die absolute Zeit für die Übertragung eines Messsignals zwischen Sender und Empfänger.

Beim Senden des Messsignals wird diesem der Timestamp t_{send} angehängt. Beim Empfangen kann dann mit Hilfe des aktuellen Timestamps t_{empf} die Laufzeit t berechnet werden:

$$t = t_{empf} - t_{send} \quad (5.1)$$

Damit das funktioniert ist eine exakte Clock-Synchronisation von Sender und Empfänger notwendig.

Die gemessene Laufzeit ist dabei proportional zur Entfernung. Durch Multiplikation der Laufzeit t mit der Ausbreitungsgeschwindigkeit c_{med} des Messsignals im entsprechenden Medium (z. B. Lichtgeschwindigkeit im Vakuum) erhält man den Abstand s (vgl. Mautz, 2012, S. 30):

$$s = c_{med} \cdot t \quad (5.2)$$

Setzt man Gleichung 5.1 in Gleichung 5.2 ein ergibt sich bei der Laufzeitmessung für die Entfernung s zwischen Sender und Empfänger:

$$s = c_{med} \cdot (t_{empf} - t_{send}) \quad (5.3)$$

Nachteilig an diesem Verfahren ist die nötige Synchronisation von Sender und Empfänger. Aufgrund der geringen Komplexität können die Messungen aber in sehr kurzen Abständen erfolgen. Mit ToA ist es möglich den tatsächlichen Abstand zwischen Sender und Empfänger zu messen. Ab drei Sendern kann mittels Lateration (siehe Abschnitt 5.2.1) die Position bestimmt werden.

Messung der Umlaufzeit

Die Messung der Umlaufzeit (engl. *Round-Trip-Time*, RTT) ist eine Erweiterung der einfachen Laufzeitmessung. Dabei wird die gesamte Zeit für die Übertragung des Messsignals vom Sender zum Empfänger und wieder zurück gemessen. Man nennt diese Messung auch *Two Way Ranging*. Mit Two Way Ranging kann auf die Synchronisierung von Sender und Empfänger verzichtet

werden. Dadurch ist der Einsatz in unkoordinierten Mesh-Netzen mit geringer Komplexität möglich. (vgl. Mautz, 2012, S. 30)

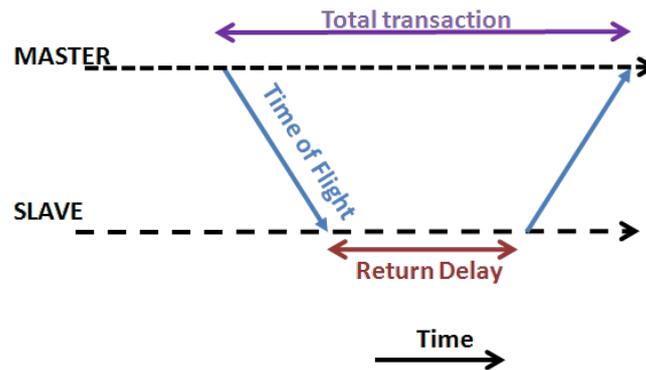


Abbildung 5.1: Two Way Ranging. Quelle: BeSpoon SAS (<http://bespoon.com/two-way-ranging/> <17-10-2017>).

Abbildung 5.1 zeigt schematisch eine Two Way Ranging Messung. Die Gesamtlaufzeit t_{total} (in der Abbildung „Total transaction“) setzt sich aus der doppelten Einzellaufzeit t (in der Abbildung „Time of Flight“) abzüglich des Return Delays t_{delay} zusammen. Das Return Delay entsteht durch die Verarbeitung im Empfänger und wird bei der Antwort mitgeschickt. Der Sender kann den Abstand s demnach – und entsprechend den Annahmen für die Umrechnung von Laufzeit aus Abstand in Gleichung 5.2 – wie folgt berechnen:

$$s = c_{med} \cdot \frac{t_{total} - t_{delay}}{2} \quad (5.4)$$

Obwohl die beiden Clocks nun nicht mehr synchronisiert sein müssen, spielt es im vorangegangenen Beispiel noch eine Rolle, wie präzise beide Clocks sind. Läuft eine nur ein wenig langsamer oder schneller, werden die Messergebnisse verfälscht. Abhilfe schafft eine erweiterte Variante der Two Way Ranging Methode, bei der der Empfänger die Antwort doppelt sendet. Abbildung 5.2 zeigt diese Methode als schematische Darstellung. Der Abstand s zwischen Sender und Empfänger würde in diesem Fall – die Annahmen für die Umrechnung von Laufzeit in Abstand aus Gleichung 5.2

mit einbeziehend – folgendermaßen berechnet:

$$s = c_{med} \cdot \frac{t_{total} - 2 \cdot t_{delay}}{2} \quad (5.5)$$

Durch den Wegfall der Synchronisation der einzelnen Sender und Empfänger ist ein System nach dem Two Way Ranging Verfahren deutlich einfacher (und somit kostengünstiger) zu realisieren. Allerdings benötigt eine Messung deutlich mehr Zeit, als beim ToA Verfahren. Daher sind beim Two Way Ranging auch nur geringere Wiederholraten möglich.

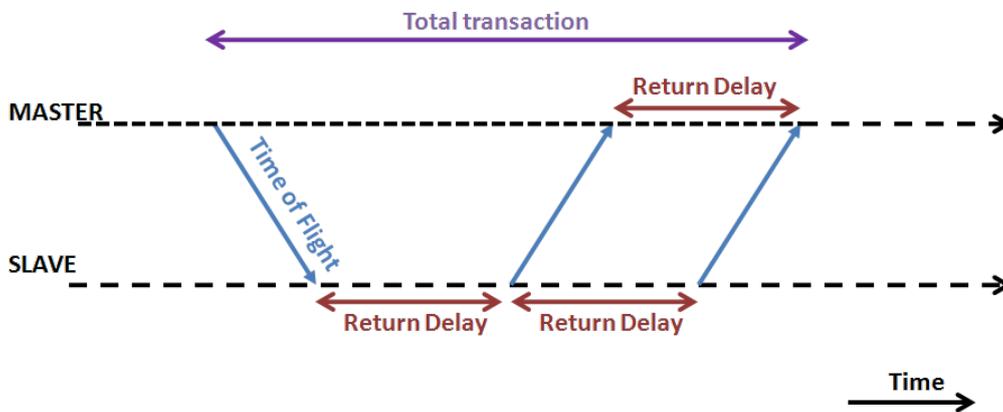


Abbildung 5.2: Two Way Ranging mit doppelter Antwort. Quelle: BeSpoon SAS (<http://bespoon.com/two-way-ranging/> <17-10-2017>).

5.1.2 Laufzeitdifferenzmessung

Die Laufzeitdifferenzmessung (engl. *Time Difference of Arrival*, TDoA) basiert auf dem ToA Verfahren. Durch Subtraktion einzelner ToA Messungen erhält man Laufzeitdifferenzen. Mit zwei Sendern mit bekannter Position erhält man somit eine Hyperbel auf der Punkte mit gleicher Laufzeit liegen. Da ausschließlich diese Differenzen betrachtet werden, kann auf die Synchronisierung der Clock des Empfängers verzichtet werden. Die Sender untereinander müssen aber weiterhin synchronisiert sein. (vgl. Mautz, 2012, S. 30)

Mit mindestens drei Sendern kann man durch Hyperbelschnitt die exakte Position bestimmen. Es ist aber nicht möglich mit nur einem Sender und einem

Empfänger eine Messung des tatsächlichen Abstandes durchzuführen. Das TDoA Verfahren ist daher nur anwendbar, wenn die Relation von Abständen zu mehreren Sendern gemessen und dadurch die Position bestimmt werden soll.

5.1.3 Messung des Einfallswinkels

Einen gänzlich anderen Ansatz verfolgt die Messung des Einfallswinkels (engl. *Angle of Arrival, AoA*) Verfahren. Mittels eines Antennenarrays – und den dort gemessenen Laufzeitunterschieden beim Empfang des Messsignals – wird der Einfallswinkel des Signals am Empfänger und somit die Richtung des Senders bestimmt. Abbildung 5.3 zeigt das Funktionsprinzip der Messung des Einfallswinkels mit einem Antennenpaar. Detektiert man das Messsignal mit mehreren Empfängern kann mittels Angulation (siehe Abschnitt 5.2.2) die Position bestimmt werden.

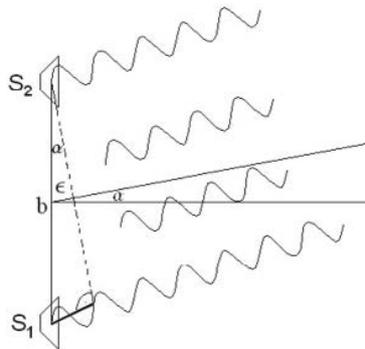


Abbildung 5.3: Messung des Einfallswinkels. Quelle: Blanckenbach, Norrdine, Schlemmer und Willert (2007, S. 171).

5.2 Lokalisierungsmethoden

Die Bestimmung der Position eines Objekts ist relativ komplex, weil es keinen Sensor gibt, der direkt eine Position bestimmen kann. Erst durch Kombination von mehreren Messungen kann die Position errechnet werden.

Die wichtigsten Lokalisierungsmethoden sind dabei die *Lateration* von Zeit- bzw. Abstandsmessungen und die *Angulation* von Winkelmessungen. Diese

beiden Verfahren werden im Folgenden ausführlich erläutert. Sie bedienen sich den im vorangegangenen Abschnitt vorgestellten Messverfahren und haben eine relativ hohe Genauigkeit. Unter „Umgebungsanalyse“ wird die Positionsbestimmung mittels *Herkunftszelle* und mittels *Szenenanalyse* kurz vorgestellt.

5.2.1 Trilateration

Die Trilateration ist ein mathematisches Verfahren um aus Abständen zu bekannten Punkten einen unbekanntem Punkt zu bestimmen. Abstände können z. B. mittels Laufzeitmessung (siehe Abschnitt 5.1.1) gemessen werden. Mit drei Abständen zu bekannten Punkten kann man eine Position – zumindest theoretisch – eindeutig bestimmen. In der Praxis spielen jedoch weitere Faktoren wie Messungenauigkeiten eine Rolle.

Abbildung 5.4 zeigt eine ideale mathematische Trilateration. Der Punkt $P(x, y)$ kann mittels der drei Abstände d_i zu den bekannten Punkten $A_i(x_i, y_i)$ bestimmt werden. Geometrisch betrachtet schneiden sich alle drei Kreise genau in Punkt P .

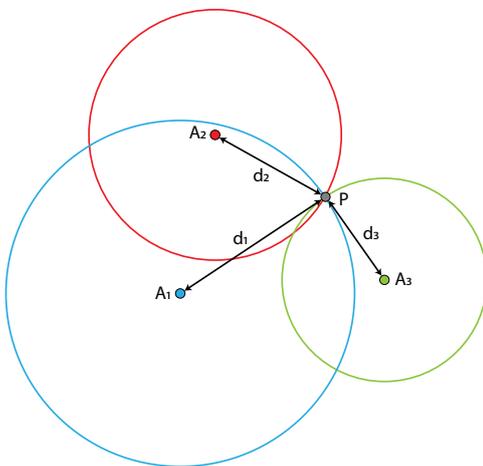


Abbildung 5.4: Ideale Trilateration.

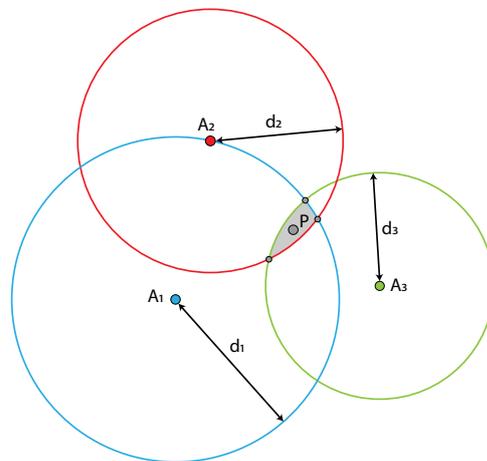


Abbildung 5.5: Trilateration mit ungenauen Messwerten.

Entsprechend der Formel für die Kreisgleichung kann folgendes Gleichungs-

system aufgestellt werden (vgl. Shchekotov, 2015, S. 188):

$$\begin{aligned}d_1^2 &= (x - x_1)^2 + (y - y_1)^2 \\d_2^2 &= (x - x_2)^2 + (y - y_2)^2 \\d_3^2 &= (x - x_3)^2 + (y - y_3)^2\end{aligned}\tag{5.6}$$

Theoretisch ist dieses Gleichungssystem eindeutig lösbar. In der Realität schwanken die Messergebnisse. Somit ist nicht mit einer exakten Lösung zu rechnen. Wie Abbildung 5.5 zeigt, kann bei Abweichung nicht mehr exakt ein Punkt bestimmt werden. Der Punkt kann aber in der Fläche zwischen den Schnittpunkten von jeweils zwei Kreisen vermutet werden.

5.2.2 Triangulation

Bei der Triangulation lässt sich anhand zwei bekannter Punkte A_i und den Winkeln dieser Punkte zum gesuchten Punkt die Position von Punkt P berechnen. In Abbildung 5.6 sind die Punkte A_1 und A_2 , sowie die Winkel α_1 und α_2 gegeben.

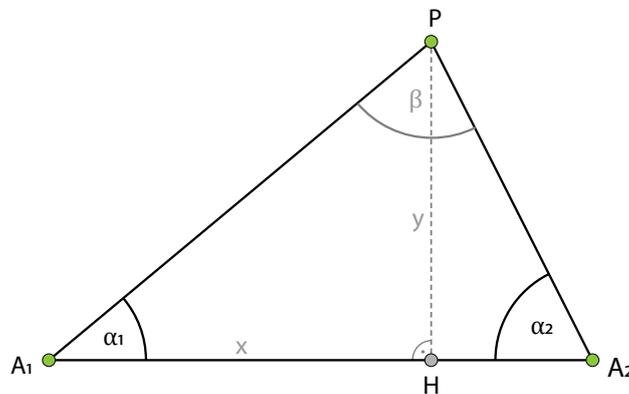


Abbildung 5.6: Triangulation.

Mit Hilfe des Sinussatzes und der Winkelsumme im Dreieck kann die Strecke $\overline{A_1P}$ berechnet werden:

$$\overline{A_1P} = \frac{\overline{A_1A_2} \cdot \sin(\alpha_2)}{\sin(\beta)}, \quad \beta = 180^\circ - \alpha_1 - \alpha_2\tag{5.7}$$

Im Hilfsdreieck $\triangle A_1PH$ lässt sich nun x und y mit Sinus und Kosinus berechnen:

$$\begin{aligned}x &= \cos(\alpha_1) \cdot \overline{A_1P} \\y &= \sin(\alpha_1) \cdot \overline{A_1P}\end{aligned}\tag{5.8}$$

5.2.3 Umgebungsanalyse

Weitere Methoden sind die Bestimmung der Herkunftszelle und die Szenenanalyse. Beide Methoden erreichen durch Analyse der Umgebung die Positionsbestimmung.

Herkunftszelle

Weisen die Basisstationen eine Zellenstruktur auf, ist eine Positionsbestimmung mittels Herkunftszelle (engl. *Cell of Origin*) möglich. Die mobile Einheit verbindet sich jeweils mit der nahegelegensten Basisstation. Nun kann angenommen werden, dass sich die mobile Einheit auch im Bereich der Basisstation aufhält. Mit diesem System ist – abhängig von der Dichte der Zellenstruktur – nur eine relativ geringe Genauigkeit zu erzielen. (vgl. Mautz, 2012, S. 31)

Denkbar ist ein solches System z. B. in einem WLAN-Netzwerk. Es ist anzunehmen, dass sich das Endgerät im Bereich des Access-Points – und somit z. B. dem entsprechenden Gebäudeteil – befindet, bei dem es angemeldet ist.

Szenenanalyse

Bei der Szenenanalyse (engl. *Scene Analysis*) werden aktuelle Messdaten (meist die Signalstärke von vorhandenen Funknetzen) mit Referenzwerten in einer Datenbank verglichen. Die Daten dieser Datenbank müssen entsprechend zuvor erhoben worden sein. Bei Übereinstimmung von aktuellem Messwert und Referenzwert, kann die Position schlussgefolgert werden. Auch hier ist die Genauigkeit relativ gering.

5.3 Technologien

In diesem Abschnitt werden geeignete Signale für die Messung und die daraus resultierenden konkreten Technologien zur Indoor-Lokalisierung vorgestellt.

5.3.1 Geeignete Signale

Für die Positionsbestimmung im Gebäude sind laut Retscher und Kistenich (2006, S. 25) *Infrarot*, *Ultraschall*, *Radiosignale* und *sichtbares Licht* die gebräuchlichsten Signale um Messungen durchzuführen.

Infrarot ist elektromagnetische Strahlung im Spektrum zwischen dem Roten Licht und Mikrowellen (Wellenlänge 0,8–300µm). Es breitet sich mit Lichtgeschwindigkeit aus, hat eine Reichweite von wenigen Metern und wird an Oberflächen reflektiert. Infrarotsignale eignen sich also nicht für Messungen über mehrere Räume hinweg. Durch die Reflexion ist eine stabile Anwesenheitserkennung (auch um Objekte herum) möglich, eine Abstandsmessung ist (aufgrund von Signalüberlagerung) hingegen relativ schwierig und ungenau. (vgl. Retscher & Kistenich, 2006, S. 26)

Ultraschall ist Schall oberhalb der Hörschwelle (Frequenz 20kHz–1000MHz). Ultraschallwellen werden an Hindernissen reflektiert. Die Reichweite ist abhängig von der Erzeugungsenergie, liegt aber im Bereich weniger Meter. Da sich Schall deutlich langsamer ausbreitet als Infrarot oder Radiosignale, ist es möglich bei gleicher zeitlicher Auflösung genauere Messergebnisse zu erhalten. (vgl. Retscher & Kistenich, 2006, S. 26)

Radiowellen (auch Funkwellen) sind elektromagnetische Schwingungen im Frequenzspektrum von 75 Hertz bis etwa 10 Gigahertz. Nutzt man die Radiowellen als Trägerwellen (z. B. mit Amplituden- oder Frequenzmodulation), können dadurch Informationen übertragen werden. Elektromagnetische Wellen können Wände und Hindernisse durchdringen, dadurch ist auch eine Lokalisierung über mehrere Räume hinweg möglich. (vgl. Retscher & Kistenich, 2006, S. 26)

Sichtbares Licht ist elektromagnetische Strahlung mit Wellenlängen zwischen 400 und 750 Nanometern. Da es tagsüber überall vorhanden ist, muss es nicht – wie die anderen vorgestellten Signale – künstlich erzeugt werden. Es ist dadurch aber auch nur für spezielle Messverfahren geeignet. (vgl. Retscher & Kistenich, 2006, S. 26)

Eine detaillierte grafische Übersicht über die verschiedenen Technologien in Abhängigkeit von Genauigkeit, Reichweite und Wellenlänge ist bei Mautz (2012, S. 10) zu finden.

5.3.2 Vergleich verschiedener Systeme

So unterschiedlich wie die Messverfahren und geeigneten Messsignale, genauso unterschiedlich gestalten sich auch die konkreten Technologien. Verschiedene Systeme zur Positionserkennung in Gebäuden werden in Tabelle 5.1 verglichen.

Tabelle 5.1: Vergleich von Positionierungssystemen in Gebäuden. Basiert auf: Retscher und Kistenich (2006, S. 33).

Systemname	Signal	Methode	Absolute Positionierung	Relative Positionierung	Positionierung	Tracking	Kosten	Genauigkeit [m]
Active Badge	IR	CoC	✓			✓	niedrig	Raum
WIPS	IR	CoC	✓		✓		niedrig	Raum
Active Bat	US	ToA	✓			✓	niedrig	0,1
Cricket	US	LM	✓	✓	✓		niedrig	1,2
A-GPS	RF	LM	✓		✓		hoch	20-25
Locata	RF	LM	✓		✓		hoch	0,1–1
Radar	RF	SS	✓		✓	✓	hoch	3–4
UWB	RF	LM/LDM	✓		✓		hoch	0,2
Bluetooth	RF	CoO	✓		✓	✓	mittel	10
SpotON	RF	SS	✓	✓	✓	✓	mittel	1 m ³
RFID	RF	CoO		✓		✓	niedrig	1–20
GSM	RF	LDM/AoA	✓			✓	niedrig	50–100
CyberCode	SL	DB		✓	✓		mittel	variabel
Ubitrack	SL	DB		✓		✓	N/A	N/A
EasyLiving	SL	DB	✓			✓	hoch	variabel

Folgende Abkürzungen werden in Tabelle 5.1 verwendet:

Signale

IR	Infrarot
US	Ultraschall
RF	Radiosignale
SL	Sichtbares Licht

Methoden der Positionsbestimmung

CoO	Cell of Origin (Angabe der Herkunftszelle)
LM	Laufzeitmessung
LDM	Laufzeitdifferenzmessung
AoA	Angle of Arrival (Messung des Einfallswinkels)
SS	Signalstärkemessung
DB	Digitale Bilder

N/A Information nicht verfügbar

Die Systeme unterscheiden sich zum Teil deutlich in der erreichbaren Genauigkeit. Für diese Arbeit ist vor allem die UWB-Technologie interessant. Sie ermöglicht eine absolute Positionsbestimmung auf etwa 20 Zentimeter genau.

5.3.3 Ultra-Breitband

Ultra-Breitband (engl. *Ultra Wide Band*, UWB) bezeichnet eine sehr breitbandige Funktechnologie. Im Gegensatz zur klassisch vorherrschenden Aufteilung des Funkspektrums in schmale Frequenzbänder, belegt UWB ein sehr breites Spektrum. Es nutzt dabei aber viel geringere Leistungen. Dadurch kann es bereits genutzte Frequenzbereiche überlagern und benötigt somit kein eigenes Frequenzspektrum. Dieses Grundprinzip von UWB ist in Abbildung 5.7 dargestellt. (vgl. Eisenacher, 2006, S. 1–3)

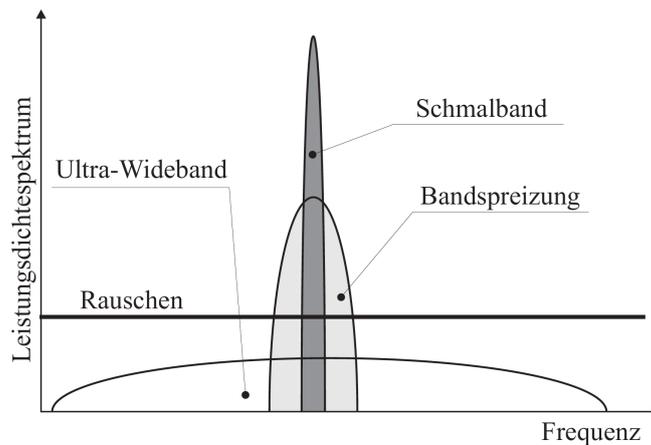


Abbildung 5.7: Prinzip von Ultra-Breitband. Quelle: Eisenacher (2006, S. 2)

Grundsätzlich sind bei UWB zwar unterschiedliche Implementierungen denkbar, der Quasistandard ist aber die Nutzung von sehr kurzen Impulsen (< 2 ns) zur Übertragung. In diesem Fall spricht man auch von ImpulseRadio-UWB (IR-UWB). Aufgrund der großen Bandbreite und dieser extrem kurzen Impulse sind auch sehr große Datenraten möglich. (vgl. Blanckenbach, Norrdine, Schlemmer und Willert, 2007, S. 172f; Eisenacher, 2006, S. 12f)

Durch die extrem große Bandbreite weißt UWB laut Blanckenbach et al. (2007, S. 173) folgende vier weiteren Merkmale auf:

- *Unempfindlichkeit gegenüber Mehrwegeausbreitung*¹: Durch die extrem kurzen Impulse können Signale, die aufgrund von Mehrwegeausbreitung nacheinander ankommen, gut voneinander getrennt werden.
- *Durchdringung von Materialien*: Da im UWB-Signal viele verschiedene Frequenzen enthalten sind, kann das Signal viele verschiedene Materialien durchdringen.
- *Störungssicherheit*: Durch die geringe Leistung sind UWB-Signale für andere Funksysteme im Grundrauschen kaum wahrnehmbar.
- *Hardware*: UWB-Systeme brauchen keine Trägerwelle und sind daher deutlich einfacher zu realisieren.

Gerade durch diese Eigenschaften eignet sich UWB zum Einsatz für Abstandsmessung bzw. Positionsbestimmung in Gebäuden. Dazu wird meist die Laufzeit oder die Umlaufzeit des UWB-Signals gemessen (siehe Abschnitt 5.1.1).

¹Mehrwegempfang tritt auf, wenn ein Signal auf mehreren Wegen – z. B. durch Reflexion an Wänden – beim Empfänger ankommt.

Kapitel 6

Umsetzung des Projekts

Im Rahmen dieser Arbeit wurde eine binaurale Rauminstallation konzipiert und umgesetzt. In diesem Kapitel wird sowohl das System und die Komponenten detailliert betrachtet, als auch auf Lösungsansätze für Schwierigkeiten bei der Umsetzung eingegangen.

6.1 Funktionsprinzip

Der Benutzer des Systems soll sich mittels eines Kopfhörers und zusätzlicher Sensorik in einem Raum in einer virtuellen, auditiven Umgebung bewegen können. Das dazu entwickelte System kann grob in drei Teile aufgeteilt werden:

- *Sensorsystem*: Misst die Orientierung des Kopfhörers und Entfernungsdaten zu bekannten Basisstationen.
- *Datenverarbeitung*: Aufarbeitung der Messwerte (z. B. Glättung, Anpassung von Wertebereichen, Trilateration).
- *Audio-Rendering*: Synthese der binauralen Audiosignale in Abhängigkeit der Hörerposition.

Diese Einteilung in der Systemarchitektur ist auch in der Übersicht in Abbildung 6.1 zu erkennen. Das Sensorsystem – welches an den Bügel des Kopfhörers montiert ist – erfasst dabei sowohl die Orientierung des Kopfes, als auch dessen Position im Raum. Für die Positionsbestimmung sind zusätzlich noch drei entfernte UWB-Basisstationen mit bekannter Position nötig. Die ermittelten Daten werden via Bluetooth an einen Rechner gesendet. Dort

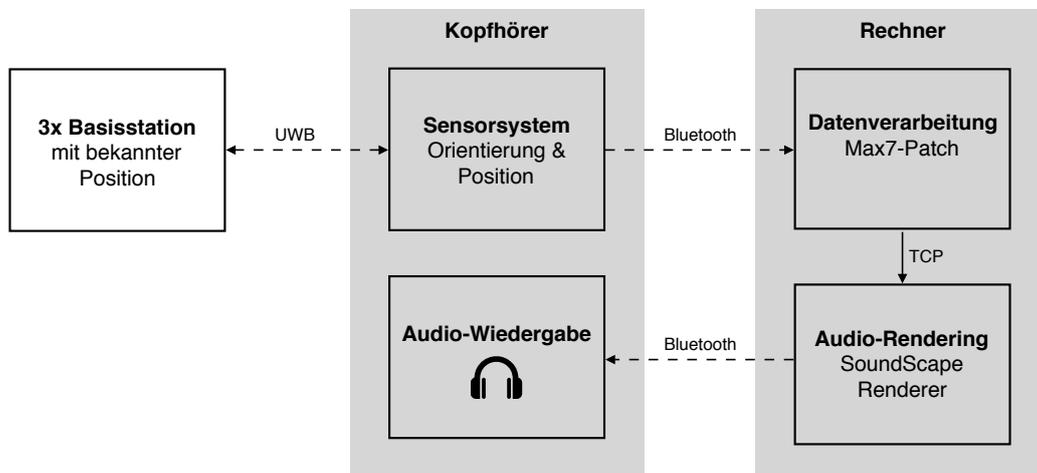


Abbildung 6.1: Schematische Darstellung des Gesamtsystems.

übernimmt ein Programm, das in der visuellen Programmierumgebung Max7 erstellt wurde, die Datenverarbeitung. Diese umfasst vor allem die Anpassung von Wertebereichen, die Glättung und Filterung der Messwerte und die Trilateration der Entfernungsdaten. Die verarbeiteten Daten werden via TCP innerhalb des Rechners an die Software SoundScape Renderer geschickt und können dort die Hörerposition fernsteuern. SoundScape Renderer synthetisiert entsprechend aus einer vorproduzierten Audioszene die binauralen Audiosignale (Ohrsignale), welche – ebenfalls via Bluetooth – an den Kopfhörer gestreamt und dort wiedergegeben werden.

Räumlich betrachtet gibt es ebenfalls mehrere Bereiche, auch diese sind in Abbildung 6.1 angedeutet. Der erste Bereich ist der Kopfhörer mit den angebauten Sensoren. Dieser ist kabellos und mobil ausgeführt. Als zweiter Bereich gelten die drei Basisstationen. Sie sind an feste bekannte Positionen gebunden. Den dritten Bereich stellt der Rechner dar, der prinzipiell keine Anforderungen an eine bestimmte Position stellt, solange er sich in Bluetooth-Reichweite des Kopfhörers befindet. Die Kommunikation zwischen den räumlichen Einheiten erfolgt drahtlos.

6.2 Sensorsystem

Das Sensorsystem erfasst die Orientierung und die Position des Kopfes im Raum. Dazu beinhaltet es zwei separate Tracking Systeme. Eine inertielle Messeinheit zur Messung der Orientierung (Head-Tracking) und ein System für die Positionsbestimmung (Indoor-Lokalisierung). Zudem behandelt dieser Abschnitt auch die Datenübermittlung von den einzelnen Sensoren, über die Mikrocontroller, bis hin zur Bluetooth-Übertragung an den Rechner.

Abbildung 6.2 zeigt den verwendeten Kopfhörer (Sennheiser MB660) mit den am Bügel befestigten zusätzlichen Bauteilen: Links ist der Localino RTLS Tag für die Positionsbestimmung angebracht, in der Mitte befindet sich eine Sparkfun Razor IMU für die Bestimmung der Orientierung und rechts ist ein Bluetooth-Modul und ein Lithium Polymer (LiPo) Akku montiert.

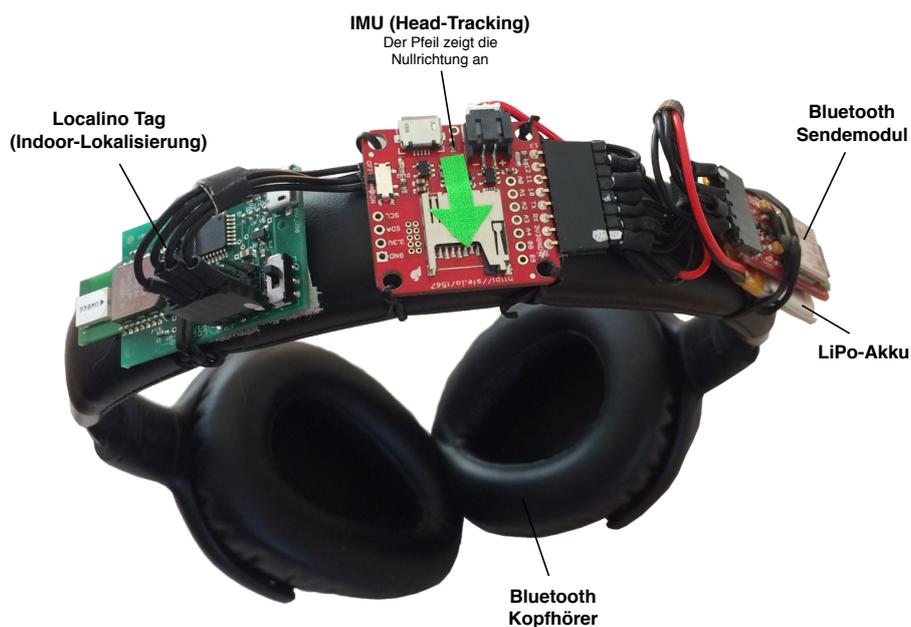


Abbildung 6.2: Kopfhörer mit angebaute Sensoreinheit.

6.2.1 Mikrocontroller

Im Sensorsystem stellen Mikrocontroller die Kommunikationsschnittstellen bereit. Die Programmierung dieser Mikrocontrollern bringt jedoch einige

Besonderheiten mit sich.

Mikrocontroller vereinen einen Prozessorkern, Speicher und Ein-/Ausgabeschnittstellen in einem einzelnen Chip, mit dem Ziel Steuerungs- und Kommunikationsaufgaben möglichst einfach zu lösen (vgl. Brinkschulte & Ungerer, 2002, S. 1). Die Leistung ist meist relativ gering und an den Verwendungszweck angepasst, daher sind sie recht preisgünstig. Der Speicher des Mikrocontrollers wird mit einem bestimmten Code bespielt, den der Chip anschließend ausführt. Bei den Speichern kommen meist entweder Nur-Lese-Speicher (ROM) oder programmierbare Nur-Lese-Speicher (EPROM, EEPROM) zum Einsatz. Eher selten finden auch Flash-Speicher Verwendung.

Die Programmierung bei Mikrocontrollern geschieht sehr nahe an der Hardware (meist speziell für einen Controller bzw. eine Architektur). Je nach Hardware ist ein spezieller Programmierer (Kombination aus Hardware und Software, die die Programmierung des Mikrocontrollers ermöglicht) notwendig. Da die Systeme in sich geschlossen sind, gibt es praktisch keine Möglichkeit für direktes Debugging des Codes. Die Ausgabe von Fehlermeldungen ist ebenfalls sehr eingeschränkt.

Meistens werden Quellcodes für Mikrocontroller in der Programmiersprache C geschrieben. Die Programmierung erfolgt sequentiell, da die Hardware in den meisten Fällen keine Nebenläufigkeit (Multi-Threading) unterstützt. Meistens gibt es eine Hauptfunktion die wiederholt ausgeführt wird.

Arduino

Arduino ist ein Open-Source Projekt das sowohl Hardware, als auch Software anbietet, die einen leichten Einstieg in die Programmierung von Mikrocontrollern bietet. Die Arduino Hardware kommt dabei mit einem speziellen Bootloader, einem vorinstallierter Code im Speicher des Mikrocontrollers, der die weitere Programmierung ohne speziellen Programmierer über eine UART-Verbindung (meist per USB) erlaubt. Die Arduino Entwicklungsumgebung (IDE) vereinheitlicht und vereinfacht zudem die Programmierung, da der Code automatisch hardwarespezifisch kompiliert wird und die IDE bereits eine große Zahl an Funktionen bereitstellt.

6.2.2 Head-Tracking

Das Head-Tracking wird mittels einer inertialen Messeinheit (siehe Kapitel 4) realisiert. Als Hardware dient eine *SparkFun 9DoF Razor IMU M0* Platine¹. Auf dieser sind folgende Komponenten verbaut:

- *Atmel Mikrocontroller* (SAM21): Kommunikation zwischen den Komponenten und nach Außen
- *9DoF-Sensor* (MPU-9250): Sensoreinheit mit einem 3-Achsen Gyroskop, einem 3-Achsen Accelerometer und einem 3-Achsen Magnetometer
- *Micro-USB Port*: Spannungsversorgung der Platine und Programmierung des Mikrocontrollers
- *LiPo-Laderegler*: Anschluss und integrierter Laderegler für einen LiPo Akku zur mobilen Spannungsversorgung
- *SD-Karten Slot* (in diesem Projekt nicht verwendet)

Der Mikrocontroller ist Arduino kompatibel und wird ab Werk bereits mit dem Arduino Bootloader ausgeliefert. Somit kann er direkt über den USB Port mit der Arduino IDE programmiert werden. Sparkfun bietet zudem eine passende Arduino Library, welche das Auslesen des 9DoF-Sensors extrem vereinfacht. Ein Teil der Library ist ein sogenannter digitaler Bewegungsprozessor (engl. Digital Motion Processor, DMP). Mit diesem Teil der Library kann die Fusion der Sensordaten konfiguriert und durchgeführt werden.

Zwar kann auch die Sparkfun Library direkt Euler Winkel berechnen und ausgeben, die berechneten Werte stimmen jedoch nicht mit den tatsächlichen Winkeln überein. Hier liegt ein Fehler in der Library vor, der jedoch durch eigene Umrechnung von Quaternionen in Euler Winkel umgangen werden kann².

Mit der folgenden Funktion können aus Quaternionen die entsprechenden Euler Winkel berechnet werden:

```
1 static void toEulerianAngle(float w, float x, float y, float z, float&  
    roll, float& pitch, float& yaw)  
2 {
```

¹Produktseite des Herstellers Sparkfun: <https://www.sparkfun.com/products/14001>

²Im GitHub Repository der Library ist der Bug bereits bekannt: https://github.com/sparkfun/9DOF_Razor_IMU/issues/9 (Zugriff 15. November 2017). Ein Workaround für den Bug ist im SparkFun Electronics Forum zu finden: <https://forum.sparkfun.com/viewtopic.php?f=14&t=45516> (Zugriff 15. November 2017).

```

3  double ysqr = y * y;
4
5  // roll (x-axis rotation)
6  double t0 = +2.0 * (w * x + y * z);
7  double t1 = +1.0 - 2.0 * (x * x + ysqr);
8  roll = atan2(t0, t1) * (4068 / 71);
9
10 // pitch (y-axis rotation)
11 double t2 = +2.0 * (w * y - z * x);
12 t2 = t2 > 1.0 ? 1.0 : t2;
13 t2 = t2 < -1.0 ? -1.0 : t2;
14 pitch = asin(t2) * (4068 / 71);
15
16 // yaw (z-axis rotation)
17 double t3 = +2.0 * (w * z + x * y);
18 double t4 = +1.0 - 2.0 * (ysqr + z * z);
19 yaw = atan2(t3, t4) * (4068 / 71);
20 }

```

Die Funktion folgt dabei den Gleichungen von Blanco (2013, S. 14) für die Umrechnung von Quaternionen in Euler Winkel:

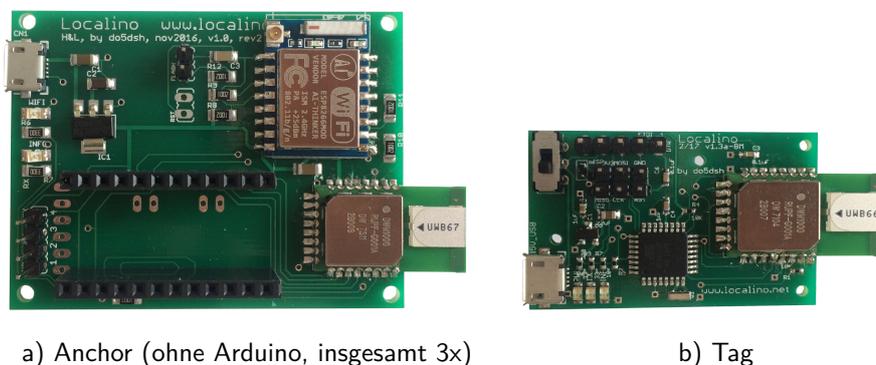
$$\begin{aligned}
 \psi &= \tan^{-1} \left(2 \frac{q_r q_z + q_x q_y}{1 - 2(q_y^2 + q_z^2)} \right) \\
 \vartheta &= \sin^{-1} \left(2(q_r q_y - q_x q_z) \right) \\
 \varphi &= \tan^{-1} \left(2 \frac{q_r q_x + q_y q_z}{1 - 2(q_x^2 + q_y^2)} \right)
 \end{aligned} \tag{6.1}$$

Außerdem werden die Ergebnisse vom Bogenmaß ins Gradmaß umgerechnet. Das geschieht mit dem Faktor $\frac{180^\circ}{\pi}$, jedoch in der für Computer leichter zu verarbeitenden, angenäherten Form $\frac{4068}{71}$. Der komplette Quellcode für das Head-Tracking ist auf dem beiliegenden Datenträger einsehbar (siehe Anhang A).

Die Positionsbestimmung gestaltet sich aufgrund der vorgestellten SparkFun Library recht einfach. Auf aufwendige Kalibrierung der einzelnen Sensortypen kann ebenfalls verzichtet werden, da durch die Sensorfusion in der Library recht stabile Ergebnisse erzielt werden können.

6.2.3 Indoor-Lokalisierung

Die Basis für die Indoor-Lokalisierung stellt ein RTLS System (siehe Kapitel 5) auf UWB-Basis dar. Wie bereits mehrfach erwähnt benötigt man dafür drei Basisstationen (Anchor) und eine mobile Einheit (Tag). Gemessen wird dann die Entfernung vom Tag zu den Basisstationen. Die Positionsbestimmung mittels Trilateration erfolgt später im Rechner.



a) Anchor (ohne Arduino, insgesamt 3x)

b) Tag

Abbildung 6.3: Localino RTLS Hardware.

Als Hardware kommen *Localino v1.3* Platinen³ zum Einsatz, diese sind in Abbildung 6.3 zu sehen. Für den Tag umfasst die Platine dabei folgende Komponenten:

- *UWB Chip* (DW1000): Kommunikation mit den Anchn zur Entfernungsbestimmung mittels TWR
- *Atmel Mikrocontroller* (ATmega 328p): Kommunikation zwischen den Komponenten und nach Außen
- *USB Port*: nur Spannungsversorgung der Platine (Programmierung des Mikrocontrollers mittels FTDI-/ICSP-Header)

Die Platine für die Anchor beinhaltet folgenden Komponenten:

- *UWB Chip* (DW1000): Kommunikation mit dem Tag zur Entfernungsbestimmung mittels TWR
- *Mikrocontroller*: Steckplatz für ein Arduino Pro Mini, Kommunikation zwischen den Komponenten und nach Außen

³Produktwebsite des Herstellers Heuel & Löher: <https://www.localino.net>

- *USB Port*: nur Spannungsversorgung der Platine (Programmierung des Mikrocontrollers mittels FTDI-Header)
- *WLAN Chip* (in diesem Projekt nicht verwendet)

Im Gegensatz zur Head-Tracking Platine, die komplett bestückt geliefert wird, müssen bei den Localino Platinen erst die einzelnen Bauteile eingelötet werden. Da es sich dabei hauptsächlich um SMD-Bauteile handelt, ist das eine anspruchsvolle, aber spannende Herausforderung.

Programmierung der Mikrocontroller

Auch die Hardware für die Indoor-Lokalisierung muss zuerst mit passendem Quellcode programmiert werden. Da auch der Mikrocontroller auf dem Tag Arduino kompatibel ist, kann sowohl für die Programmierung der Anchor – das ist ja Arduino Hardware (Arduino Pro Mini) – als auch für die des Tags die Arduino IDE benutzt werden. Auf den Mikrocontroller des Tags muss dazu zuvor mit einem Programmer (hier ein ISP Programmer) der Arduino Bootloader gebrannt werden. Die anschließende Programmierung wird auch hier durch Einbindung einer passenden Arduino Library vereinfacht. Die DW1000 Library implementiert das Kommunikationsprotokoll für die Verbindung zwischen Mikrocontroller und UWB Chip. Der Chip misst den Abstand zu anderen Chips in Reichweite mittels TWR.

Die Implementierung der Abstandsmessungen ist dabei überschaubar, weil die Library viel Arbeit abnimmt und einigen Minimalbeispiele enthält. Im Quellcode der Anchor muss beachtet werden, dass für jede Basisstation eine eigene ID vergeben wird, da sie nur so später unterschieden werden können. Die Ergebnisse der einzelnen Abstandsmessungen können prinzipiell sowohl am jeweiligen Anchor, als auch am Tag ausgelesen werden. Im konkreten Projekt werden sie jedoch nur am Tag (also am Kopfhörer) ausgelesen und weiterverarbeitet. Die Anchor brauchen somit keine weitere Datenverbindung, eine Stromversorgung genügt.

Nach einigen ersten Funktionstests konnte keine vollständige Funktionsfähigkeit hergestellt werden. Die Abstandsmessung mit bis zu zwei Basisstationen funktioniert ohne Probleme. Sobald jedoch ein dritter Anchor in Reichweite ist, kommt keine durchgehende Kommunikation mehr zustande und die Abstandsmessungen stocken. Nach einiger Recherche konnte der

Fehler zwar nicht direkt lokalisiert werden, das Problem aber durch einen Workaround behoben werden⁴. Dazu ist es nötig, den Quellcode nicht über die Arduino IDE auf die Mikrocontroller aufzuspielen, sondern den Quellcode mit Atmel Studio (IDE für Atmel Mikrocontroller) zu programmieren. Nach der Neuprogrammierung mittels Atmel Studio funktioniert auch die Abstandsmessung mit drei Basisstationen.

Der vollständige Quellcode ist in Form der Atmel Studio Projekte auf dem beiliegenden Datenträger einsehbar (siehe Anhang A).

6.2.4 Datenübermittlung

Die Übermittlung der Messdaten an den Rechner erfolgt via Bluetooth. Dazu wird ein *Sparkfun Bluetooth Mate Gold* verwendet⁵. Diese Platine besitzt eine UART Schnittstelle und kann Datenströme an dieser Schnittstelle via Bluetooth senden und empfangen. Im Fall dieses Projekts sendet das Modul die Messergebnisse an den Rechner, empfängt aber keine eingehenden Daten (z. B. Steuerbefehle). Die Platine wird mit fertiger Firmware ausgeliefert und ist direkt einsatzbereit.

Die Daten von den beiden Mikrocontrollern müssen für eine Übertragung zuerst an das Bluetooth-Modul übermittelt werden. Die Razor Platine dient dabei als eine Art Relais für die Daten des Localinos. Das Localino ist via UART mit dem Razor verbunden. Dieses leitet die Daten des Localinos und die eigenen Messdaten an das Bluetooth-Modul weiter – das geschieht ebenfalls via UART. Die genaue Verkabelung der einzelnen Platinen kann anhand der Abbildung 6.4 nachvollzogen werden.

Softwareseitig wurde auf dem Razor – zusätzlich zum standardmäßig verfügbaren Hardware Serial Port, der für die Verbindung zum Bluetooth Modul genutzt wird – ein weiterer Serial Port an den Pins 10 und 12 implementiert, der die Daten des Localinos empfängt.

Die Spannungsversorgung erfolgt zentral über die Razor Platine. Dort ist ein Anschluss für einen LiPo Akku verbaut. Dieser Anschluss hat zudem

⁴Fehlerbeschreibung und Lösungsvorschläge im GitHub Repository der DW1000 Arduino Library: <https://github.com/thotro/arduino-dw1000/issues/179> (Zugriff 16. November 2017).

⁵Produktseite des Herstellers Sparkfun: <https://www.sparkfun.com/products/12580>

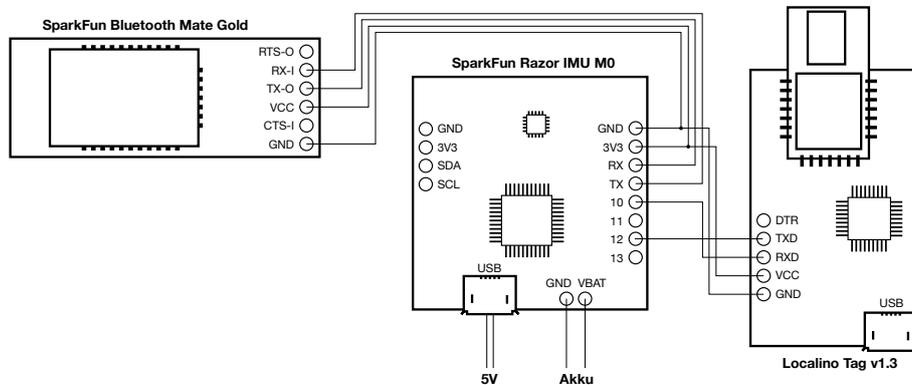


Abbildung 6.4: Signalplan der Sensoren am Kopfhörer.

einen Laderegler integriert. So kann der Akku auch über die Platine geladen werden.

Übertragungsprotokoll

Sowohl die serielle Datenübertragung, als auch die Übertragung per Funk (Bluetooth) haben spezielle Anforderungen an die Kommunikation. Dafür wurde ein Kommunikationsprotokoll konzipiert und implementiert. Folgende Spezifikationen soll das Protokoll erfüllen:

- Automatische Synchronisation
- Prüfung der Datenintegrität
- Möglichkeit der Zuordnung der verschiedenen Datenarten

Das entwickelte Protokoll teilt dazu den Datenstrom in einzelne Pakete auf. Beispielhaft sind zwei Datensätze in Abbildung 6.5 dargestellt. Die Pakete unterscheiden sich für die zwei unterschiedlichen Datentypen Entfernung (Abb. 6.5a) und Euler Winkel (Abb. 6.5b).

Die ersten Bytes jedes Pakets sind Startbytes. Diese dienen zum einen der automatischen Synchronisation. Der Empfänger erkennt den Beginn eines Datenpakets durch den Empfang einer kompletten Startbytefolge. Zum anderen kann das Paket durch die Startbytes auch direkt einer Datenart zugeordnet werden, weil unterschiedliche Bytefolgen verwendet werden.

An die Startbytes wird der Datenblock angehängt. Bei den Datensätzen

0x55	0x55	0x55	0x01	0x3F	0x9D	0x70	0xA4	0x77
Start			Anchor ID	Entfernung (1.23)			Check Bitweises XOR der Datenbytes	
			Daten					

a) Entfernungsdaten aus dem Localino RTLS

0x56	0x56	0x56	0x42	0xB4	0x00	0x00	0xC2	0xB4	0x00	0x00	0x42	0x34	0x00	0x00	0xF6
Sync			Yaw (90.0)			Pitch (-90.0)			Roll (45.0)			Check Bitweises XOR der Datenbytes			
			Daten												

b) Euler Winkel aus der Razor IMU

Abbildung 6.5: Beispieldatenpakete des Übertragungsprotokolls.

aus dem Localino RTLS ist das ein Byte mit der Anchor ID und vier Bytes in denen die Entfernung kodiert ist. Die Darstellung von Fließkommazahlen erfolgt entsprechend der IEEE-754 mit vier Bytes. Die Razor IMU sendet hingegen dreimal je vier Bytes in welche die Eulerwinkel kodiert sind.

Das letzte Byte in jedem Datenpaket dient der Prüfung der empfangenen Daten. Der Sender berechnet hierzu ein bitweises XOR aller Datenbytes und schreibt das Ergebnis in dieses letzte Byte. Anhand dieses Bytes überprüft der Empfänger dann die Korrektheit der empfangenen Daten, indem er diese Rechenoperation ebenfalls durchführt und sein Ergebnis mit dem Wert im letzten Byte vergleicht. Korrupte Datenpakete werden verworfen.

Auf Endbytes kann verzichtet werden, da die zwei unterschiedlichen Datenarten verschiedene Startbytefolgen nutzen. Somit ist eine Zuordnung möglich und folglich auch die Länge des Datenpakets bekannt.

Dieses Kommunikationsprotokoll wird sowohl für die Übertragung zwischen den Mikrocontrollern der Sensoreinheit, als auch für die Übertragung zum Rechner verwendet. Die Implementierung geschieht in Form einer Arduino Library, die in den einzelnen Quellcodes für die verschiedenen Mikrocontrollern integriert ist. Diese sind auf dem beiliegenden Datenträger einsehbar (siehe Anhang A).

6.3 Datenverarbeitung

Den zweiten großen Teil des Projekts stellt die Datenverarbeitung im Rechner dar. Hier werden vor allem Wertebereiche angepasst, Messergebnisse gefiltert und geglättet und die Trilateration der Entfernungsdaten durchgeführt.

Grob teilt sich die Datenverarbeitung bzw. die dafür nötige Programmierung in drei Bereiche:

- *Head-Tracking*: Nullposition zurücksetzen
- *Indoor-Lokalisierung*: Anpassung Wertebereich, Glättung/Filterung, Trilateration
- *Rendering*: Visuelle Darstellung der Messwerte

Am aufwändigsten gestaltet sich dabei die Verarbeitung der Indoor-Lokalisierung, da hier zum einen viele Anpassungen der Messwerte nötig sind und zum anderen die Trilateration der Entfernungen durchgeführt werden muss. Das Head-Tracking hingegen benötigt wenig Anpassung, hier wird nur ein Zurücksetzen der Nullposition implementiert. Der Bereich Rendering umfasst eine visuelle Darstellung der Orientierung und der Position. Das ist relativ trivial und wird daher im Folgenden nur in Randbereichen betrachtet.

6.3.1 Visuelle Programmierumgebung

Die Datenverarbeitung wurde mit der visuellen Programmierumgebung Max (Version 7) realisiert⁶. Sie wird vor allem für die Erstellung von interaktiven Multimediainhalten verwendet. Mit Max können sowohl Audiosignale, Videosignale, als auch Steuer- oder Berechnungsdaten (z. B. Midi) erzeugt, verarbeitet und miteinander verknüpft werden. In einem sogenannten Patcher (Programmfenster) verbindet man dazu einzelne Objekte – das können Quellen, Knoten und Senken sein – mit sogenannten Patchcords – Linien die den Signalfluss repräsentieren. Bindet man Patcher wiederum in andere Patcher ein, nennt man diese Subpatcher. Das Ganze geschieht „live“, es wird also nicht zwischen Programmierung und Ausführung unterschieden. Sobald man einen Patcher öffnet, beginnt dessen Laufzeit.

Erweitern kann man Max-Patcher mit Externalen. Das sind weitere Objekte,

⁶Website des Herstellers Cycling74: <https://cycling74.com/max7>. Bis Version 5 wurde die Software unter dem Namen Max/MSP vertrieben.

die zusätzliche zum normalen Funktionsumfang eingefügt werden können. Es gibt eine große Community um Max in der solche Externalen entwickelt und verfügbar gemacht werden⁷. Neben der visuellen Programmierung bietet Max auch die Möglichkeit JavaScript Code einzubinden. Diese Möglichkeit wird in diesem Projekt genutzt um komplexe Berechnungen einzubinden, für die die grafische Darstellung an ihre Grenzen kommt.

Für eine vereinfachte Bedienung gibt es den Presentation-Mode. Dazu werden einzelne Objekte einer separaten Ansicht hinzugefügt. Hier können sie unabhängig von der Darstellung im Edit-Mode platziert werden. So können einfache Benutzeroberflächen generiert werden.

6.3.2 Head-Tracking

Die Messdaten werden – wie bereits im vorausgegangenen Abschnitt erwähnt – per Bluetooth mit einem speziellen Kommunikationsprotokoll an den Rechner gesendet. Dazu muss dieses Protokoll auch im Max-Patch integriert sein, damit die Daten richtig empfangen werden können. Das umfasst sowohl das Erkennen, Einlesen und Überprüfen von Paketen, als auch die Rückwandlung der einzelnen Bytes in Fließkommazahlen.

Die so empfangenen Messwerte des Head-Trackings (Euler Winkel) können direkt verwendet werden. Da die Umrechnung vom Bogenmaß ins Gradmaß bereits auf dem Mikrocontroller stattfindet, entfällt eine Umrechnung des Wertebereichs. Dieser umfasst $\alpha \in]-180^\circ, 180^\circ]$.

Da die IMU am Kopfhörer ihre Nullposition im Moment des Einschaltens festlegt, wurde eine Funktion implementiert die die Orientierung zurücksetzt. Der tatsächliche Wert zum Zeitpunkt des Zurücksetzens wird hierbei zwischengespeichert und dieser anschließend von allen Messwerte abgezogen. Entstehen dabei überstumpfe Winkel ($180^\circ < \alpha < 360^\circ$ bzw. $-180^\circ > \alpha > -360^\circ$) wird entsprechend der negative ergänzende Winkel ($-(360^\circ - \alpha)$ bzw. $-(-360^\circ - \alpha)$) verwendet. Dies gewährleistet, dass sich anschließend alle Werte wieder im ursprünglichen Wertebereich befinden.

⁷Zum Beispiel im Package Manager von Cycling74 (<https://cycling74.com/packages>) oder in der Max Object Database (<http://www.maxobjects.com>).

6.3.3 Indoor-Lokalisierung

Bei der Indoor-Lokalisierung finden im Gegensatz zum Head-Tracking viele Berechnungen im Rechner statt. Vor allem für die Trilateration sind neben den Messwerten noch zusätzliche Angaben, wie etwa die Position der Basisstationen, notwendig. Grundsätzlich wäre eine Umsetzung direkt auf dem Mikrocontroller zwar denkbar, diese würde jedoch erheblichen Mehraufwand mit sich bringen. Beispielsweise müssten nicht nur Messwerte vom Mikrocontroller zum Rechner, sondern auch Steuerdaten vom Rechner zurück zum Mikrocontroller gesendet werden. Ein Vorteil von der Auslagerung in den Rechner ist also der direkte Zugriff auf Parameter. Zusätzlich besteht auf dem Rechner die Möglichkeit die Messungen zu visualisieren, was sich bei der Konfiguration des Systems als sehr hilfreich erweist..

6.3.3.1 Glättung

Die Messwertewerte vom Mikrocontroller unterliegen kurzfristigen Schwankungen (Rauschen), die vor allem später bei der Trilateration relativ große Auswirkungen haben können. Ein Verfahren zur Minimierung des Rauschens ist Glättung. An die Glättung einer Messung werden dabei im Allgemeinen zwei gegensätzliche Anforderungen gestellt:

- *Stabilisierung*: Wenn sich die tatsächlichen Werte nicht ändert, sollten sich auch die gemessenen Werte möglichst nicht bzw. nur minimal ändern.
- *Änderungssensibilität*: Ändern sich die tatsächlichen Werte, sollen die Messwerte möglichst direkt folgen.

Diese beiden Anforderungen gilt es zu optimieren.

Einfache Glättungsverfahren wie z. B. Mittelwertbildung können zwar Rauschen gut unterdrücken, folgen bei Änderung aber nur langsam, weil viele zurückliegende Werte berücksichtigt werden. Exponentielle Glättung bildet zwar auch Durchschnittswerte, gewichtet aktuelle Werte aber höher als weiter Zurückliegende. Der geglättete Wert x_t^* errechnet sich als mit a gewichteter Durchschnitt aus dem aktuellen Wert x_t und dem geglätteten Wert des vorherigen Durchgangs x_{t-1}^* (vgl. Schröder, 2012, S. 24f):

$$x_t^* = a \cdot x_t + (1 - a) \cdot x_{t-1}^* \quad (6.2)$$

Der Gewichtungsfaktor a liegt dabei zwischen 0 und 1. Für große Werte wird wenig geglättet (im Extremfall $a = 1$ erfolgt keine Glättung), somit folgen die Werte schnell. Bei kleineren Werten, die eine starke Glättung zur Folge haben, braucht das Filter länger um Änderungen zu folgen.

Eine Implementierung eines einfachen Exponentialfilters in JavaScript könnte z. B. so aussehen:

```
output += (input - output) * a;
```

Durch dynamische Anpassung der Gewichtung, kann das Ergebnis weiter verbessert werden. Ändern sich die tatsächlichen Werte nur langsam, kann stark geglättet werden, weil in diesem Fall keine große Änderungssensibilität nötig ist. Ändern sich die Werte sehr stark, ist hingegen eine hohe Änderungssensibilität nötig, die Glättung wird dabei unwichtiger. Die dynamische Änderung kann z. B. durch eine invertierte Hyperbelfunktion ausgedrückt werden:

$$a = 1 - \frac{1}{\Delta x + 1}, \quad \Delta x = x_t - x_{t-1}^* \quad (6.3)$$

Diese dynamische exponentielle Glättung verbessert die Messung nicht unerheblich. Abbildung 6.6 veranschaulicht das für eine Entfernungsmessung, indem sie den unbearbeiteten Messverlauf und geglättete Messverläufe gegenüberstellt. Das Exponentialfilter mit dem Gewichtungsfaktor $a = 0.5$ liefert einen Messverlauf mit guter Stabilität in den statischen Passagen, es braucht aber relativ lange, um Änderungen zu folgen. Im Gegensatz dazu folgt das Filter mit Gewichtungsfaktor $a = 0.25$ zwar der Entfernungsänderung recht gut, schafft aber in den statischen Passagen keine gute Glättung. Das Exponentialfilter mit dynamischem Gewichtungsfaktor besitzt in den statischen Passagen eine gute Stabilität, aber auch eine hohe Änderungssensibilität.

Wie gut die Glättung funktioniert, zeigt auch eine Betrachtung der Varianz in einem statischen Bereich. Für die ungefilterten Messwerte in Sekunde 1–6 ergibt sich eine Varianz $\sigma^2 = 7,6973$. Mit dem dynamischen Exponentialfilter ergibt sich eine Varianz $\sigma^2 = 0,0221$ für Messwerte dieses Bereichs. Die verschiedenen Varianzen und sämtliche Messwerte aus Abbildung 6.6 sind auf dem beilegenden Datenträger einsehbar (siehe Anhang A). Ebenfalls ist dort die Implementierung des dynamischen Exponentialfilters in Form eines in den Max-Patcher eingebundenen JavaScript Quellcodes hinterlegt.

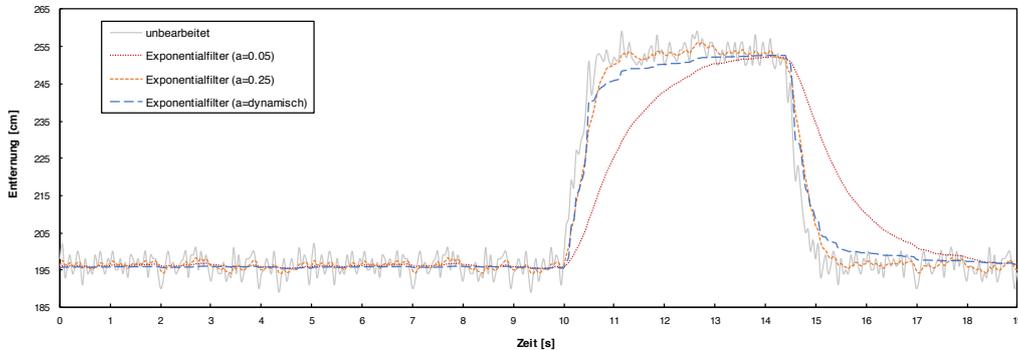


Abbildung 6.6: Plot einer Entfernungsmessung mit unterschiedlicher Einstellung des Glättungsfilters.

6.3.3.2 Kalibrierung

Für den Localino Mikrocontroller ist eine Anpassung des Wertebereichs nötig, da die Messwerte an sich nicht mit der tatsächlichen Entfernung übereinstimmen. Aus diesem Grund wurde eine Messung zur Kalibrierung durchgeführt. Im Bereich von 20 bis 200 Zentimeter wurde alle 10 Zentimeter mit einer Stichprobe von 100 Werten der Abstand gemessen. Die Mittelwerte und die Varianz für die verschiedenen Stichproben sind in Abbildung 6.7

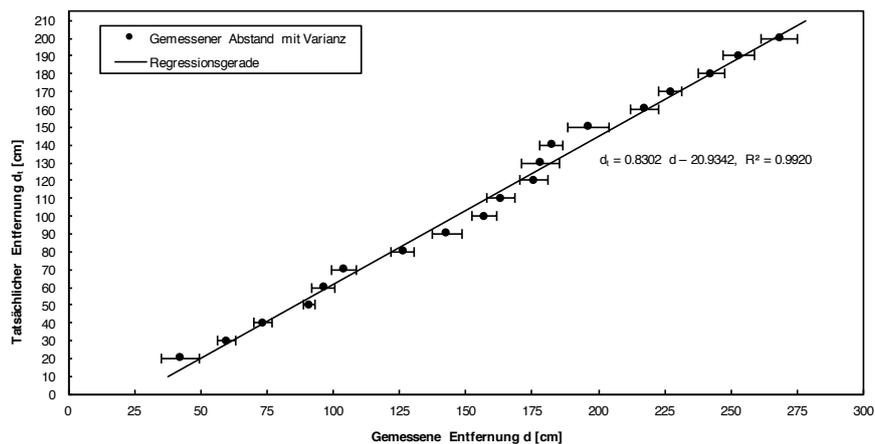


Abbildung 6.7: Entfernungsmessungen zur Bestimmung einer Korrekturfunktion.

aufgetragen. Wie man erkennt, handelt es sich um eine lineare Funktion. Die Regressionsgerade erreicht ein Bestimmtheitsmaß von 0,9920. Für die Ermittlung der Regressionsgerade wird die Excel-Formel für lineare Regression verwendet. Die Berechnung – ebenso wie die detaillierten Messwerte – können auf dem beiliegenden Datenträger eingesehen werden (siehe Anhang A).

Die Messreihe gibt außerdem Aufschluss über die Genauigkeit des UWB-Systems. Berechnet man anhand der Regressionsgleichung die idealen Messwerte, kann man die Abweichung der Messung mit der idealen Messung bestimmen und anschließend die reale Abweichung berechnen. Diese beträgt im Mittel 3,89 Zentimeter. Die größte Abweichung bei den Stichproben beträgt jedoch 9,64 Zentimeter. Die Genauigkeit liegt also etwa im Dezimeterbereich und deckt sich damit mit den meisten Angaben in der Literatur (siehe Tabelle 5.1).

6.3.3.3 Trilateration

Im nächsten Schritt kann nun aus den Entfernungen vom Tag zu den drei Basisstationen mittels Trilateration die Position des Tags bestimmt werden. Dafür wird der folgende Algorithmus von Konrad Dzwintel⁸ verwendet. Für die Objekte A1, A2, A3 = {x:0, y:0} müssen zuvor entsprechend die Koordinaten der Basisstationen gesetzt sein:

```

1 function trilaterate(r1, r2, r3) {
2
3   var S = (Math.pow(A3.x, 2.) - Math.pow(A2.x, 2.) + Math.pow(A3.y, 2.)
4     - Math.pow(A2.y, 2.) + Math.pow(r2, 2.) - Math.pow(r3, 2.)) / 2.0;
5   var T = (Math.pow(A1.x, 2.) - Math.pow(A2.x, 2.) + Math.pow(A1.y, 2.)
6     - Math.pow(A2.y, 2.) + Math.pow(r2, 2.) - Math.pow(r1, 2.)) / 2.0;
7   var y = ((T * (A2.x - A3.x)) - (S * (A2.x - A1.x))) / (((A1.y - A2.y)
8     * (A2.x - A3.x)) - ((A3.y - A2.y) * (A2.x - A1.x)));
9   var x = ((y * (A1.y - A2.y)) - T) / (A2.x - A1.x);
10
11  return {x:x, y:y};
12 }
```

Auf eine Problematik des Algorithmus sei an dieser Stelle hingewiesen. Wenn die x Koordinaten von Basisstation 1 und 2 zusammenfallen, kommt

⁸Veröffentlicht auf GitHub Gist: <https://gist.github.com/kdzwintel/8235348> (Zugriff 18. November 2017).

es in Zeile 6 zu einer Division durch Null. Dies kann jedoch leicht über die Positionierung der Basisstationen gelöst werden. Bereits ein Versatz von 1 Zentimeter würde ausreichen. Diese Abweichung liegt außerhalb des Genauigkeitsbereichs des Messsystems und ist somit nicht relevant.

Ansonsten funktioniert der Algorithmus sehr zuverlässig, auch wenn durch Ungenauigkeiten in der Messung keine exakte Lösung für die Trilateration gefunden werden kann. Abbildung 6.8 veranschaulicht geometrisch, wie der Algorithmus diese Ungenauigkeiten ausgleicht. Im Grunde skaliert er die Messwerte so, dass sie sich in einem Punkt schneiden. Dabei behält er die Verhältnisse der Einzelmessungen bei.

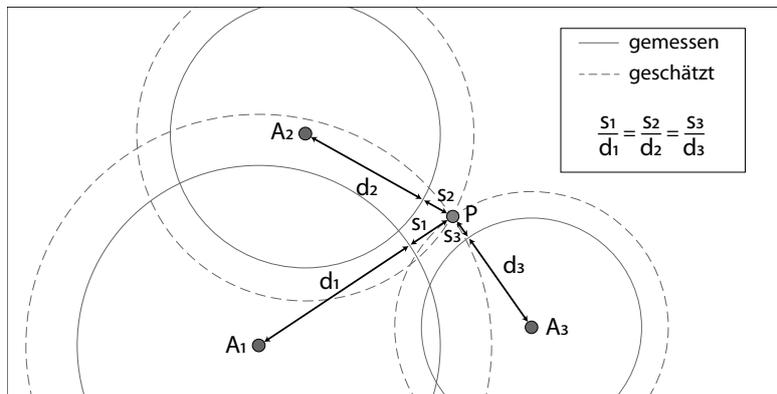


Abbildung 6.8: Geometrische Darstellung des Ungenauigkeitsausgleichs durch den Trilaterationsalgorithmus.

Neben dem nun implementierten Algorithmus, wurde ein Algorithmus von Gabor Heja⁹ getestet. Er erwies sich für diese Zwecke jedoch als weniger leistungsfähig, obwohl er zwar grundlegend genauer rechnet, durch die Messung entstehende Ungenauigkeiten jedoch schlechter ausgleicht.

Positionierung der Basisstationen

Für gute Messergebnisse ist auch die Positionierung der Basisstationen ausschlaggebend. Hierbei ist zu beachten, dass sich die Anchor auf einer Ebene mit dem Tag befinden. Im Fall dieses Projekts also etwa auf Kopfhöhe, da

⁹Veröffentlicht auf GitHub: <https://github.com/gheja/trilateration.js> (Zugriff 18. November 2017).

auch die Trilateration in dieser Ebene stattfindet. Auch sollten die Basisstationen ein möglichst gleichmäßiges Feld aufspannen, am besten mit annähernd gleichen Abständen zwischen den Basisstationen. Die Fläche zwischen den Basisstationen ist somit am größten.

Die Positionen der Basisstationen können im Max-Patcher grundsätzlich frei angegeben werden, um den Prozess der Positionierung der Basisstationen zu vereinfachen wurde jedoch eine Kalkulation über die Abstände der Anchor implementiert. In Abbildung 6.9 sieht man, welche Anordnung die Anchor dabei im Raum haben sollten.

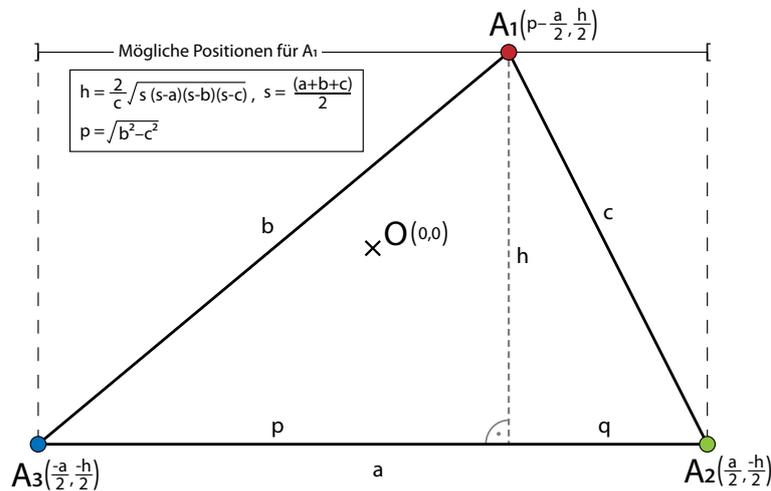


Abbildung 6.9: Berechnung der Koordinaten der Basisstationen über deren Abstände zueinander.

Für die Berechnung sind die Seiten a , b und c (Abstände zwischen den Basisstationen) gegeben. Der Ursprung wird in der Mitte des von der Grundlinie a und der Höhe h aufgespannten Rechtecks definiert. Zur Berechnung von Höhe h und der Seite p wird die Formel für den Flächeninhalt eines Dreiecks nach Heron verwendet. Die Seite p wird über den Satz des Pythagoras im rechtwinkligen Dreieck berechnet.

6.3.4 Benutzeroberfläche

Damit die Bedienung der Software erleichtert wird, wurde eine Benutzeroberfläche umgesetzt. Dazu wurden die wichtigsten Parameter und Bedienelemente dem Presentation Mode hinzugefügt.

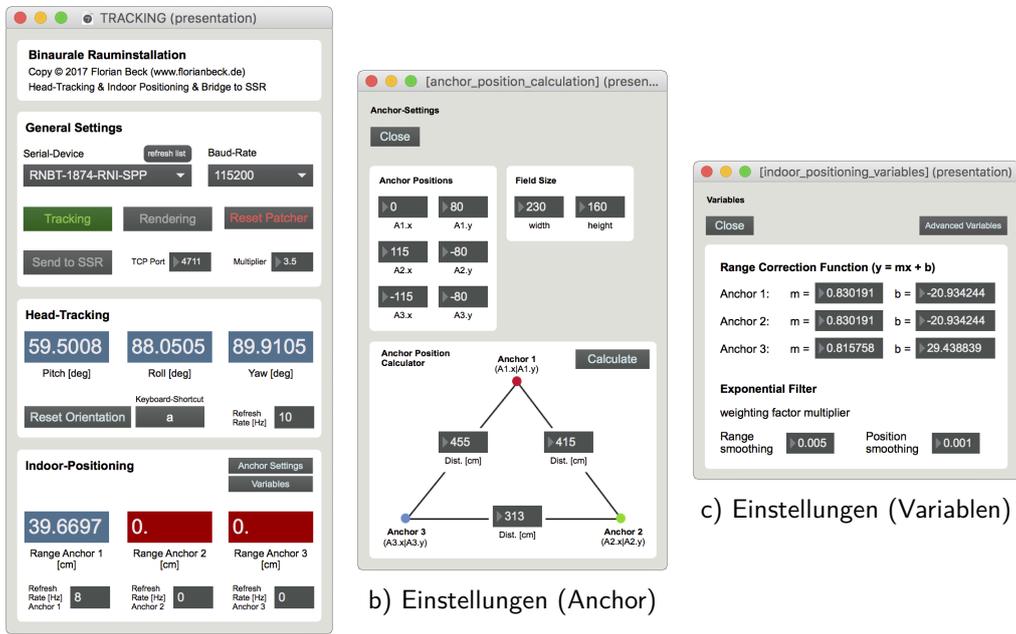
Das Hauptfenster (siehe Abbildung 6.10a) beinhaltet drei Blöcke. Der erste Block beinhaltet die Grundeinstellungen. Hier kann die serielle Schnittstelle konfiguriert werden (*Serial-Device* und *Baud-Rate*). *Tracking* öffnet diese Verbindung und startet das Einlesen und Verarbeiten der Messwerte. *Rendering* öffnet ein Fenster mit einer visuellen Darstellung der aktuellen Messwerte. *Send to SSR* aktiviert den Versand der Messwerte via TCP an die Software SoundScape Renderer. Dabei kann zusätzlich der Port festgelegt und ein Faktor angegeben werden, mit dem die Messwerte multipliziert und somit im Wertebereich an den SoundScape Renderer angepasst werden.

Der zweite Block zeigt die Messwerte des Head-Tracking für die drei Euler Winkel an. *Reset Orientation* setzt die Nullposition zurück, dazu kann auch ein Tastaturkürzel konfiguriert werden. Des weiteren wird die Wiederholrate der Messung angezeigt.

Die Messwerte zur Indoor-Lokalisierung sind im dritten Block dargestellt. Für jeden Anchor wird der aktuelle Entfernungsmesswert und die Wiederholrate angezeigt. Die Messwertfelder sind im Normalfall Blau dargestellt. Werden gerade keine Messwerte empfangen, wechseln die Felder in die Farbe Rot. Das ist z. B. in Abbildung 6.10a für Anchor 2 und 3 der Fall. Über die Buttons *Anchor Settings* und *Variables* können Fenster mit weiteren Einstellungen geöffnet werden.

Das in Abbildung 6.10b dargestellte Fenster ermöglicht die Konfiguration der Basisstationen für die Indoor-Lokalisierung. Die Koordinaten können entweder direkt eingegeben oder im unteren Teil anhand der Abstände zwischen den Basisstationen berechnet werden. Abbildung 6.10c zeigt ein weiteres Einstellungsfenster. Hier können die Korrekturfunktionen für die Abstandsmessung und die Gewichtungsfaktoren für die Glättung angegeben werden.

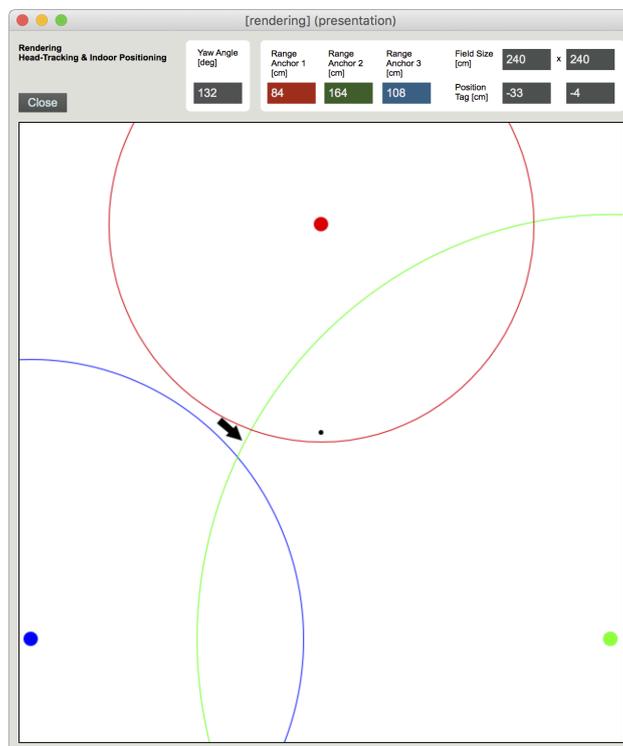
Eine visuelle Darstellung von Head-Trackings und Indoor-Lokalisierung ist im *Rendering* Fenster in Abbildung 6.10d implementiert. Kreise um die Anchor repräsentieren die Abstandsmessungen. Die Position und die Orientierung des Kopfhörers wird durch einen schwarzen Pfeil dargestellt. Die numerischen Messwerte der Absände und Winkel werden wiederholt. Die Position wird



a) Hauptfenster

b) Einstellungen (Anchor)

c) Einstellungen (Variablen)



d) Rendering

Abbildung 6.10: Benutzeroberfläche der Datenverarbeitungsanwendung.

numerisch in Zentimetern ausgegeben. Unter *Field Size* wird zusätzlich die Größe des Rendering-Rechteck angezeigt.

Die so verarbeiteten und visualisierten Daten werden per TCP innerhalb des Rechners an das Audio-Rendering weitergegeben.

6.4 Audio-Rendering

Das Audio-Rendering stellt den dritten großen Projektteil dar. Dazu wird das freie Tool *SoundScape Renderer* (SSR) verwendet¹⁰. Es ist Gemeinschaftsentwicklung des Quality and Usability Lab der Technischen Universität Berlin und des Instituts für Nachrichtentechnik der Universität Rostock. Das Tool ist in C++ geschrieben und die Benutzeroberfläche in Qt4. Es läuft unter Linux und Mac OS X. Es sind sowohl Quellcodes verfügbar, die selbst kompiliert werden können, als auch ein Debian Package (für Linux) und ein Application Bundle (für Mac OS X). Da der SSR das *JACK Audio Connection Kit* für das Audio Routing nutzt, ist dieses Tool ebenfalls nötig¹¹.

Der SSR zeichnet sich vor allem durch zwei Eigenschaften aus. Zum einen unterstützt er mehrere Renderer für verschiedene Wiedergabeverfahren. Neben der dynamischen binauralen Synthese – sie wird in diesem Projekt verwendet – bietet das Tool die Möglichkeit, Lautsprecher-synthesesignale für Wellenfeldsynthese, Higher Order Ambisonics und Vector Base Amplitude Panning zu generieren. Dadurch können z. B. leicht verschiedene Verfahren verglichen werden. Zum anderen benutzt ein spezielles, offenes, objektbasiertes Format aus Audio- und Metadaten für die Beschreibung von Audioszenen.

Binauraler Renderer

Für die Binauralsynthese nutzt der SoundScape Renderer eine Faltung mit einem Set von kopfbezogenen Übertragungsfunktionen (HRIR). Dieses besitzt eine Auflösung von 1°, also 360 HRIRs für das linke Ohr und 360 HRIRs für das rechte Ohr. Die einzelnen Impulsantworten werden in einem multichannel WAV File verpackt. Dabei enthält der erste Audiokanal die Impulsantwort des linken Ohres für eine 0° Quelle, der zweite Audiokanal die des rechten

¹⁰Projektwebsite: <http://spatialaudio.net/ssr/>

¹¹Projektwebsite: <http://jackaudio.org>

Ohres für eine 0° Quelle. Kanal 3 enthält 1° links, Kanal 4 enthält 1° rechts und so weiter.

Mit dem SSR werden zwei unterschiedliche HRIR Sets mitgeliefert. FABIAN (Lindauer & Weinzierl, 2007) und KEMAR (Wierstorf, Geier, Raake & Spors, 2011). Für beider Sets gibt es jeweils eine Version in die zusätzlich eine Kopfhörerkompensation eingerechnet ist.

Audio Scene Description Format

Mit dem SoundScape Renderer ist eine objektbasierte Produktion möglich. Dazu müssen nicht nur reine Audiofiles, sondern auch Metadaten verarbeitet werden. Der SSR nutzt dazu das auf XML basierende Audio Scene Description Format (ASDF) nach Geier und Spors (2008). In diesem Format kann einzelnen Mono-Soundfiles bzw. einzelnen Audiokanälen eines mehrkanaligen Soundfiles eine Position in der Audioszene zugeordnet werden. Neben der Wiedergabelautstärke, kann zudem angegeben werden, ob die Soundquelle als Punktschallquelle oder als ebene Wellenfront gerendert werden soll. Das ASDF ist leider nicht besonders umfangreich dokumentiert, jedoch ist das Format auch relativ simpel gestaltet und die Lernkurve daher entsprechend steil.

Benutzeroberfläche

Die Benutzeroberfläche des SoundScape Renderers ist relativ übersichtlich. Beim Starten muss man direkt den Renderer wählen; für dieses Projekt *Binaural (using HRIRs)*. Anschließend öffnet sich das Rendering- und das Controls-Fenster. Lädt man nun über das *File* Dropdown-Menü eine Audioszene, erscheinen im Rendering Fenster die einzelnen Soundquellen. Mit den üblichen Controls (Play, Pause, Stopp) lässt sich die Audioszene abspielen. Das ist auch der Programmzustand, der in Abbildung 6.11 zu sehen ist. Über die Benutzeroberfläche kann man die Soundquellen nun verschieben und die wichtigsten Einstellungen vornehmen. In der Abbildung geschieht das für die Soundquelle mit dem Namen *Möbius*.

Ansonsten sind die Funktionen sehr beschränkt. Audioquellen können z. B. nicht über die Benutzeroberfläche hinzugefügt werden, dafür muss man die ASDF Datei anpassen. Über die TCP Schnittstelle lassen sich alle eben

vorgestellten Parameter fernsteuern. Für dieses Projekt geschieht das mit der Position und der Orientierung des Hörers (*Reference Position* und *Reference Orientation*).

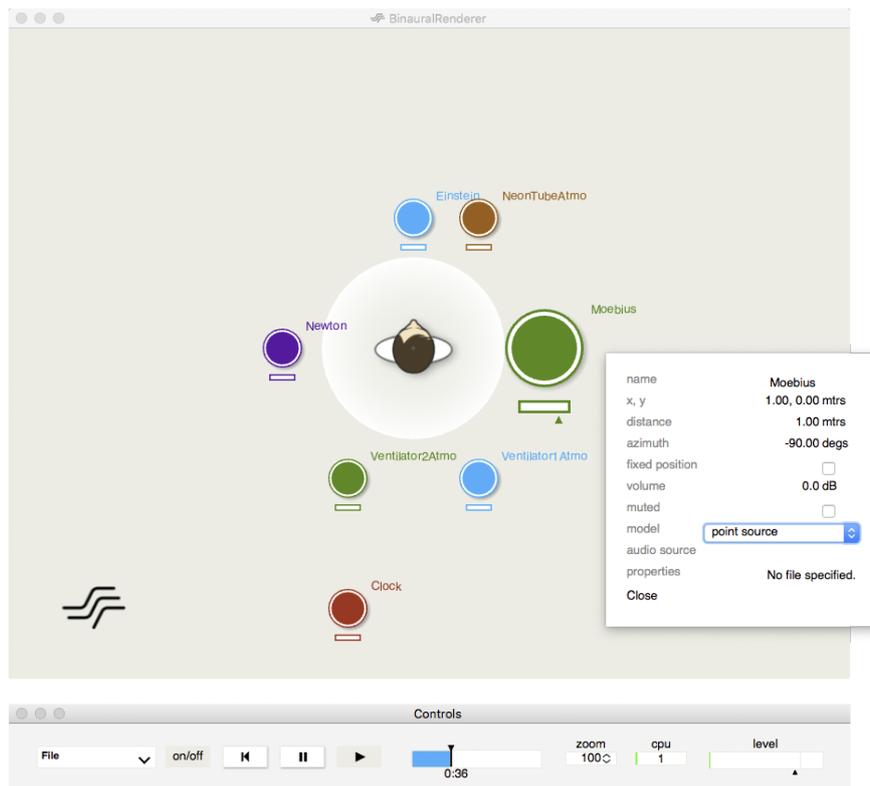


Abbildung 6.11: Benutzeroberfläche des SoundScape Renderers.

Audioszenen

Zum Testen des Systems wurden verschiedene Audioszenen verwendet, die hauptsächlich auf bestehenden Vorlagen der Fachhochschule Köln und der Hochschule für Musik und Theater Rostock basieren.

In einer Audioszene begibt man sich in einen Ausschnitt aus der Komödie *Die Physiker* von Friedrich Dürrenmatt. Man befindet sich direkt im Gespräch der drei Physiker Newton, Einstein und Möbius. Die Aufnahmen stammen von Stefan Greuel und Philipp Stade (FH Köln, 2011). Zusätzlich zu den

Sprachaufnahmen sind Atmos in Form von Ventilatorengeräuschen und den Sounds einer Neonröhre vorhanden.

Eine weitere Audioszene beinhaltet ein Cover des Song *How will I know* von Whitney Houston – hier mit zwei Gitarren, Bass, Vocals und Backing Vocals. Die Aufnahmen stammen ebenfalls von Stefan Greuel und Philipp Stade (FH Köln).

Die dritte Audioszene ist das Musikstück *Las Vegas Tango* von Gil Evans. Die Besetzung ist Schlagzeug, Bass, Gitarre, Piano, Saxophon und Gesang. Die Aufnahmen stammen von der HMT Rostock und wurden dort im Tonstudioseminar von Studenten unter Leitung von Carsten Storm aufgezeichnet.

Audiowiedergabe

Die Wiedergabe der binauralen Audiosignale benötigt Kopfhörer. Für das Projekt werden dazu Bluetooth Kopfhörer verwendet, da eine kabelgebundene Lösung für die Bewegung im Raum hinderlich wäre. Die Audiosignale werden über JACK direkt aus dem SoundScape Renderer an die Bluetooth Kopfhörer geroutet.

Kapitel 7

Schlussbetrachtungen

7.1 Ergebnisse

In dieser Arbeit ist ein funktionsfähiges System entstanden. Mit den gewählten Technologien kann durchaus eine technische Umgebung für immersive Rauminstallationen entwickelt werden. In einigen Bereichen bleibt jedoch Potential für Verbesserungen.

Das Head-Tracking mittels Inertialsensoren funktioniert sehr gut. Die Genauigkeit ist dabei sehr hoch und die Messwerte sind sehr stabil. Die Umsetzung gestaltet sich relativ einfach. Bei der Indoor-Lokalisierung ist hingegen mehr Aufwand für die Umsetzung nötig. Trotz Kalibrierung, Korrektur und vielen Anpassungen, ist nur eine Genauigkeit im Dezimeterbereich möglich. Eine absolute Positionierung gestaltet sich relativ komplex. Für diese Anwendung sind jedoch relative Positionsangaben ausreichend. Wie die verschiedenen Messungen zeigen, bewirkt die Datenverarbeitung der Messwerte eine sehr große Verbesserung.

Subjektiv ist bei schnelleren Bewegungen durchaus eine Verzögerung wahrzunehmen. Leider war es nicht möglich die einzelnen Verzögerungen in der Signalkette des Systems durch Messung zu bestimmen. Eine objektive Aussage über die Gesamtverzögerung ist somit nicht möglich.

Die Einhüllung des Zuhörers kann vom Autor nur bedingt bestätigt werden. Teilweise kommt es zu einer In-Kopf-Lokalisierung, die dem sehr entgegenwirkt. Vor allem das Head-Tracking bringt hier jedoch eine große Verbesserung mit sich. Damit können mit einer gewissen „Einhörzeit“ durchaus umhüllende

Klangfelder wahrgenommen werden. Die einhüllende Wirkung von binauralen Tonaufnahmen ist jedoch – wie im Abschnitt über Binauraltechnik beschrieben – sehr individuell und stark vom verwendeten HRIR Set abhängig. Für den Autor konnten mit dem KEMAR Set die besten Ergebnisse erzielt werden.

Damit die vorgestellten Ergebnisse besser nachvollzogen werden kann, sind auf dem beiliegenden Datenträger Filmaufnahmen und Fotos hinterlegt, die einzelne Details und das komplette System in Verwendung zeigen.

7.2 Diskussion und Ausblick

Der momentane Entwicklungsschritt lässt einigen Raum für Verbesserungen, erreicht in vielen Bereichen aber auch durchaus akzeptable Ergebnisse. Das meiste Verbesserungspotential birgt wohl eine Individualisierung des eingesetzten HRIR Sets. Vermutlich könnte damit vor allem die In-Kopf-Lokalisierung minimiert und die Immersion deutlich gesteigert werden. Ebenso verhält es sich mit einer Kompensation der Impulsantworten für den verwendeten Kopfhörer.

Das entwickelte System verwendet momentan einen separaten Rechner. Einerseits wäre eine Adaption z. B. auf ein Raspberry Pi denkbar. Alle Teile der Signalkette könnten dann direkt am Kopfhörer abgebildet werden. Somit wäre es denkbar, auch mehrere solcher Kopfhörer in einer Rauminstallation zu verwenden. Andererseits wäre aber auch eine Umsetzung auf dem Smartphone interessant. Diese Geräte haben in der Regel bereits Inertialsensoren an Bord und die Indoor-Lokalisierung wäre z. B. über Bluetooth denkbar. Dadurch könnte eine Rauminstallation ohne spezielle Hardware auskommen und so z. B. auch in öffentlichem Raum lokalisiert sein, wo ansonsten der Einsatz von Technik aufgrund von Vandalismus oft ausgeschlossen ist.

Betrachtet man den Bereich der Immersion detaillierter, wäre es interessant weiterführend zu untersuchen, welche Anteile an der Umhüllung die verschiedenen Technologien Head-Tracking und Indoor-Lokalisierung haben, und ob diese auch einzeln oder nur in Kombination funktionieren. Benutzt man nur Head-Tracking könnte man z. B. auch binaurale Raumsimulation mittels BRIRs verwenden. Diese gelten allgemein als näher am natürlichen Hören, besitzen jedoch nicht die Flexibilität, die eine dynamische Synthese mittels HRIRs mitbringt.

Zuletzt ist eine Umsetzung weiterer Klanglandschaften mit dem System anzustreben. Die in dieser Arbeit verwendeten Audioszenen lassen zwar eine Beurteilung des Systems zu, zeigen jedoch nicht die kreativen Möglichkeiten auf, die ein solches System ermöglicht. Zuletzt ist auch eine Evaluation durch weitere Zuhörer wünschenswert, um letztlich zu klären, ob ein solches System auch in einer konkreten Ausstellung eingesetzt werden kann.

7.3 Fazit

Zusammenfassend ist mit dieser Arbeit ein erster Entwicklungsschritt für ein neuartiges System geschaffen worden. Sicher ist noch Potential für Verbesserungen, aber es wird aufgezeigt, dass so ein System grundsätzlich umsetzbar ist. Die Auseinandersetzung mit den verwendeten Technologien ist und bleibt spannend. Gerade der Bereich Indoor-Lokalisierung ist momentan noch mit erheblichem technischen Aufwand verbunden. Technische Fortschritte in diesem Bereich sind jedoch absehbar. Somit bleibt zu hoffen, dass sich die Möglichkeiten in der Indoor-Lokalisierung in naher Zukunft deutlich erweitern und die Umsetzung solcher Systeme somit vereinfacht wird. Wie aufgezeigt bleibt Potential für eine weitere Auseinandersetzung mit der Thematik. Vor allem die tatsächliche Anwendung der entwickelten technischen Umgebung wäre ein erstrebenswertes Ziel.

Literaturverzeichnis

- Arcoverde, E., Duarte, R., Barreto, R., Paulo Magalhaes, J., Bastos, C., Ren, T. & Cavalcanti, G. (2014). Enhanced real-time head pose estimation system for mobile device. *Integrated Computer Aided Engineering*, 21(3), S. 281–293.
- Auro Technologies & Barco. (2012). Auro 11.1 – A new dimension in cinema sound. Zugriff 11. November 2017 unter http://www.barco.com/secureddownloads/cd/MarketingKits/3d-sound/White%20papers/Auro%2011.1_A%20new%20dimension%20in%20cinema%20sound.pdf
- Auro Technologies & Barco. (2015). AUROMAX. Next generation immersive Sound system. Zugriff 11. November 2017 unter https://www.auro-3d.com/wp-content/uploads/documents/AuroMax_White_Paper_24112015.pdf
- Blanckenbach, J., Norrdine, A., Schlemmer, H. & Willert, V. (2007). Indoor-Positionierung auf Basis von Ultra Wide Band. *Allgemeine Vermessungs-Nachrichten*, 2007(114), S. 169–178.
- Blanco, J.-L. (2013). *A tutorial on $SE(3)$ transformation parameterizations and on-manifold optimization* (Technical Report Nr. 012010). Universität Malaga.
- Blauert, J. (1974). *Räumliches Hören*. Monographien der Nachrichtentechnik. Stuttgart: Hirzel.
- Blauert, J. & Braasch, J. (2008). Räumliches Hören. In S. Weinzierl (Hrsg.), *Handbuch der Audiotechnik* (S. 87–121). Berlin, Heidelberg: Springer.
- Brinkschulte, U. & Ungerer, T. (2002). *Mikrocontroller und Mikroprozessoren*. Berlin: Springer.
- Dickreiter, M., Dittel, V., Hoeg, W. & Wöhr, M. (Hrsg.). (2014). *Handbuch der Tonstudioteknik* (8. überarbeitete und erweiterte Auflage). Berlin: De Gruyter Saur.
- Dickreiter, M. & Goeres-Petri, J. (2014). Schallwahrnehmung. In M. Dickreiter, V. Dittel, W. Hoeg & M. Wöhr (Hrsg.), *Handbuch der Tonstudioteknik*

- (8. überarbeitete und erweiterte Auflage, Bd. 1, S. 115–135). Berlin: De Gruyter Saur.
- Dolby Laboratories. (2014). Dolby Atmos Next-Generation Audio for Cinema. Zugriff 11. November 2017 unter <https://www.dolby.com/us/en/technologies/dolby-atmos/dolby-atmos-next-generation-audio-for-cinema-white-paper.pdf>
- Dolby Laboratories. (2016). Dolby Atmos: Breaking the Sound Barrier. Zugriff 21. November 2017 unter <http://investor.dolby.com/releasedetail.cfm?ReleaseID=1003122>
- Eberhard, P. & Fleißner, F. (2017). Euler- und Kardan-Winkel: Merkblatt Maschinendynamik M8.1. Institut für Technische und Numerische Mechanik, Universität Stuttgart. Zugriff 6. Oktober 2017 unter <http://www.itm.uni-stuttgart.de/courses/madyn/Merkblaetter/M08.pdf>
- Eisenacher, M. (2006). *Optimierung von Ultra-Wideband-Signalen (UWB)* (Dissertation, Universität Karlsruhe). Zugriff 27. Oktober 2017 unter <https://publikationen.bibliothek.kit.edu/1000004809>
- Geier, M. & Spors, S. (2008). ASDF – Audio Scene Description Format. In *ICMC – International Computer Music Conference*.
- Geier, M. & Spors, S. (2012). Spatial Audio with the SoundScape Renderer. In *Proceedings of the 27th Tonmeistertagung – VDT International Convention* (S. 646–655). Köln.
- Görne, T. (2017). *Sounddesign*. Klang, Wahrnehmung, Emotion. München: Carl-Hanser-Verl.
- Hannoschöck, E. (2009). Soundscapes und Lärm. Zur kulturellen Wahrnehmung und Deutung von Klängen. *VOKUS. Volkskundlich kulturwissenschaftliche Schriften*, 19(2), S. 37–51.
- Heinzmann, M., Kruse, M., Puente León, F., Schilling, K., Roth, H. & Wolf, W. (2014). Signalverarbeitung bei Multisensoren. In H.-R. Tränkler & L. Reindl (Hrsg.), *Sensortechnik: Handbuch für Praxis und Wissenschaft* (2., völlig neu bearb. Aufl, S. 1205–1322). Berlin: Springer Vieweg.
- Hoffmann, P. (2009). Freie Rotation im Raum: Quaternionen und Matrizen. Zugriff 6. Oktober 2017 unter <https://www.uninformativ.de/bin/SpaceSim-2401fee.pdf>
- Hollweger, F. (2008). An Introduction to Higher Order Ambisonics. Zugriff 5. November 2017 unter <http://flo.mur.at/writings/HOA-intro.pdf>
- Horn, M. & Höflinger, F. (2014). Inertialsensoren. In H.-R. Tränkler & L. Reindl (Hrsg.), *Sensortechnik: Handbuch für Praxis und Wissenschaft* (2., völlig neu bearb. Aufl, S. 541–571). Berlin: Springer Vieweg.
- Lindauer, A. & Weinzierl, S. (2007). FABIAN – schnelle Erfassung binauraler Raumimpulsantworten in mehreren Freiheitsgraden. In *Fortschritte der Akustik (DAGA)* (S. 633–634). Stuttgart.

- Mautz, R. (2012). *Indoor positioning technologies*. ETH Zurich. doi:10.3929/ethz-a-007313554
- Mulder, J. D., Jansen, J. & van Rhijn, A. (2003). An Affordable Optical Head Tracking System for Desktop VR/AR Systems. In *Proceedings of the Workshop on Virtual Environments 2003* (S. 215–223). EGVE '03. Zurich, Switzerland.
- Pulkki, V. (1997). Virtual sound source positioning using vector base amplitude panning. *Journal of the Audio Engineering Society*, 45(6), S. 456–466.
- Pulkki, V. (2001). *Spatial sound generation and perception by amplitude panning techniques* (Techn. Ber. Nr. 62). Laboratory of Acoustics und Audio Signal Processing. University of Technology Helsinki.
- Retscher, G. & Kistenich, M. (2006). Vergleich von Systemen zur Positionsbestimmung und Navigation in Gebäuden. *zfv – Zeitschrift für Geodäsie, Geoinformation und Landmanagement*, 2006(1), S. 25–35.
- Ribo, M. (2001). *State of the Art Report on Optical Tracking* (Technical Report Nr. VRVis 2001-25). TU Wien.
- Romanov, M., Frank, M., Zotter, F. & Nixon, T. (2016). Manipulations improving amplitude panning on small standard loudspeaker arrangements for surround with height. In *Proceedings of the 29th Tonmeistertagung – VDT International Convention* (S. 33–37). Köln.
- Schafer, R. M. (1994 [1977]). *The Soundscape: Our sonic environment and the tuning of the world*. Rochester, Vermont. (Ursprünglich veröffentlicht unter: *The Tuning of the World*, New York 1977)
- Schröder, M. (2012). Einführung in die kurzfristige Zeitreihenprognose und Vergleich der einzelnen Verfahren. In P. Mertens & S. Rässler (Hrsg.), *Prognoserechnung* (7., wesentlich überarb. und erw. Aufl, S. 11–45). Heidelberg: Physica-Verlag.
- Sengpiel, E. (1992). „Richtungsbestimmende Bänder“ und „Kurven gleicher Lautstärke“. Zugriff 31. Oktober 2017 unter <http://www.sengpielaudio.com/RichtungsbestimmendeBaenderKurvenGleiLautst.pdf>
- Shchekotov, M. (2015). Indoor localization methods based on Wi-Fi lateration and signal strength data collection. In *Proceedings of the 17th Conference of Open Innovations Association (FRUCT) 2015* (S. 186–191). IEEE.
- Silzle, A., Sazdov, R., Weitnauer, M., Warusfel, O., Bleisteiner, W., Vanselow, P., ... Herzog, U. (2016). The EU Project ORPHEUS: Object-based Broadcasting – for Next Generation Audio Experiences. In *Proceedings of the 29th Tonmeistertagung – VDT International Convention* (S. 140–149). Köln.

- Slavik, K. M. & Weinzierl, S. (2008). Wiedergabeverfahren. In S. Weinzierl (Hrsg.), *Handbuch der Audiotechnik* (S. 609–686). Berlin, Heidelberg: Springer.
- Theile, G. (2014). Tonaufnahme und Wiedergabe. In M. Dickreiter, V. Dittel, W. Hoeg & M. Wöhr (Hrsg.), *Handbuch der Tonstudiotechnik* (8. überarbeitete und erweiterte Auflage, Bd. 1, S. 217–369). Berlin: De Gruyter Saur.
- Theile, G., Wittek, H. & Reisinger, M. (2002). Wellenfeldsynthese-Verfahren: Ein Weg für neue Möglichkeiten der räumlichen Tongestaltung. In *Bericht zur 22. Tonmeistertagung*. Hannover.
- Weinzierl, S. (Hrsg.). (2008). *Handbuch der Audiotechnik*. Berlin, Heidelberg: Springer.
- Wetzstein, G. (2017). Course Notes: 3-DOF Orientation Tracking with IMUs. Stanford University. Zugriff unter <https://stanford.edu/class/ee267/lectures/lecture10.pdf>
- Wierstorf, H., Geier, M., Raake, A. & Spors, S. (2011). A Free Database of Head-Related Impulse Response Measurements in the Horizontal Plane with Multiple Distances. In *Proceedings of the 130th Convention of the Audio Engineering Society (AES)*. London.
- Winkler, J. (1995). *Klanglandschaften. Untersuchungen zur Konstitution der klanglichen Umwelt in der Wahrnehmungskultur ländlicher Orte in der Schweiz* (Dissertation, Basel). 2. ergänzte Ausgabe, 2006/2009. Zugriff 31. Oktober 2017 unter http://www.iacsa.eu/jw/winkler_klanglandschaften_1995.pdf
- Woodman, O. J. (2007). *An introduction to inertial navigation* (Technical Report Nr. 696). Computer Laboratory. University of Cambridge.
- Zwiers, U. (2009). Technische Mechanik II: Kinematik des starren Körpers. Hochschule Bochum. Zugriff 1. November 2017 unter http://www.hochschule-bochum.de/fileadmin/media/fb_m/Institute/Mechanik/Zwiers/TM2/TM2_02_Kinematik_SK_handout.pdf

Anhang

A Inhalt des beiliegenden Datenträgers

Dieser Arbeit liegt ein Datenträger mit Quellcodes, Projektdateien, Dokumentation und sonstigen Begleitmaterialien bei.

Format: DVD-ROM, Single-Layer, ISO9660-Format

Hinweis: Sind in den nachfolgenden Auflistungen Ordner angegeben, enthalten diese meistens mehrere Dateien – vor allem im Fall von Quellcodes in spezieller Projektstruktur. Dazu sind weitere Hinweise in `readme`-Files in den entsprechenden Ordnern hinterlegt.

Pfad: /

MA_FlorianBeck.pdf Masterarbeit in digitaler Form

Mikrocontroller

Pfad: /Mikrocontroller

IMU/. Arduino-Projekt für den Sparkfun Razor IMU

RTLS_Anchor/. Atmel-Studio-Projekt für die Localino Anchor

RTLS_Tag/. Atmel-Studio-Projekt für den Localino Tag

SerialPackets_Lib/. Arduino-Library des Übertragungsprotokolls

Datenverarbeitung

Pfad: /Datenverarbeitung

Tracking/ Max7-Projekt der Tracking-Software

Audioszenen

Pfad: /Audioszenen

die_physiker/ Audioszene „Die Physiker“

how_will_i_know/ Audioszene „How will I know“

las_vegas_tango/ Audioszene „Las Vegas Tango“

Messungen

Pfad: /Messungen

Glättungsfiler/ Entfernungsmessung mit verschiedenen
Glättungsfiltern

Korrekturfunktion/ Bestimmung der RTLS-Korrekturfunktion

Position/ Messung verschiedener Positionen

Videos

Pfad: /Videos

Demo1.mp4 Demonstration Head-Tracking

Demo2.mp4 Demonstration Indoor-Lokalisierung