

Jugend forscht Arbeit

Akustische Richtungsbestimmung

Robin Heinemann
Jaro Habiger

18. April 2016

Erstellt im Schülerforschungszentrum Nordhessen



Inhaltsverzeichnis

| | |
|--|-----------|
| 1 Akustische Richtungsbestimmung | 1 |
| 2 Bestehende Lösungen | 1 |
| 3 Konzept | 2 |
| 3.1 Modul 1: Eingabe/Aufnahme | 2 |
| 3.2 Modul 2: Signal nach Frequenzen aufteilen | 2 |
| 3.3 Modul 3: Richtungsbestimmung | 2 |
| 3.4 Modul 4: Ausgabe | 3 |
| 4 Umsetzung der einzelnen Module | 3 |
| 4.1 Audiosimulation (Modul 1) | 3 |
| 4.2 Hardware (Modul 1) | 4 |
| 4.2.1 Mikrofonarray | 4 |
| 4.2.2 Audiointerface | 6 |
| 4.2.3 Software | 6 |
| 4.3 Fourier-Transformation (Modul 2) | 6 |
| 4.4 Richtungsmodul (Modul 3) | 7 |
| 4.5 Ausgabemodul (Modul 4) | 7 |
| 4.6 Zentrale Steuerung | 7 |
| 4.7 Testen der einzelnen Module | 8 |
| 5 Eindimensionale Richtungsbestimmung | 9 |
| 6 Zweidimensionale Richtungsbestimmung | 10 |
| 6.1 Erweiterung der Theorie | 10 |
| 6.2 Analytische Lösung des Gleichungssystems | 10 |
| 7 Dreidimensionale Richtungsbestimmung | 11 |
| 7.1 Erweiterung der Theorie | 11 |
| 7.2 Numerische Lösung des Gleichungssystems mit Singulärwertzerlegung und Gauss-Markow Algorithmus | 11 |
| 7.3 Zusammenfassen von ähnlichen Richtungen | 12 |
| 8 Evaluation | 12 |
| 8.1 Audiosimulation | 12 |
| 8.2 Echtwelt | 14 |
| 9 Ausblick | 16 |
| 10 Fazit | 16 |
| Literaturverzeichnis | i |

1 Akustische Richtungsbestimmung

Der Mensch hat die Fähigkeit, direkional zu hören, also die Richtung zu bestimmen, aus der ein Geräusch kommt. Dies bringt ihm enorme Vorteile bei der Erkennung von gesprochener Sprache und bei anderen akustischen Aufgaben. Diese Fähigkeit auf eine technische Apparatur zu übertragen und die Vorteile des räumlichen Hörens auch für diese nutzbar zu machen, hätte viele Anwendungsbereiche, die unser alltägliches Leben erleichtern könnten.

Ein gutes Beispiel für eine solche Anwendung wäre ein Rettungsroboter, der hilfesuchende Menschen anhand von Hilferufen lokalisiert. Auch könnte man verschiedene Hilfsmittel für den Menschen konstruieren, die ein solches technisches Verfahren verwenden. Ein Beispiel hierfür wäre eine Apparatur, die Rettungsteams verwenden können, um in verrauchten Umgebungen Geräuschen wie z.B. schreienden Babys zu folgen.

Weiterhin sind auch Anwendungen aus komplett anderen Anwendungsbereichen denkbar: So wäre es möglich eine Anwendung zu entwickeln, die die Richtungsinformationen verwendet, um Audiosignale zu filtern. Hiervon würde zum Beispiel die Spracherkennung profitieren, bei der störungsfreie Audiosignale benötigt werden und diese mit Hilfe der Richtungsinformationen aus dem Signal gefiltert werden könnten[1]. Auch andere Anwendungen, die störungsfreie Audiosignale benötigen, könnten die Vorteile der Richtungserkennung nutzen.

ist das
besser?

2 Bestehende Lösungen

Es gibt einige bestehende Ansätze, die die verschiedenen Aspekte des direktonalen Hörens auf eine technische Apparatur übertragen. Diesen begegnen wir in unserem alltäglichen Leben relativ häufig. Die einfachste Form eines solchen Verfahrens ist das Richtmikrofon. Dieses führt allerdings keine aktive Ortung durch, sondern kann lediglich in eine bestimmte Richtung besonders gut Schall aufnehmen. Um diese Fähigkeit für eine Richtungsbestimmung zu nutzen, müsste man also das Richtmikrofon drehen, oder es auf eine andere Weise aktiv nach der Schallquelle ausrichten. Ein weiterer Ansatz, dem wir in unserem alltäglichen Leben noch sehr viel häufiger begegnen, steckt in fast allen Mobiltelefonen. Diese filtern beim Telefonieren verschiedene Umgebungsgeräusche aus dem Mikrofonsignal, um die Sprachqualität zu verbessern. Die hierzu verwendeten Verfahren sind allerdings meist eher einfach gehalten und erlauben keine Bestimmung der Herkunftsrichtung eines Geräusches. Das Resultat dieses Verfahrens ähnelt sehr dem eines Richtmikrofons. Bei dieser Technik passiert ein Teil der Geräuschunterdrückung in der Signalverarbeitung, also nach der eigentlichen Schallwandlung durch das Mikrofon. Hierdurch unterscheidet sich dieser Ansatz deutlich von dem des Richtmikrofones. Allerdings können auch mithilfe dieser Störgeräuschunterdrückung noch keine Positionen ermittelt werden. Auch moderne Hörgeräte verwenden ein ähnliches Verfahren, welches allerdings auch bei einer weiteren Distanz zwischen Mikrofon und Schallquelle funktioniert, und der gesuchten Lösung somit näher kommt. Sowohl die Geräuschunterdrückung in Handys, als auch die Filtertechniken in Hörgeräten verwenden meist zwei oder drei Mikrofone.

Eine andere existierende Lösung ist die akustische Kamera (siehe Abb. 1). Sie wird dazu verwendet, lärmemittigende Positionen an Produkten zu finden, um diese optimieren zu können [2]. Die akustische Kamera verwendet allerdings bedeutend mehr Mikrofone als die anderen Verfahren. Einige Modelle verwenden mehr als 350 Mikrofone [3]. Außerdem können bestehende akustische Kameras nicht in andere technische Kontexte eingebettet werden und sind so nicht universell einsetzbar. Da uns alle vorhandenen Lösungen nicht zufrieden gestellt haben, wollten wir ein eigenes Verfahren für die akustischen Richtungsbestimmung entwickeln, das schon mit einer geringen Anzahl von Mikrofonen eine komplette Richtungsbestimmung ermöglicht und leicht erweiterbar auf weitere Auswertungsschritte ist.

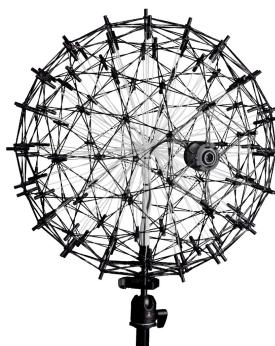


Abbildung 1: Ein Beispiel für eine akustische Kamera [2]

Verfahren für die akustischen Richtungsbestimmung entwickeln, das schon mit einer geringen Anzahl von Mikrofonen eine komplette Richtungsbestimmung ermöglicht und leicht erweiterbar auf weitere Auswertungsschritte ist.

3 Konzept

Bei der Entwicklung unseres neuen Verfahrens zur Richtungsbestimmung haben wir uns stark am menschlichen Gehör und seiner Fähigkeit des Richtungshörens orientiert. Da beim menschlichen Gehör verschiedene Verarbeitungsstufen durchlaufen werden, ist auch unsere Software modular aufgebaut (siehe Tabelle 1). Nachfolgend werden die einzelnen Module in der Reihenfolge, in der das Signal sie durchläuft, beschrieben. Um unser Konzept umzusetzen und zu evaluieren, haben wir uns entschlossen, einerseits eine Computersimulation zu schreiben und andererseits mithilfe einer realen Messapparatur zu überprüfen, ob unser Verfahren auch in der Echtwelt verwendbar ist. Der reale Aufbau soll zuerst in einer Schallkammer, also einem Raum mit wenigen akustischen Störquellen von außen und mit wenigen akustischen Reflexionen an den Wänden, und erst danach unter Einfluss von Störungen getestet werden. Dieses Vorgehen hat den Vorteil, dass man zuerst die Theorie entwickeln kann, und danach die Theorie und ihre Implementation in den verschiedenen Schritten zum Endprodukt immer weiter an Realwelteffekte, wie zum Beispiel Rauschen, anpassen kann. Auch hierbei hilft der modulare Aufbau, da so möglichst viele Programmteile sowohl in der Simulation, als auch in realen Aufbauten verwendet werden können. Unser Konzept sieht vier Module vor, welche über TCP/IP und Websockets verbunden sind. Dies sind Standards, mit denen über ein Computernetzwerk Daten ausgetauscht werden können [13] [9]. Diese beiden Protokolle sind sehr zuverlässig und lange erprobt. Außerdem garantieren sie, dass alle Daten in der Reihenfolge, in der sie losgeschickt werden, ankommen. Die Verwendung von Netzwerkprotokollen hat den Vorteil, dass die Module nicht unbedingt auf demselben Computer ausgeführt werden müssen und so der Rechenaufwand bei Bedarf auf mehrere Computer verteilt werden kann.

| Mensch | Unser Verfahren |
|-------------------------------|---|
| Ohren | Mikrofonarray |
| Haarzellen im Ohr | Fourier Transformation |
| Richtungsbestimmung im Gehirn | Computeralgorithmus zur Richtungsbestimmung |
| Selektives Hören | Weiterverarbeitung durch Programme Dritter |

Tabelle 1: Die Analogie zwischen dem menschlichen Gehör und unserem Verfahren

3.1 Modul 1: Eingabe/Aufnahme

Das erste Modul in dieser Kette stellt die Rohdaten für die weitere Verarbeitung bereit. Diese können entweder von einer Gruppe real existierender Mikrofone stammen oder, im Falle der Simulation, die Signale einer Gruppe simulierter Mikrofone, die simulierte Schallquellen aufnehmen, sein. Dieses Modul entspricht dem menschlichen Ohr und seiner Aufgabe, den Schall aufzunehmen.

3.2 Modul 2: Signal nach Frequenzen aufteilen

Im zweiten Modul der Kette wird das Signal, welches aus dem ersten Modul stammt, in die einzelnen im Signal enthaltenen Wellen aufgeteilt, also von einer zeitaufgelösten Form in ein frequenzaufgelöstes Signal umgewandelt. Nach diesem Schritt liegt also für jede Frequenz eine Amplitude und eine Phase vor. Dieser Schritt wird im menschlichen Ohr durch eine mechanische Konstruktion, die verschiedene Haarzellen für verschiedene Frequenzen anregt, vorgenommen. Unser technisches Verfahren verwendet hierzu die Fouriertransformation. Diese Transformation hat den Vorteil, dass jede Frequenz mit ihrer zugehörigen Amplitude und Phase einzeln analysiert werden kann. In diesem Modul werden außerdem Frequenzen, welche nicht oder nur sehr leise in dem Eingangssignal vorkommen, herausgefiltert, um das Rauschen und den benötigten Rechenaufwand in den nächsten Modulen zu verringern.

3.3 Modul 3: Richtungsbestimmung

Das nächste Modul empfängt die Daten des vorherigen Moduls und errechnet zunächst für jede Frequenz aus den transformierten Daten den Gangunterschied zwischen den Signalen der Mikrofone. Aus diesen werden dann die möglichen Ursprungsrichtungen der einzelnen Sinusschwingung ermittelt. Da dieser Vorgang für jede Frequenz einzeln vorgenommen wird, können auch mehrere Schallquellen mit unterschiedlichen Frequenzen gleichzeitig untersucht werden. Im menschlichen Gehör wird diese Richtungsbestimmung im Gehirn vorgenommen.

3.4 Modul 4: Ausgabe

Das letzte Modul in der Signalkette ist die Ausgabe. Sie bekommt die fertig gruppierten Messergebnisse über eine Netzwerk-Verbindung und bereitet sie für den Nutzer auf. Dieses Modul kann auf die jeweilige Anwendung angepasst werden und es können mehrere Output-Module mit dem gleichen Richtungsmodul verbunden werden. Hier kann dann die Endanwendung, die das Richtungshören verwenden möchte, *angeschlossen* werden.

Der Mensch kann sich zum Beispiel mithilfe der bestimmten Richtungen auf einzelne Schallquellen konzentrieren und andere Geräusche ausblenden.

4 Umsetzung der einzelnen Module

Der Kern unserer Idee ist die Richtungsbestimmung, die aus den Phasendifferenzen der verschiedenen Wellen bei verschiedenen Mikrofonen eine Richtung berechnet. Um diese Komponente überprüfen und das Verfahren anwenden zu können, haben wir die verschiedenen Module implementiert (Siehe Abbildung 2). Hierbei haben wir diese wie vorgesehen so universell umgesetzt, dass wir das Richtungsbestimmungsmodul sowohl mittels einer Simulation als auch mittels eines praktischen Aufbaus überprüfen können. Das nachfolgende Kapitel beschreibt eben diese Umsetzung in der Reihenfolge, mit der das Signal verarbeitet wird; von der Aufnahme der Schallquellen bis zur Ausgabe der Positionsdaten.

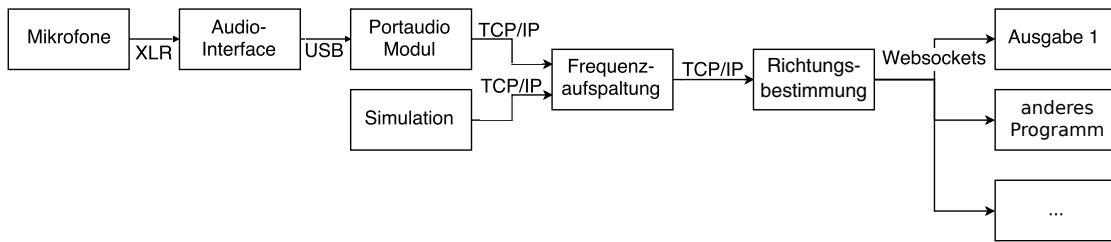


Abbildung 2: Der modulare Aufbau unseres Konzeptes

4.1 Audiosimulation (Modul 1)

Um die Richtungsbestimmung unabhängig von Störfaktoren wie Rauschen und Reflexionen des Schalls an Wänden oder anderen Gegenständen zu überprüfen, haben wir zunächst eine Simulation entwickelt. Außerdem lässt sich mit dieser gezielt der Einfluss verschiedener Störfaktoren untersuchen. Die Simulation kann beliebig viele Mikrofone an beliebigen Positionen simulieren. Leider konnten wir keine bestehenden Lösungen für die Simulation von dreidimensionalem Ton, wie *OpenAL* [11], welches eine Programmmbibliothek für die Simulation von Schall ist, verwenden, da bei diesen die Anzahl der Mikrofone limitiert ist und die Phase des Audiosignals in der Simulation vernachlässigt wird. Da die Funktion unserer Richtungsbestimmung unabhängig von den Amplituden der Schallquellen bei den einzelnen Mikrofonen ist, haben wir eine Simulation geschrieben, die rein auf die Phase und die Laufzeit beschränkt ist. Um eine interaktive Benutzung und eine leichte Überprüfung der Richtungsbestimmung zu gewährleisten, besitzt die Simulation eine graphische Benutzeroberfläche, mit der man interaktiv Schallquellen hinzufügen und entfernen kann. Außerdem kann die Simulation auch die bestimmten Richtungen darstellen und ermöglicht damit einen einfachen Vergleich der bestimmten und tatsächlichen Richtungen. Das Simulationsmodul wurde in der Programmiersprache *C++* implementiert, da diese eine gute *OpenGL* Integration bietet.

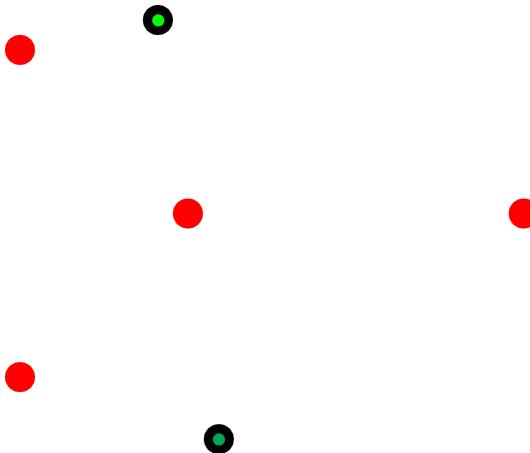


Abbildung 3: Screenshot der Simulation, die roten Punkte stellen die Mikrofone dar, die schwarzen die Schallquellen und die grünen die georteten Positionen

4.2 Hardware (Modul 1)

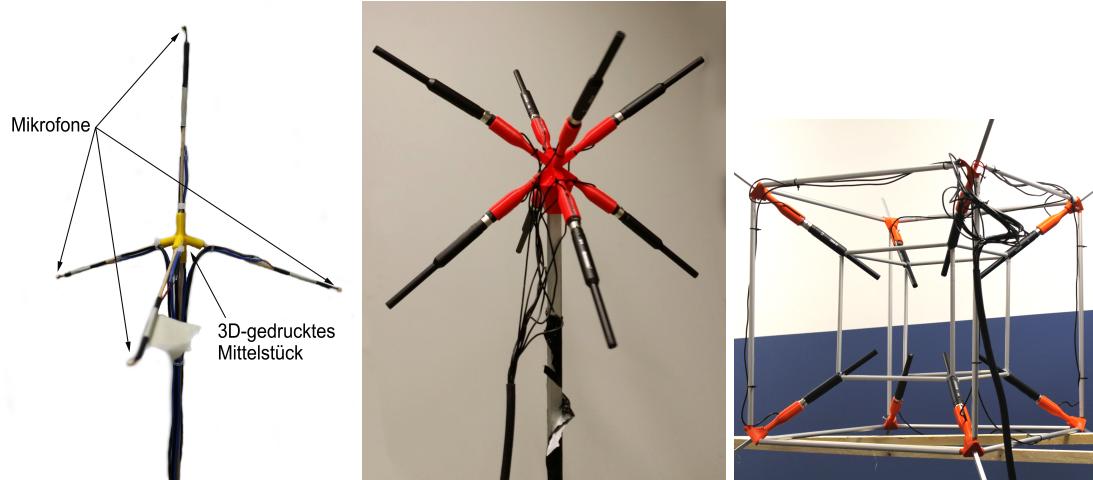


Abbildung 4: Aufbau unserer Messapparatur

Um das durch die Simulation evaluierbare Verfahren praktisch zu testen und zu nutzen, muss nur die Quelle der Daten, also das erste Modul, ausgetauscht werden. Die Aufgabe des ersten Moduls ist es nun nicht mehr, virtuelle Mikrofone zu simulieren, die virtuelle Schallquellen aufnehmen. Stattdessen müssen die Signale echter Mikrofone, welche echte Schallquellen aufnehmen eingelesen werden. Um die Signale der Mikrofone mit einem Computer zu verarbeiten, müssen sie digitalisiert werden. Zusätzlich benötigt man eine alternative Implementation des ersten Moduls, die die Daten von der Hardware annimmt und an das zweite Modul weiterleitet. Hierbei kommen wieder die Vorteile unserer modularen Vorgehensweise zum Tragen, da nur das erste Modul ersetzt werden muss und die gesamte restliche Software beibehalten werden kann. Dies sorgt auch dafür, dass die Simulation und die Realwelttests immer die gleichen Auswertungsalgorithmen verwenden und so sehr gut vergleichbar sind.

add more
LaTeXskill

4.2.1 Mikrofonarray

Die erste Komponente, die es bei einem realen Aufbau der Messapparatur eine wichtige Rolle spielt, ist die der Schallwandlung. Dies wird durch Mikrofone realisiert, an die es einige Anforderungen gibt. Die wichtigste Anforderung ist, dass sie eine möglichst kugelförmige Charakteristik haben. Außerdem erzeugen kleine Mikrofone weniger Schallschatten und Reflexionen. Des Weiteren sollte das Signal-Rausch-Verhältnis möglichst groß sein, da Rauschen die Messungen unpräziser macht [4].

Wir haben drei Prototypen unseres Mikrofonarrays konstruiert und diese dabei Optimiert. Der erste Prototyp unseres Mikrofonarrays bestand aus 4 Mikrofonen, die in einem Tetraeder angeordnet sind, da dies das absolute Minimum für eine dreidimensionale Richtungsbestimmung ist. Für diesen haben wir Elektretmikrofonkapseln verwendet, da diese klein sind und verhältnismäßig gute Signalqualität bieten, und teilweise eine Kugelcharakteristik besitzen. Diese sollte möglichst gleichmäßig sein, da nur so aus allen Richtungen Töne mit gleicher Qualität aufgenommen werden können. Wenn zum Beispiel alle Mikrofone eine Nierencharakteristik aufweisen und sie in einem gleichseitigen Dreieck angeordnet sind, liefert mindestens ein Mikrofon ein deutlich schwächeres Signal als die anderen, was dazu führt, dass die Phasenlage der einzelnen Wellen ungenauer bestimmt werden, und mit ihnen auch die Richtungsbestimmung ungenauer wird. Um unter diesem Aspekt geeignete Elektretmikrofonkapseln zu finden, haben wir die Charakteristiken verschiedener Mikrofone mittels einer selbst entwickelten Messapparatur und einer selbst entwickelten Messsoftware vermessen. Die hierfür entwickelte Messapparatur sendet mittels eines Lautsprechers eine Sinus-Schwingung aus und misst, wie stark diese vom Mikrofon aufgenommen wurde. Danach dreht sie das Mikrofon um einen festgelegten Winkel weiter und fertigt erneut eine Messung an. Dieser Vorgang wird solange wiederholt, bis das Mikrofon einmal um 360° gedreht wurde.

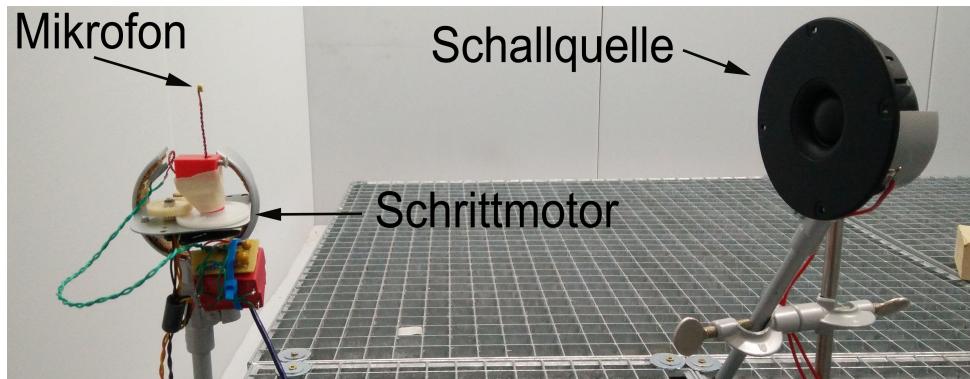


Abbildung 5: Unsere Messapparatur für die Charakteristik eines Mikrofons

Die so ermittelten Daten können nun mittels des freien Plottingprogramms *gnuplot* [15] visualisiert werden, um die Richtcharakteristik abzulesen.

messapparatur
raus?

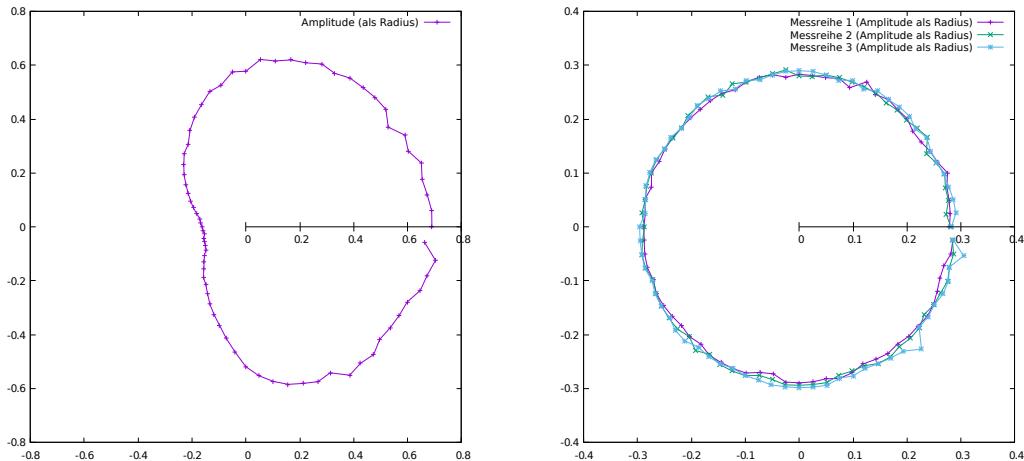


Abbildung 6: Auf den beiden Grafiken ist die gemessene Amplitude des Mikrofons über den Winkel aufgetragen. Bei der linken Grafik kann man feststellen, dass das vermessene Mikrofon eine sehr ungleichmäßige Charakteristik aufweist, diese Mikrofone haben wir zu Anfang verwendet. Auf der rechten Seite ist die Charakteristik der Mikrofone zu sehen, die wir in unserem aktuellen Testaufbau verwenden. Diese ist sehr gleichmäßig, weshalb wir diese Mikrofone verwendet haben.

Dadurch, dass unser Tetraeder symmetrisch ist, sind die Fehler in alle Richtungen ungefähr gleich groß und die Richtungsbestimmung ist in allen Richtungen gleich gut. Um die Charakteristik der Mikrofone möglichst wenig zu verändern, haben wir die Mikrofone nur an ihrem Kabel mit dem Tetraeder verbunden. Dadurch ist der Schallschatten durch den Tetraeder relativ gering. Das Mittelstück des Tetraeders

haben wir mit Hilfe eines 3D-Druckers hergestellt (siehe Abbildung 4).

Der zweite Optimierungsschritt des Mikrofonarrays beinhaltete eine Erhöhung der Mikrotonanzahl von 4 auf 8 und einen Umstieg auf eine würfelförmige Anordnung. Hierdurch stehen mehr Informationen für den Richtungsbestimmungsalgorithmus zur Verfügung und das Gleichungssystem wird überbestimmt. Auch sind wir von Elektretmikrofonkapseln auf Professionelle Messmikrofone umgestiegen, da diese eine Definierte Charakteristik haben und ein besseres Signal-Rausch Verhältnis aufweisen.

Da beim zweiten Prototyp des Mikrofonarrays die Mikrofone der Abstand zwischen den Mikrofonen sehr groß war und dadurch das nutzbare Frequenzspektrum nach oben hin sehr stark eingeschränkt wurde, war das Ziel des dritten Prototyps diese wieder näher aneinander zu rücken. Dieses Ziel erreichten wir durch eine Umkehrung der Mikrotonrichtung, so dass diese nun nicht mehr von innen nach außen, sondern von Außen nach innen zeigen.

4.2.2 Audiointerface

Auch für das AudioInterface, also die Verbindung zwischen Mikrofonen und Computer, gibt es bestimmte Voraussetzungen. So benötigt unser Verfahren mindestens 4 Mikrofone, jedoch lässt es sich einfach auf mehr Kanäle erweitern, was der Genauigkeit zugute kommt. Daher haben wir ein AudioInterface mit möglichst vielen Kanälen gesucht. Eine weitere wichtige Anforderung an das AudioInterface ist eine hohe Auflösung, da hierdurch die Signalqualität verbessert wird und digitale Verstärkung bei ausreichender Audioqualität möglich ist. Um die Elektretmikrofonkapseln an das AudioInterface anzuschließen, benötigt man zusätzlich eine Schaltung, welche das unsymmetrische Signal der Elektretmikrofonkapsel in ein symmetrisches Signal für das AudioInterface umwandelt. Außerdem muss diese die Phantomspeisung, die das AudioInterface bereitstellt und eine Spannung von 48V hat, in eine Tonaderspeisung für das Mikrofon konvertieren. Hierfür kommt die Schaltung von [5] zum Einsatz.

4.2.3 Software

Um die echten Mikrofone für die Richtungsbestimmung zu verwenden, muss noch eine Verbindung zwischen dem AudioInterface und dem nächsten Modul geschaffen werden. Durch unseren modularen Aufbau lässt sich dies leicht implementieren. Wir haben dazu ein Programm in Java entwickelt, das fähig ist, mehrkanalige Audiosignale in Echtzeit aufzunehmen und über TCP/IP an die Fourier-Transformation weiterzuleiten. Zur Umsetzung haben wir die Programmbibliothek *portaudio* [6] verwendet. Diese Programmbibliothek hat den Vorteil, dass mehrere Audiokanäle zeitsynchronisiert eingelesen werden können, was sehr wichtig ist, damit unser Verfahren zur Richtungsbestimmung, welches auf der relativen Phasenlage basiert, funktioniert. *Portaudio* haben wir in Java über das Java Native Interface (JNI) benutzt.

4.3 Fourier-Transformation (Modul 2)

Dieses Modul teilt die Audio-Signale in einzelne Sinuswellen und bestimmt deren Phase und Amplitude. Um dies zu bewerkstelligen, lässt sich eine diskrete Fourier-Transformation verwenden. Die diskrete Fourier-Transformation bestimmt aus einem zeitdiskreten Signal die einzelnen Sinusschwingungen mit ihrer zugehörigen Phase und Amplitude, die zusammen das Signal bilden. Ein schneller Algorithmus, um die diskrete Fourier-Transformation eines Signals zu berechnen, ist die Fast Fourier-Transformation (FFT). Dieser ist schnell genug, um eine Echtzeitverarbeitung des Signals zu ermöglichen. Als Implementation der FFT haben wir *FFTW*[10] verwendet, da *FFTW* kostenlos, opensource und vergleichsweise schnell ist. Das Fourier-Transformations-Modul wurde aus Performancegründen in C++ implementiert. Durch eine diskrete Fourier-Transformation von n reellen Zahlen erhält man eine Liste aus n komplexen Zahlen. Eine komplexe Zahl z an der Stelle j enthält die Amplituden- und Phaseninformation für die Frequenz f :

$$f = \frac{j \cdot r}{n}$$

Wobei r die Abtastrate des Signals ist. Die Amplitude A des Sinus lässt sich mit dem Betrag der komplexen Zahl berechnen und die Phase ϕ mit dem Arcus-Tangens. Dies entspricht der Koordinatentransformation von einem kartesischen in ein polares Koordinatensystem:

$$A = \sqrt{\Re(z)^2 + \Im(z)^2}$$

$$\phi = \text{atan2}(\Im(z), \Re(z))$$

$$\text{atan2}(y, x) := \begin{cases} \arctan \frac{y}{x} & \text{für } x > 0 \\ \arctan \frac{y}{x} + \pi & \text{für } x < 0, y \geq 0 \\ \arctan \frac{y}{x} - \pi & \text{für } x < 0, y < 0 \\ +\pi/2 & \text{für } x = 0, y > 0 \\ -\pi/2 & \text{für } x = 0, y < 0 \\ 0 & \text{für } x = 0, y = 0 \end{cases}$$

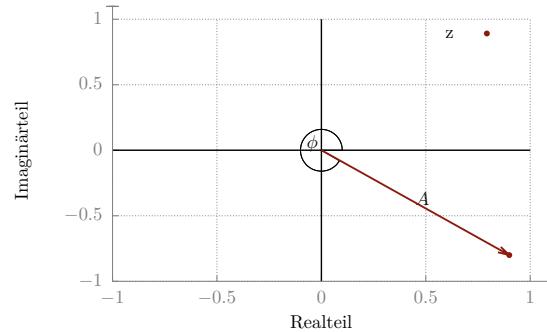


Abbildung 7: Transformation von kartesischen zu polaren Koordinatensystem

Die Verwendung von $\text{atan2}(y, x)$ anstelle von $\arctan \frac{y}{x}$ sorgt dafür, dass der richtige Winkel berechnet wird. $\arctan \frac{y}{x}$ liefert nur Winkel von -90° bis 90° , deswegen muss anhand des Vorzeichens von x und y bestimmt werden, in welchem Quadranten der Punkt liegt und der von $\arctan \frac{y}{x}$ gelieferte Winkel dementsprechend verschoben werden.

Das in Frequenz, Phase und Amplitude konvertierte Ergebnis der Fourier-Transformation wird dann gefiltert. Alle Frequenzen mit einer Amplitude, die kleiner als eine bestimmte Untergrenze sind, werden verworfen. Die verbleibenden Frequenzen werden an das Richtungsmodul übermittelt.

4.4 Richtungsmodul (Modul 3)

Die von der Fourier-Transformation bestimmten Tripel aus Frequenz, Phase und Amplitude werden vom Richtungsmodul weiter verarbeitet. In das Richtungsmodul können verschiedene Methoden der Richtungsbestimmung eingesetzt werden. Die ermittelten Richtungen werden an die Ausgabe weiter gesendet, um sie dort auszugeben, oder zur Filterung des Audio-Signals zu verwenden. Das Richtungsmodul haben wir in C++ geschrieben und verwendet LAPACK[8].

4.5 Ausgabemodul (Modul 4)

Wir haben ein Beispiel-Ausgabemodul in Javascript implementiert. Mit diesem ist es möglich, die Positionsdaten zu visualisieren, was das unmittelbare Überprüfen unseres Verfahrens stark vereinfacht. Die Programmiersprache Javascript haben wir gewählt, damit dieses Ausgabemodul auf jedem Endgerät mit modernem Webbrowser, wie z.B. Smartphones oder Laptops, ausgeführt werden kann.

Die Simulation enthält eine weitere Implementation eines Ausgabemoduls. Auch diese visualisiert die bestimmten Richtungen, erlaubt es aber, diese auf der gleichen Benutzeroberfläche wie die Sollrichtung anzuzeigen. Dies gibt ein sehr direktes Feedback beim Entwickeln des Algorithmus zur Richtungsbestimmung. Dieses letzte Modul könnte allerdings, dank unseres modularen Konzeptes, bei Bedarf auch anders, z.B. als Plugin für eine *Digital Audio Workstation (DAW)* implementiert werden.

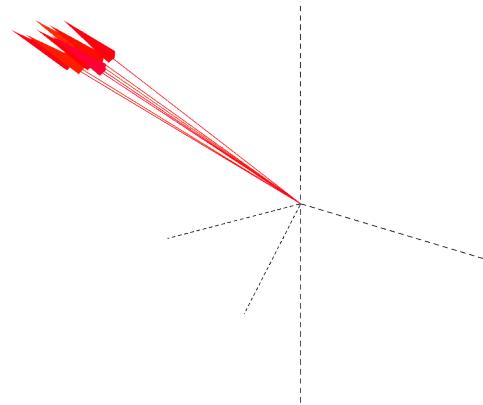


Abbildung 8: Screenshot unseres Beispieldatenmoduls

4.6 Zentrale Steuerung

Um die Konfiguration der einzelnen Module zu vereinfachen und schneller an neue Situationen anpassen zu können haben wir ein Programm entwickelt, dass alle Module mit der gewünschten Konfiguration startet und mit dem man alle Module konfigurieren kann. Außerdem überwacht das Programm die Ausführung

aller Module, gibt etwaige Fehlermeldungen aus und startet die einzelnen Module nach kritischen Fehlern neu.

sehr geile
formulie-
rung ;)

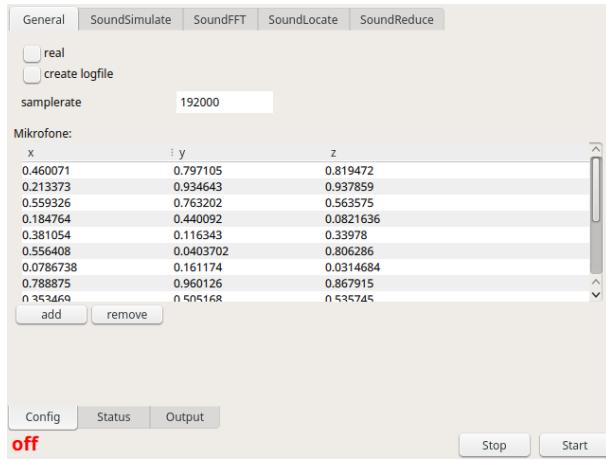


Abbildung 9: Screenshot der Steuerungsoberfläche

4.7 Testen der einzelnen Module

Ein weiterer Vorteil der Modularität ist, dass jedes Modul unabhängig von den anderen Modulen funktionsfähig ist. Dadurch kann die korrekte Funktionsweise für jedes Modul einzeln überprüft werden und man kann Fehler besser lokalisieren. Die Audiosimulation haben wir mithilfe eines selbstgeschriebenen Plottingprogramms überprüft. Dieses stellt die von der Audiosimulation versendeten Samples in Abhängigkeit zur Zeit dar. Damit kann manuell die Phasendifferenz von dem Graphen abgelesen und mit dem erwarteten Wert verglichen werden.

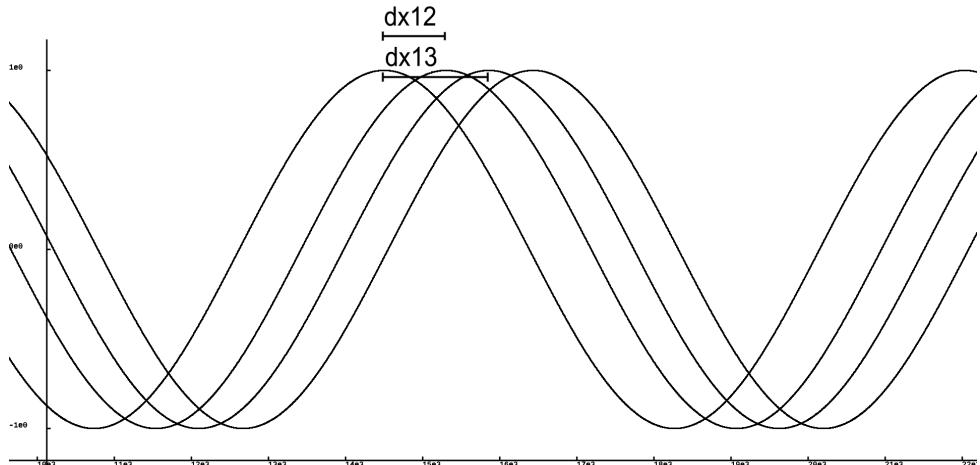


Abbildung 10: Screenshot unseres Plottingprogramms

Die Fourier-Transformation konnten wir mit der vorher überprüften Audiosimulation testen. Dazu haben wir die von der Fourier-Transformation bestimmten Tripel aus Frequenz, Phase und Amplitude mit den tatsächlich in der Simulation eingestellten Werten verglichen. Durch das Testen der einzelnen Module konnten wir effizient die vorhandenen Fehler, wie die falsche Berechnung der Phase und einen Fehler in der Distanzberechnung der Audiosimulation, finden und beheben.

5 Eindimensionale Richtungsbestimmung

Um den Algorithmus, der aus den Phasendifferenzen die Richtung errechnet, zu entwickeln, haben wir mit der einfachsten Stufe der Richtungsbestimmung, der eindimensionalen Richtungsbestimmung, angefangen:

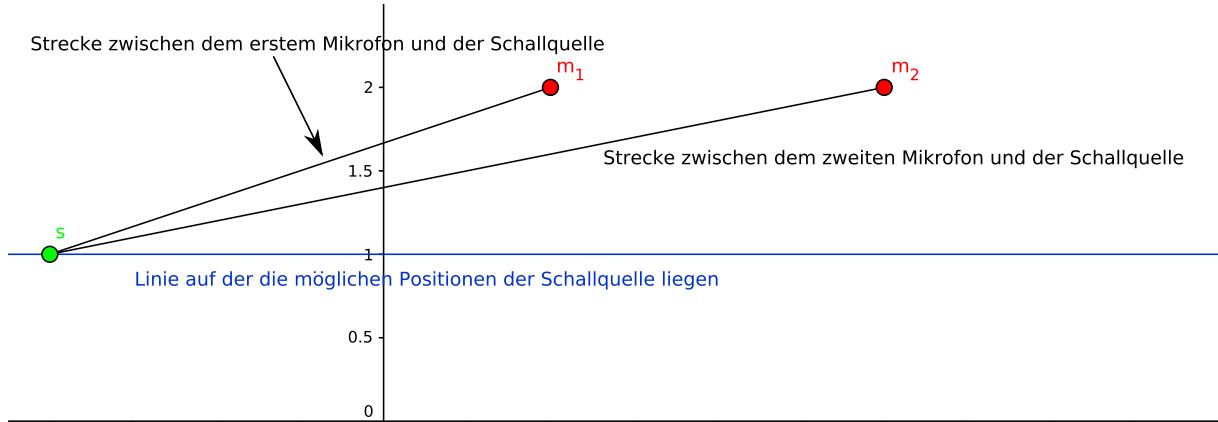


Abbildung 11: Skizze einer eindimensionalen Richtungsbestimmung

In Abbildung 11 sieht man zwei Mikrofone M_1 und M_2 , die den Schall der Schallquelle S aufnehmen. Dadurch, dass M_2 weiter von der Schallquelle entfernt ist als M_1 , braucht der Schall länger, um M_2 zu erreichen. Diesen Zeitunterschied sieht man in den von den Mikrofonen aufgenommenen Signalen als Gangunterschied. Man kann von den Wellen, die durch die Mikrofone aufgenommen werden, nur die Phasenverschiebung bestimmen. Deswegen können wir den Gangunterschied nicht direkt bestimmen, er setzt sich aus einer unbekannten Anzahl von kompletten Schwingungen und der Differenz d der Phasenverschiebungen zusammen. Man erhält für die Differenz der Abstände einer Schallquelle S von zwei Mikrofonen M_1 und M_2 mit der gemessenen Phase ϕ_1 und ϕ_2 :

$$d = n\lambda + \Delta x_{12}$$

λ ist hierbei die Wellenlänge, n die unbekannte Anzahl an Schwingungen und Δx_{12} der Gangunterschied, den man aus dem Phasenunterschied bestimmen kann:

$$\Delta x_{ij} = \frac{c(\phi_i - \phi_j)}{2\pi f} \text{ mit } i = 1 \text{ und } j = 2$$

c ist die Schallgeschwindigkeit und f die Frequenz des Schallwellen. Jedes Mikrofon M_i hat einen zugehörigen Ortsvektor $\vec{m}_i = \begin{pmatrix} m_{ix} \\ m_{iy} \end{pmatrix}$, die Schallquelle hat den Ortsvektor $\vec{s} = \begin{pmatrix} s_x \\ s_y \end{pmatrix}$. Der Abstand eines Mikrofons von der Schallquelle ist gegeben durch den Satz des Pythagoras:

$$|\vec{m}_i - \vec{s}| = \sqrt{(m_{ix} - s_x)^2 + (m_{iy} - s_y)^2}$$

Damit erhält man für die Differenz der Abstände der Mikrofone die Gleichung

$$|\vec{m}_1 - \vec{s}| - |\vec{m}_2 - \vec{s}| = n\lambda + \Delta x_{12}$$

$$\sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{2x} - s_x)^2 + (m_{2y} - s_y)^2} = n\lambda + \Delta x_{12}$$

s_y ist konstant für alle möglichen Positionen der Schallquelle. Dadurch hat die Gleichung aber immer noch zwei Unbekannte, s_x und n , und ist deswegen nicht eindeutig lösbar. Man kann eine eindeutig lösbare Gleichung erhalten, wenn der Abstand der Mikrofone maximal halb so groß wie die Wellenlänge ist. Dann

macht die Welle maximal eine halbe Schwingung mehr zu einem Mikrofon als zu dem anderen und die unbekannte Anzahl an Schwingungen $n\lambda$ entfällt. Dann hat die Gleichung nur eine Unbekannte, s_x

$$|\vec{m}_1 - \vec{s}| - |\vec{m}_2 - \vec{s}| = \Delta x_{12}$$

$$\sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{2x} - s_x)^2 + (m_{2y} - s_y)^2} = \Delta x_{12}$$

Den dadurch bestimmten Ortsvektor kann man dann in einen Strahl g umwandeln, der in die Richtung der Schallquelle weist:

$$g : \vec{x} = \vec{m} + r \cdot (\vec{s} - \vec{m}) \quad r \in \mathbb{R}^+$$

\vec{m} entspricht dabei der Mitte der Mikrofone und \vec{x} einer beliebigen Position auf der Geraden.

Da eine eindimensionale Ortung sehr einfach und ohne großen Anwendungsbereich ist, haben wir diesen Algorithmus anschließend auf zwei Dimensionen übertragen.

6 Zweidimensionale Richtungsbestimmung

6.1 Erweiterung der Theorie

Um die eindimensionale Richtungsbestimmung auf den zweidimensionalen Raum zu erweitern, muss man lediglich s_y nicht mehr als konstant betrachten. Dadurch hat die Gleichung, die für das Eindimensionale eindeutig war, nun zwei Unbekannte: s_x und s_y . Um im Zweidimensionalen eine Ortung durchzuführen, wird deswegen ein drittes Mikrofon M_3 benötigt. Auch der Abstand zwischen dem dritten Mikrofon und dem ersten darf maximal so groß wie die Wellenlänge sein. Mit dem dritten Mikrofon kann eine weitere Gleichung, die den Gangunterschied zwischen dem ersten und dem dritten Mikrofon Δx_{13} enthält, aufgestellt werden. Man erhält das Gleichungssystem:

$$\left| \begin{array}{l} |\vec{m}_1 - \vec{s}| - |\vec{m}_2 - \vec{s}| = \Delta x_{12} \\ |\vec{m}_1 - \vec{s}| - |\vec{m}_3 - \vec{s}| = \Delta x_{13} \end{array} \right|$$

$$\left| \begin{array}{l} \sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{2x} - s_x)^2 + (m_{2y} - s_y)^2} = \Delta x_{12} \\ \sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{3x} - s_x)^2 + (m_{3y} - s_y)^2} = \Delta x_{13} \end{array} \right|$$

Graphisch veranschaulicht entspricht dieses Gleichungssystem der Suche nach dem Schnittpunkt von zwei halben Hyperbeln um zwei der Mikrofone

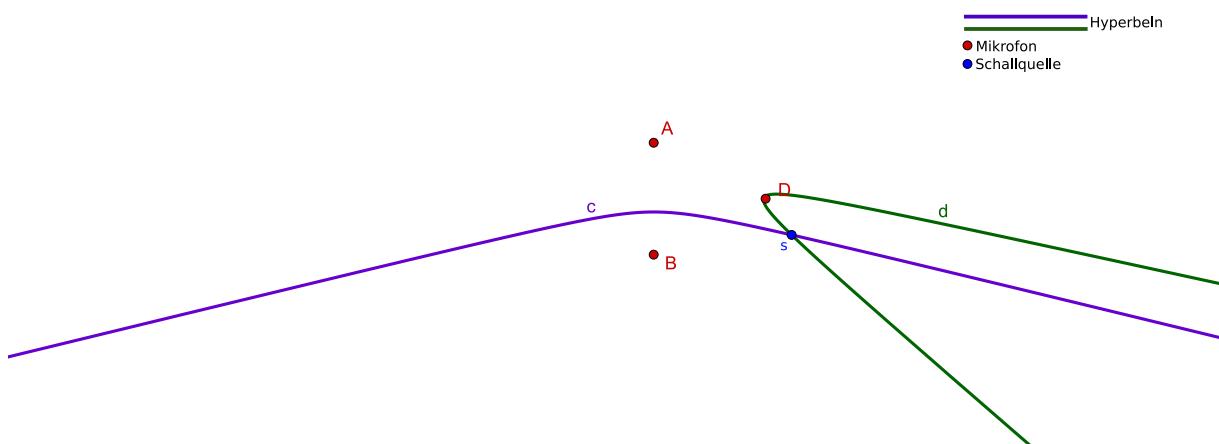


Abbildung 12: Veranschaulichung des Gleichungssystems durch den Schnittpunkt von zwei Hyperbeln. Die roten Punkte stellen die Mikrofone dar und der blaue Punkt die Schallquelle

6.2 Analytische Lösung des Gleichungssystems

Dieses Gleichungssystem haben wir mit Hilfe des Computer Algebra Systems *Wolfram Mathematica*[14] nach \vec{s} aufgelöst:

Listing 1: Befehl um das Gleichungssystem in Mathematica zu lösen

```

1 Solve[{  
2   Sqrt[(m1x - sx)^2 + (m1y - sy)^2] - Sqrt[(m2x - sx)^2 + (m2y - sy)^2] = dx12,  
3   Sqrt[(m1x - sx)^2 + (m1y - sy)^2] - Sqrt[(m3x - sx)^2 + (m3y - sy)^2] = dx13,}  
4 {sx, sy}]
```

Da das Gleichungssystem nicht linear ist, ist die Lösung sehr kompliziert, gedruckt entspricht sie fünf Seiten. *Mathematica* findet für dieses Gleichungssystem außerdem zwei allgemeine reelle Lösungen, man erhält also zwei mögliche Richtungen.

7 Dreidimensionale Richtungsbestimmung

7.1 Erweiterung der Theorie

Um die Richtungsbestimmung dann auf drei Dimensionen zu erweitern, müssen zuerst die Ortsvektoren der Mikrofone und der Schallquelle auf drei Dimensionen erweitert werden:

$$\vec{m}_i = \begin{pmatrix} m_{ix} \\ m_{iy} \\ m_{iz} \end{pmatrix} \quad \vec{s} = \begin{pmatrix} s_x \\ s_y \\ s_z \end{pmatrix}$$

Mit den angepassten Ortsvektoren erhält man eine neue Formel für die Abstandsberechnung zwischen einem Mikrofon und der Schallquelle:

$$|\vec{m}_i - \vec{s}| = \sqrt{(m_{ix} - s_x)^2 + (m_{iy} - s_y)^2 + (m_{iz} - s_z)^2}$$

Man erhält eine weitere Unbekannte, m_{iz} . Dadurch wird für die dreidimensionale Ortung ein vierter Mikrofon benötigt. Mit dem vierten Mikrofon erhält man eine dritte Gleichung, die den Gangunterschied zwischen dem ersten und dem vierten Mikrofon enthält, dadurch wird das Gleichungssystem wieder eindeutig lösbar:

$$\left| \begin{array}{l} |\vec{m}_1 - \vec{s}| - |\vec{m}_2 - \vec{s}| = \Delta x_{12} \\ |\vec{m}_1 - \vec{s}| - |\vec{m}_3 - \vec{s}| = \Delta x_{13} \\ |\vec{m}_1 - \vec{s}| - |\vec{m}_4 - \vec{s}| = \Delta x_{14} \\ \\ \sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{2x} - s_x)^2 + (m_{2y} - s_y)^2 + (m_{2z} - s_z)^2} = \Delta x_{12} \\ \sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{3x} - s_x)^2 + (m_{3y} - s_y)^2 + (m_{3z} - s_z)^2} = \Delta x_{13} \\ \sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2} - \sqrt{(m_{4x} - s_x)^2 + (m_{4y} - s_y)^2 + (m_{4z} - s_z)^2} = \Delta x_{14} \end{array} \right|$$

7.2 Numerische Lösung des Gleichungssystems mit Singulärwertzerlegung und Gauss-Markow Algorithmus

Schon für zwei Dimensionen ist die analytische Lösung des Gleichungssystem sehr kompliziert. Für drei Dimensionen ist die Formel der Lösung in Textform über 300Mb groß. Dadurch können wir sie nur durch Einschränkung der Mikrofon-Positionen verwenden. Um diese Einschränkung zu umgehen, lösen wir das Gleichungssystem numerisch. Dazu verwenden wir das mehrdimensionale Newtonverfahren. Hierbei gehen wir von einer Startposition \vec{s}_0 aus, die iterativ verbessert wird. Als Startposition haben wir das Zentrum der Mikrofone gewählt. Für den Iterationsvektoren erhält man eine neue Formel für die Abstandsberechnung zwischen einem Mikrofon und der Schallquelle:

$$|\vec{m}_i - \vec{s}| = \sqrt{(m_{ix} - s_x)^2 + (m_{iy} - s_y)^2 + (m_{iz} - s_z)^2}$$

Man erhält eine weitere nsschritt i erhält man die neuen Positionen mit $\vec{s}_{i+1} = \vec{s}_i + \vec{\Delta s}_i$. Das Gleichungssystem für die dreidimensionale Ortung lässt sich umschreiben zu:

$$\left| \begin{array}{l} \sqrt{(m_{1x} - s_x)^2 + (m_{1y} - s_y)^2 + (m_{1z} - s_z)^2} - d = 0 \\ \sqrt{(m_{2x} - s_x)^2 + (m_{2y} - s_y)^2 + (m_{2z} - s_z)^2} - d = \Delta x_{12} \\ \sqrt{(m_{3x} - s_x)^2 + (m_{3y} - s_y)^2 + (m_{3z} - s_z)^2} - d = \Delta x_{13} \\ \sqrt{(m_{4x} - s_x)^2 + (m_{4y} - s_y)^2 + (m_{4z} - s_z)^2} - d = \Delta x_{14} \end{array} \right|$$

Um dieses Gleichungssystem mit dem Newtonverfahren zu lösen wird es mit der Taylorentwicklung bis zur ersten Ordung linearisiert:

$$\left| \begin{array}{l} -r_{1ca} = \frac{s_{ix}-m_{1x}}{r_{1ca}} \cdot \vec{\Delta s}_{ix} + \frac{s_{iy}-m_{1y}}{r_{1ca}} \cdot \vec{\Delta s}_{iy} + \frac{s_{iz}-m_{1z}}{r_{1ca}} \cdot \vec{\Delta s}_{iz} - d \\ \Delta x_{12} - r_{2ca} = \frac{s_{ix}-m_{2x}}{r_{2ca}} \cdot \vec{\Delta s}_{ix} + \frac{s_{iy}-m_{2y}}{r_{2ca}} \cdot \vec{\Delta s}_{iy} + \frac{s_{iz}-m_{2z}}{r_{2ca}} \cdot \vec{\Delta s}_{iz} - d \\ \Delta x_{13} - r_{3ca} = \frac{s_{ix}-m_{3x}}{r_{3ca}} \cdot \vec{\Delta s}_{ix} + \frac{s_{iy}-m_{3y}}{r_{3ca}} \cdot \vec{\Delta s}_{iy} + \frac{s_{iz}-m_{3z}}{r_{3ca}} \cdot \vec{\Delta s}_{iz} - d \\ \Delta x_{14} - r_{4ca} = \frac{s_{ix}-m_{4x}}{r_{4ca}} \cdot \vec{\Delta s}_{ix} + \frac{s_{iy}-m_{4y}}{r_{4ca}} \cdot \vec{\Delta s}_{iy} + \frac{s_{iz}-m_{4z}}{r_{4ca}} \cdot \vec{\Delta s}_{iz} - d \end{array} \right|$$

$$r_{ica} = \sqrt{(m_{ix} - s_{ix})^2 + (m_{iy} - s_{iy})^2 + (m_{iz} - s_{iz})^2}$$

$$\begin{pmatrix} \vec{\Delta s}_{ix} \\ \vec{\Delta s}_{iy} \\ \vec{\Delta s}_{iz} \\ d \end{pmatrix} = \begin{pmatrix} \frac{s_{ix}-m_{1x}}{r_{1ca}} & \frac{s_{iy}-m_{1y}}{r_{1ca}} & \frac{s_{iz}-m_{1z}}{r_{1ca}} & -1 \\ \frac{s_{ix}-m_{2x}}{r_{2ca}} & \frac{s_{iy}-m_{2y}}{r_{2ca}} & \frac{s_{iz}-m_{2z}}{r_{2ca}} & -1 \\ \frac{s_{ix}-m_{3x}}{r_{3ca}} & \frac{s_{iy}-m_{3y}}{r_{3ca}} & \frac{s_{iz}-m_{3z}}{r_{3ca}} & -1 \\ \frac{s_{ix}-m_{4x}}{r_{4ca}} & \frac{s_{iy}-m_{4y}}{r_{4ca}} & \frac{s_{iz}-m_{4z}}{r_{4ca}} & -1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} -r_{1ca} \\ \Delta x_{12} - r_{2ca} \\ \Delta x_{13} - r_{3ca} \\ \Delta x_{14} - r_{4ca} \end{pmatrix}$$

Für die Berechnung der Inverse der Matrix gibt es viele verschiedene Methoden, wir benutzen die Singulärwertszerlegung. Die numerische Lösung des Gleichungssystems benötigt eine initiale Position, für die die Mitte der Mikrofone verwendet wird. Wenn allerdings bereits eine alte Position für die vorliegende Frequenz bekannt ist, wird diese alte Position als Startwert der Iteration verwendet, da dadurch, wenn sich die Position der Schallquelle nicht zu sehr geändert hat, die Iteration schneller konvergiert. Als Konvergenzkriterium verwenden wir $|\vec{\Delta s}_i| < 0.0001\text{m}$. Da das Newtonverfahren keine garantiierte Konvergenz hat, wird die Iteration nach 50 Iterationsschritten abgebrochen. Ein Vorteil der numerischen Lösung ist, dass sie leicht auf beliebig viele Mikrofone erweitert werden kann. Das Gleichungssystem wird überbestimmt, wenn man mehr als vier Mikrofone verwendet, aber indem man die Methode der kleinsten Quadrate auf die Berechnung von $\vec{\Delta s}_i$ anwendet, erhält man den Ort, an dem die Summe der Fehlerquadrate für alle Mikrofone am geringsten ist, es also am wahrscheinlichsten ist, dass sich die Schallquelle wirklich dort befindet. Die Least Squares Lösung entspricht für unser Problem, nach dem Gauss-Markow-Theorem einem minimalvarianten linearen erwartungstreuen Schätzer. Außerdem kann man mit mehr Mikrofonen die Richtungsbestimmung auch unabhängig von der Schallgeschwindigkeit machen, denn diese kann als eine weitere Variable eingeführt werden.

7.3 Zusammenfassen von ähnlichen Richtungen

Signale in der Echtwelt bestehen aus vielen verschiedenen Frequenzen, dadurch erhält man anstatt einer Richtung für eine Schallquelle, wie eine sprechender Mensch, nicht nur eine Richtung, sondern viele ähnliche Richtungen. Um die ermittelten Richtungen eindeutiger zu machen fassen wir deswegen diese ähnliche Richtungen zusammen. Dazu verwenden wir ein agglomeratives Clusterverfahren mit der Summe der minimalen Winkel zwischen zwei Richtungen als Unähnlichkeit. Von jeder Frequenz werden die letzten zehn Richtungen gespeichert und diese fließen mit in die Clusterbildung mit ein. Um geringe Änderungen in der Richtung auszugleichen wird außerdem ein laufendes Mittel der Richtung für jede Frequenz über drei Richtungen gebildet. Die Richtung eines Clusters wird über ein nach dem Amplituden gewichtetes Mittel aus allen zu dem Cluster gehörenden Richtungen berechnet.

8 Evaluation

Um unser Verfahren zur Richtungsbestimmung zu testen und die Genauigkeit zu bestimmen, haben wir es in Simulation und Echtwelt evaluiert. Dabei haben wir die Abweichung der Richtungsbestimmung für verschiedene Richtungen und Frequenzen bestimmt.

8.1 Audiosimulation

Zunächst haben wir unser Verfahren in der Simulation getestet, da diese automatisierte Tests ermöglicht und somit leicht sehr viele verschiedene Richtungen und Frequenzen getestet werden können. Dabei haben wir die Schallquellen für verschiedene Radien auf einer Kugeloberfläche positioniert. Der Tetraeder wurde mit 0.28m Kantenlänge simuliert, dies entspricht dem Tetraeder, den wir auch in echt gebaut haben. Die Schallquellen wurden mit einer 500Hz-Sinusschwingung simuliert.

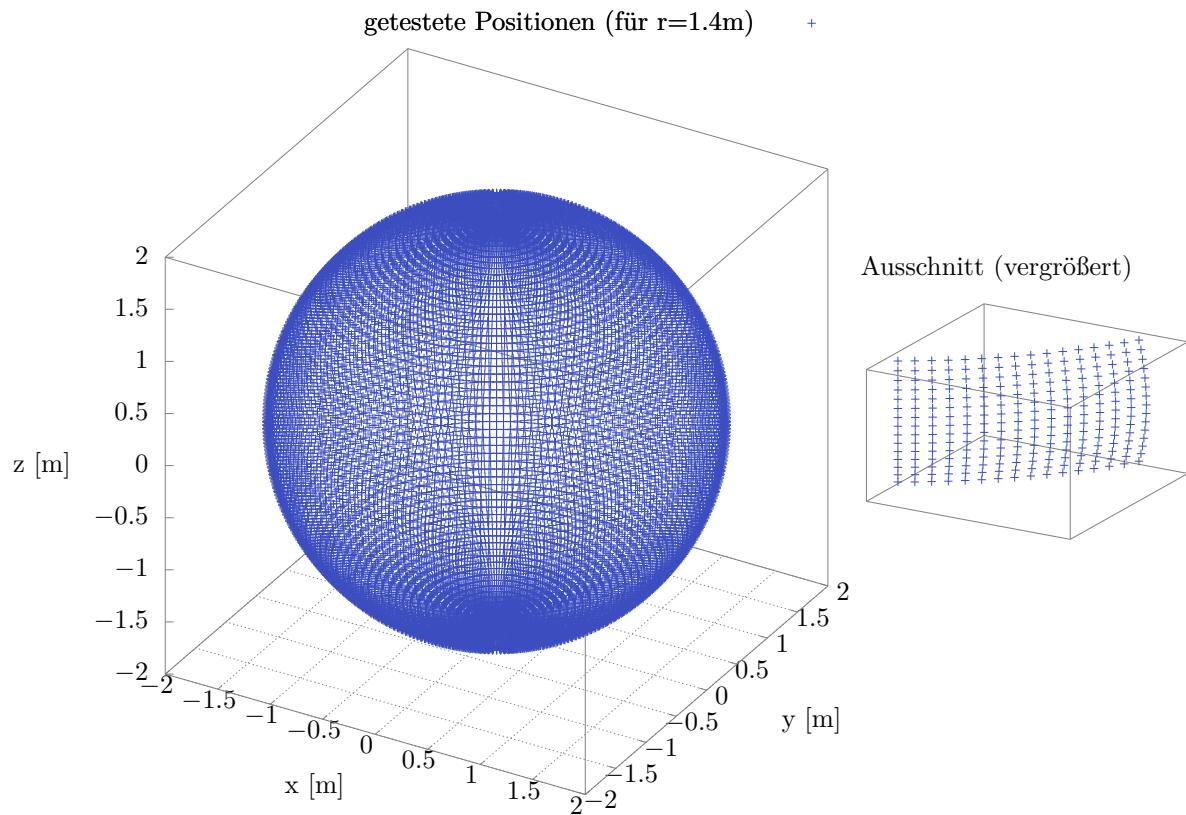


Abbildung 13: Visualisierung der getesteten Positionen

Für jeden getesteten Radius haben wir Punkte auf der Kugeloberfläche in 2° Auflösung, also insgesamt 32400 Positionen, getestet.

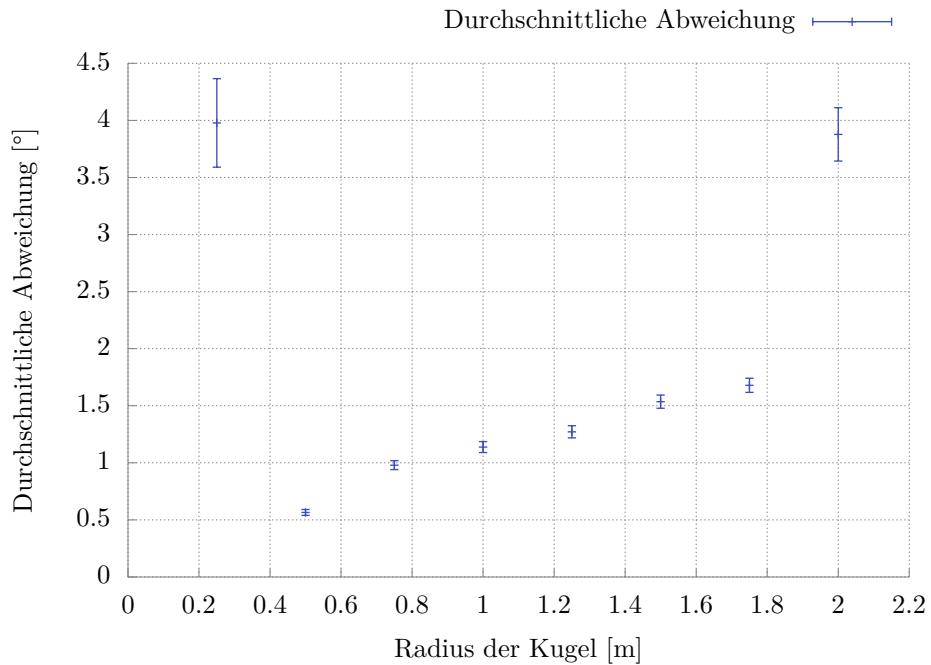


Abbildung 14: Genauigkeit für verschiedene Abstände

Bei einem Radius von 0.25m gibt es einen Ausreißer mit $4.0^\circ \pm 0.4^\circ$, da sich die Positionen innerhalb des Tetraeders befinden. Bis 1.75m bleibt die Abweichung deutlich unter 2° , ist also deutlich besser als beim Menschen. Dieser kann auf sehr kurze Distanzen zwischen 10cm und 40cm gerade einmal mit 3° Genauigkeit die Richtung bestimmen. [12] Um die Abhängigkeit der Genauigkeit der Richtungsbestimmung von der Frequenz zu untersuchen, haben wir die Schallquellen wieder auf einer Kugeloberfläche mit 2° Auflösung platziert und haben anstelle des Radius die Frequenz variiert. Für den Radius haben wir 1.4m gewählt.

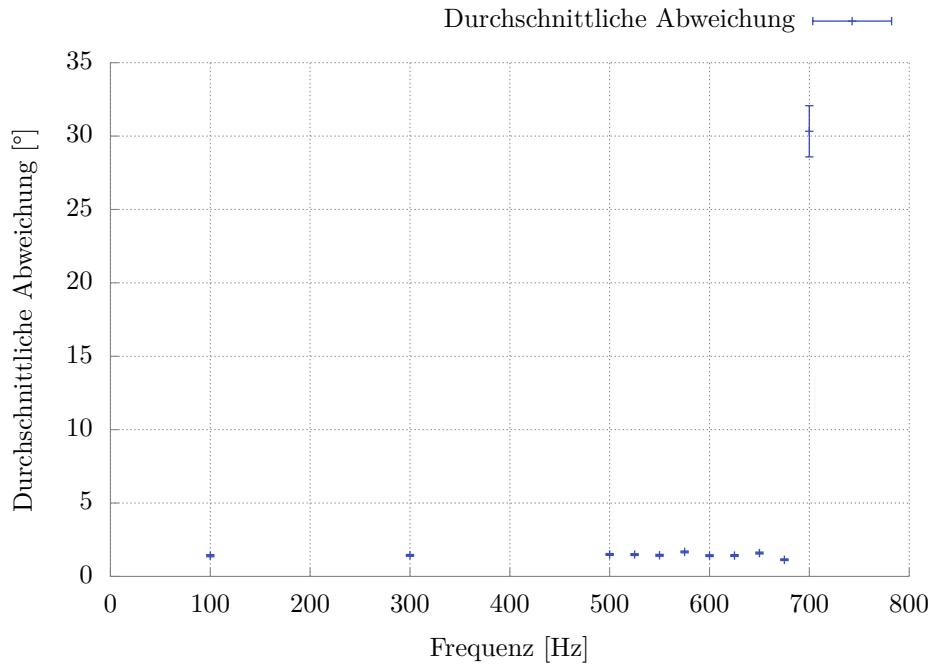


Abbildung 15: Genauigkeit für verschiedene Frequenzen

Man sieht, dass die Genauigkeit der Richtungsbestimmung (fast) unabhängig von der Frequenz der Schallquelle ist. Für Frequenzen über 675Hz ist die Bedingung, dass die Abstände der Mikrofone kleiner als die halbe Wellenlänge sein muss, nicht mehr erfüllt.

8.2 Echtwelt

In unseren Simulationen hat sich gezeigt, dass unser Verfahren sehr genau arbeitet. Um zu untersuchen, wie genau es unter dem Einfluss von Störungen und mit den Fehlern durch die Aufnahme mit Mikrofonen arbeitet, haben wir für acht verschiedene Positionen der Schallquelle die Abweichung der Richtung bestimmt. Diese Messung haben wir in einem mit speziellem Schaumstoff [7] ausgekleideten Raum durchgeführt, um Störungen wie Reflexionen, in der Testphase, zu minimieren. Der Lautsprecher war jeweils 0.75m von dem Zentrum der Mikrofone entfernt.

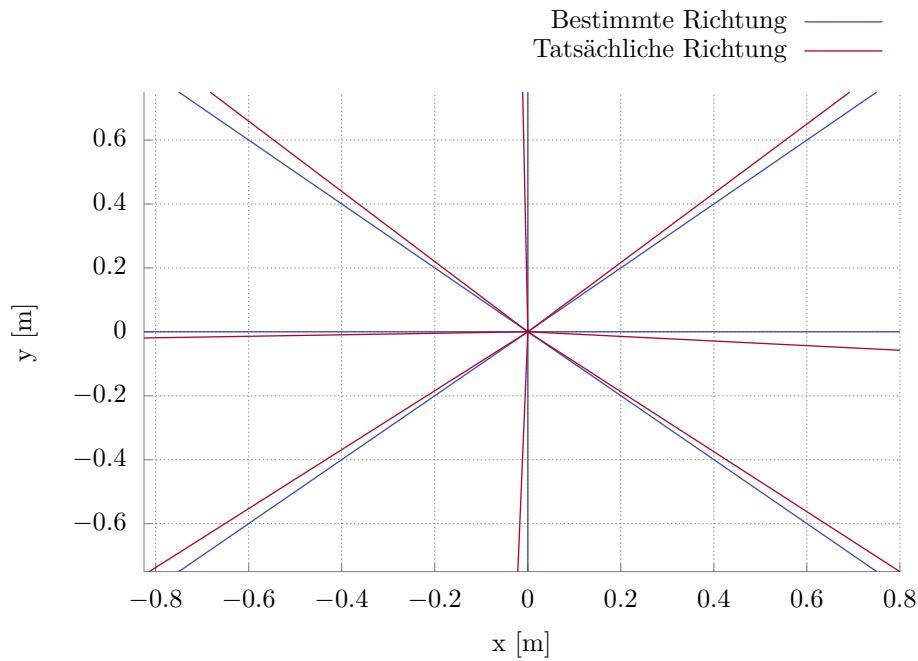


Abbildung 16: Genauigkeit in der Echtwelt



Abbildung 17: Versuchsaufbau zur Messung der Genauigkeit der Richtungsbestimmung in der Schallkammer

Für die acht verschiedenen Richtungen hatte die Richtungsbestimmung eine Genauigkeit von $3.9^\circ \pm 0.13^\circ$. Allerdings hatte der Lautsprecher bei dem von uns gewählten Abstand eine Winkelgröße von 3.2° , weshalb dies ein sehr guter Wert ist. Die Richtungsbestimmung mit unserem Verfahren ist also auch in der Echtwelt mit großen Abständen genauer als beim Menschen.

8.3 Alltagstest

Um die Verwendbarkeit unseres Verfahren auch außerhalb einer Schallkammer zu testen, haben unser System mit verschiedenen Schallquellen unter Realbedingungen getestet.

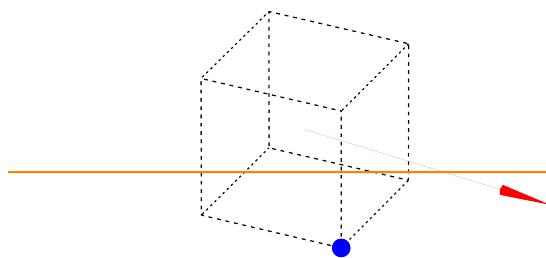
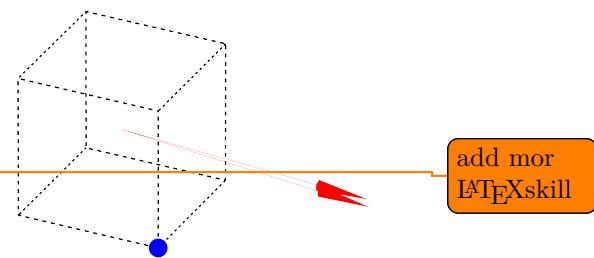


Abbildung 18: Die Richtungsbestimmung einer Si-nusschwingung



add more
LATEXskill

Abbildung 19: Die Richtungsbestimmung einer Rechteckschwingung

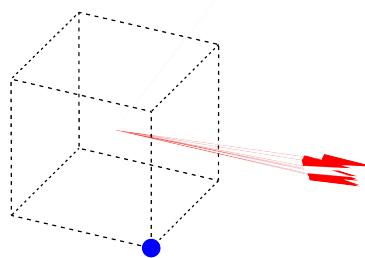
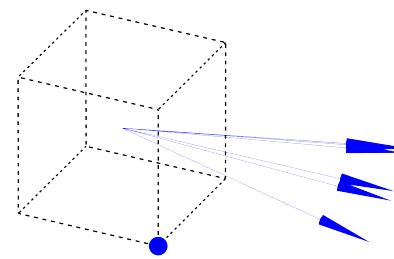


Abbildung 20: Die Richtungsbestimmung von Musik



me Abbildung 21: Die Richtungsbestimmung von Stimme

Man kann erkennen, dass die mit unserem Verfahren bestimmte Richtung sehr gut mit der tatsächlichen Richtung übereinstimmt, obwohl Störungen wie Reflexionen auftreten. Des weiteren ist zu erkennen, dass selbst bei komplizierten Signalen mit einem Breiten Frequenzspektrum, wie Musik, die Ortung gut funktioniert. Dies zeigt, dass unser Verfahren auch für Echtweltsituationen gut geeignet ist und es auch hierbei nur eine sehr geringe Streuung der Richtung gibt.

9 Ausblick

Dadurch, dass wir anstatt einer Richtung eigentlich eine Position bestimmen, könnte man unser Verfahren nach weiterer Optimierung auch für die Positionsbestimmung von Schallquellen verwenden.

10 Fazit

Insgesamt ist es uns im Rahmen dieser Jugend-Forschung Arbeit gelungen, ein neues Verfahren zur Richtungsbestimmung von Schallquellen zu entwickeln, zu optimieren und zu evaluieren, das gerade einmal vier Mikrofone benötigt, um besser als der Mensch die Richtung zu bestimmen. Die Implementation unseres Verfahrens ist modular aufgebaut und ermöglicht dadurch eine leichte Integration in bestehende Programme und weitere Verarbeitungs- und Auswertungsschritte. Hierdurch ist es möglich, das von uns entwickelte System einfache in technische Anwendungen zu integrieren und somit die Fähigkeit des Richtungshörens zu nutzen. Außerdem kann das Verfahren ohne großen Aufwand an spezielle Anforderungen angepasst und für diese optimiert werden.

rewrite;
sachen die
gehen in
Arbeit pa-
cken; Am-
plituden
komplett
raus

vll. mal
wirklich
unabhön-
gigkeit v.
schallge-
schwin-
digkeit
einbauen?

Literatur

- [1] URL: <http://nt.uni-paderborn.de/forschung/sprachsignal-verarbeitung-und-spracherkennung/geraeuschoschrobuste-spracherkennung/> (besucht am 11.01.2016).
- [2] URL: <http://www.acoustic-camera.com/de.html> (besucht am 11.01.2016).
- [3] URL: <http://www.campbell-associates.co.uk/saleacocam.htm> (besucht am 13.01.2016).
- [4] URL: <https://de.wikipedia.org/w/index.php?title=Signal-Rausch-Verh%C3%83%C2%A4ltnis&direction=prev&oldid=151031958> (besucht am 02.01.2016).
- [5] URL: http://www.epanorama.net/circuits/microphone_powering.html#phantom (besucht am 20.12.2015).
- [6] URL: <http://www.portaudio.com/> (besucht am 03.01.2016).
- [7] URL: http://www.plasticsportal.net/wa/plasticsEU~EN_GB/portal/show/content/products/foams/basotect (besucht am 16.01.2016).
- [8] E. Anderson u. a. „LAPACK: A Portable Linear Algebra Library for High-performance Computers“. In: *Proceedings of the 1990 ACM/IEEE Conference on Supercomputing*. Supercomputing '90. New York, New York, USA: IEEE Computer Society Press, 1990, S. 2–11. ISBN: 0897914120. URL: <http://dl.acm.org/citation.cfm?id=110382.110385>.
- [9] I. Fette und A. Melnikov. *The WebSocket Protocol*. RFC 6455. RFC Editor, Dez. 2011. URL: <http://www.rfc-editor.org/rfc/rfc6455.txt>.
- [10] M. Frigo und S.G. Johnson. „The Design and Implementation of FFTW3“. In: *Proceedings of the IEEE* 93.2 (Feb. 2005), S. 216–231. DOI: 10.1109/jproc.2004.840301. URL: <http://dx.doi.org/10.1109/JPROC.2004.840301>.
- [11] Garin Hiebert. *OpenAL 1.1 Specification and Reference*. <https://www.openal.org/documentation/openal-1.1-specification.pdf>. 2005.
- [12] John C Middlebrooks und David M Green. „Sound localization by human listeners“. In: *Annual review of psychology* 42.1 (1991), S. 135–159.
- [13] Jon Postel. *Transmission Control Protocol*. STD 7. RFC Editor, Sep. 1981. URL: <http://www.rfc-editor.org/rfc/rfc793.txt>.
- [14] Inc. Wolfram Research. *Mathematica*. 2012.
- [15] Thomas Williams, Colin Kelley und many others. *Gnuplot 4.4: an interactive plotting program*. <http://gnuplot.sourceforge.net/>. März 2010.