
Entwicklung von Methoden und Konzepten für ein spielerisches Trainingsprogramm zur Gehörbildung

Bachelor-Thesis
Lukas Rohde
KOM-type-number



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Elektrotechnik
und Informationstechnik
Fachbereich Informatik (Zweitmitglied)

Fachgebiet Multimedia Kommunikation
Prof. Dr.-Ing. Ralf Steinmetz

Entwicklung von Methoden und Konzepten für ein spielerisches Trainingsprogramm zur Gehörbildung

Development of methods and concepts for a game-based ear training program

Bachelor-Thesis

Studiengang: Informatik

KOM-type-number

Eingereicht von Lukas Rohde

Tag der Einreichung: 15. Juli 2021

Gutachter: Prof. Dr.-Ing. Ralf Steinmetz

Betreuer: Dr.-Ing. Stefan Göbel

Technische Universität Darmstadt

Fachbereich Elektrotechnik und Informationstechnik

Fachbereich Informatik (Zweitmitglied)

Fachgebiet Multimedia Kommunikation (KOM)

Prof. Dr.-Ing. Ralf Steinmetz

Erklärung zur Abschlussarbeit gemäß § 22 Abs. 7 und § 23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Lukas Rohde, die vorliegende Bachelor-Thesis gemäß § 22 Abs. 7 APB der TU Darmstadt ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§38 Abs.2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Thesis stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäß § 23 Abs. 7 APB überein.

Darmstadt, den 15. Juli 2021

Lukas Rohde



Inhaltsverzeichnis

1	Einleitung	3
2	Grundlagen	5
2.1	Signalverarbeitung	5
2.2	Musikalische Grundlagen	5
2.2.1	Intervalle	5
2.2.2	Die Grundfrequenz und Obertöne	5
2.2.3	Tonsysteme und Stimmungsarten	6
2.3	Datenverarbeitung	6
3	Analyse	7
3.1	Analyse von Trainingsprogrammen zur Gehörbildung	7
3.1.1	State of the Art	7
3.1.2	Analyse der vorhandenen Features in State of the Art(?)	9
3.2	Analyse von Methoden u. Konzepten zur echtzeit Erkennung und Verarbeitung von Audio Input vom Anwender mit Unity	9
3.2.1	Recherche	10
3.2.2	Analyse der Ansätze zur echtzeit Audio Verarbeitung	11
4	Konzeption eines spielerischen Trainingsprogramms zur Gehörbildung	13
4.1	Aufbau für so ein Programm	13
4.2	Systemsicht: I/O Verarbeitung, Gui wie wird interagiert	13
4.3	Evaluationskonzept	13
5	Prototypische Realisierung in Unity3D (nicht zu ausführlich schreiben)	15
6	Evaluation / Validierung der erarbeiteten Methoden und Konzepte	17
7	Zusammenfassung	19
	Literaturverzeichnis	19



Zusammenfassung

Test



1 Einleitung

- Thema - Warum relevant? Wie ist es einzuordnen? (Singen macht Spaß -> Problem: Viele haben keine ausgeprägte Gehörbildung, Training für Lernende) - Motivation: Viele haben Schwierigkeit Gehörbildung anständig zu lernen (fehlende Programme, schlechte Handhabung, festes Lernkit) - Allg. Fragestellung: "Viele Menschen haben Probleme..., Wollen Lernen..., Müssen im Rahmen ihres Studiums lernen... techn. Problemstellung: "Realtime Soundverarbeitung unter UNity mit angemessener Fehlerquote / Intuitives Design der Notenausgabe/Platzierung auf das Notensystem"



2 Grundlagen

2.1 Signalverarbeitung

- Frequenz - Abtasttheorem - FFT - Unity eingebauten Methoden

2.2 Musikalische Grundlagen

2.2.1 Intervalle

Ein Intervall im musikalischen Sinn bezeichnet den Tonabstand zwischen zwei Tönen. Man kann dabei zwischen den sogenannten harmonischen Intervallen und den melodischen Intervallen unterscheiden. Bei den harmonischen Intervallen erklingen die Töne gleichzeitig, wohingegen bei melodischen Intervallen die Töne hintereinander erklingen. Intervalle lassen sich mathematisch als Proportionen der erklingenden Töne zueinander beschreiben, das wichtigste Intervall ist hierbei die Oktave. Die Oktave hat ein Frequenzverhältnis von 2:1, sie entspricht also immer der doppelten Frequenz von einem Ton zu dem anderen und kann von mehreren *Tonsystemen* unterteilt werden. Alle dieser Unterteilungen benennen die Schritte innerhalb einer Oktave nach den lateinischen Ordinalzahlen, wobei Oktave Acht bedeutet. Wie der Name Oktave also schon vermuten lässt, liegen innerhalb einer Oktave sieben andere Intervalle. Diese Annahme ist allerdings nur bedingt richtig, da neben den sieben nach den Ordinalzahlen benannten Intervallen, auch große und kleine, sowie verminderte und übermäßige Intervalle existieren. Diese weitere Einstufung der Intervalle ermöglicht es neben Intervalle mit einem halben Ton abstand zu notieren, neben den Ganztönen. Berücksichtigt man also alle diese Tatsachen, so lässt sich eine Oktave in 12 Halbtöne aufteilen, wobei jeder mögliche Abstand zwischen diesen Halbtönen benannt werden kann.

Es gibt neben den Einteilungen von Intervallen innerhalb einer Oktave auch bezeichnungen für Intervalle, welche einen Abstand größer als eine Oktave umspannen. Ich möchte diese aufgrund der Vollständigkeit auch erwähnen, jedoch werden diese nicht weiter in der folgenden Arbeit berücksichtigt. Diese Intervalle folgen weiterhin der lateinischen Ordinalzahlreihe, sodass ein Intervall, welches einen Ganzton weiter umspannt als die Oktave, dementsprechend None genannt wird.

2.2.2 Die Grundfrequenz und Obertöne

Bei allen musikalisch erzeugten Tönen treten neben dem sogenannten Grundton bzw. der Grundfrequenz auch Nebenschwingungen oder Obertöne mit auf. Bei der Grundfrequenz handelt es sich um die tiefste auftretende Schwingung in diesem Spektrum, die restlichen messbaren Schwingungen sind die sogenannten Obertöne, diese bestimmen vorallem die Klangfarbe, verändern aber nicht den erklingenden Ton in seiner Höhe. In der Musik bestimmt also die Grundfrequenz den wahrgenommenen erklingenden Ton und die Obertöne nur die Wahrnehmung des Tons. Die Obertöne können dabei aber häufig einen stärkeren Ausschlag verursachen, untersucht man einen Ton mithilfe einer Fouriertransformation. Ein Ton im musikalischen Sinn beschreibt also nicht eine reine Sinuswelle, sondern vielmehr eine Überlagerung mehrerer Schwingungen, welche zusammen den Ton mit Klangfarbe ausmachen. Wichtig zu wissen, ist dass die Obertöne fast immer ganzzahlige vielfache des Grundtons sind [Sen01], Töne mit dieser Eigenschaft nennt man Harmonisch. Ausnahmen zu dieser Regel sind vorallem Klänge welche für das menschliche Ohr als unschön oder störend empfunden werden, wie beispielsweise Glocken oder fallende Rohre und Stangen, sie werden auch unharmonisch genannt.

2.2.3 Tonsysteme und Stimmungsarten

Wie bereits in *Intervalle* angesprochen, gibt es verschiedene Systeme um die Tonabstände zu bestimmen, ich möchte hier nun die wichtigsten dieser Arten beleuchten und auf ihre Anwendungen und Herkünfte eingehen. Verschiedene Arten von Unterteilungen werden auch Stimmungen genannt. Betrachten wir zunächst den Tonraum, in welchem wir uns befinden.

In dem geordneten Tonraum lässt sich jedem Ton genau eine Frequenz zuordnen, wobei gilt, umso höher die Frequenz, desto höher ist auch der erklingende Ton. Außerdem ist jedem Paar von Tönen t_1, t_2 bzw. Frequenzen f_1, f_2 genau ein Intervall i_{12} zugeordnet, wir können daraus folgern, dass das Intervall der Töne, ein Frequenzverhältnis von $f_2 : f_1$ hat. Es wird außerdem in der Musik die Einheit Cent herangezogen um genauere Angaben zu verschiedenen hohen Tönen zu machen. Ein Cent ist definiert als $1/1200$ Oktave. Betrachtet man den Raum der Intervalle und der Frequenzverhältnisse, so fällt auf, dass ein Homomorphismus zwischen diesen beiden Räumen existiert. Addiert man zwei Intervalle im Intervallraum, so multipliziert man deren Frequenzverhältnisse bzw. Proportionen.

Ein kurzes Beispiel soll diesen Zusammenhang verdeutlichen. Nehmen wir die kleine Terz $i_1 = 316\text{Cent}$ und die Quinte $i_2 = 701\text{Cent}$ und ihre dazugehörigen Frequenzverhältnisse $p_1 = 6/5$ und $p_2 = 3/2$ als gegeben an.

$$i = i_1 + i_2 = 1017(\text{kleineSeptime})$$

$$p = p_1 * p_2 = 9/5(\text{kleineSeptime})$$

$$\Rightarrow 1200 * \log_2 9/5 \approx 1017$$

- Cent - Reine Stimmung (stimmungsarten)
- diatonik - tonreihen (mannheim website) - Reichweite der mensch. Stimme

2.3 Datenverarbeitung

- Ringbuffer

3 Analyse

- Wie in der Recherche vorgegangen (Kontakte, Anhaltspunkte, Wissenschaftler, Suchbegriffe, wichtigste Filter) -> Top Referenzen - Studentin befragt nach bestem SoTA - Was fehlt, was könnte besser sein?
- Hobby Musiker befragt ob sie etwas kennen und Interesse hätten, wie wäre es Interessant? - Gehörbildungslehrer / Musiklehrer befragen, woran es häufig hakt und was interessant ist in so einem Programm
 - tech. Knackpunkt: Wie macht die Game Engine etwas? (Sound Verarbeitung etc.) - Auslesen der Mikrophon Daten und direkte Umrechnung mithilfe von gegebenen FFT Methoden in Unity - PDA im Rahmen dieser Möglichkeiten - Warum keinen komplexeren PDA implementiert ohne eingebaute Unity FFT

3.1 Analyse von Trainingsprogrammen zur Gehörbildung

- auf bestimmte Aspekte achten, bei der Analyse - Audio Input? - Arten von Übungen - Vielfalt - Abdeckung der musikalischen Bereiche (Intervalle, Akkorde,) - Übungsvielfalt - festes Set oder Endlosübungen - Navigierbarkeit

3.1.1 State of the Art

Zu den State of the Art Programmen gehören vor allem online Tools zur Unterstützung bei der Gehörbildung. Zur Recherche nach State of the Art Programmen habe ich zunächst eine einfache Suchanfrage bei Google getätigt, um so die gängigsten Programme und Tools zu finden zu diesen zählen:

Gehörbildungswebsite der staatlichen Hochschule für Musik und darstellende Kunst Mannheim

Hierbei handelt es sich um ein online Tool, bei welchem aus einer festen Anzahl an aufgenommen Hörbeispielen Übungen generiert werden. Dabei werden die Themen Intervall- und Akkord-Intonation, sowie eine Möglichkeit zur Überprüfung der eigenen Fähigkeiten angeboten. Das Intervalllernen ist so strukturiert, dass der Nutzer Töne vorgegeben bekommt und die Intervalle zwischen den Tönen bzw. zu einem Grundton bestimmen soll. Die Website überprüft dabei nicht selbst, ob der Nutzer die Aufgabe richtig absolviert hat, der Nutzer muss sich selbst überprüfen. Es gibt mehrere Arten von Intervalltraining, so kann man seine Fähigkeiten in diatonischen Stufen, also innerhalb einer Tonleiter ohne Halbtonen, oder reines Intervalle erkennen zwischen einzelnen Tönen. Es gibt weiterhin die Möglichkeit in ganzen Tonreihen direkte und indirekt erklingende Intervalle zu erhören bzw. wahrzunehmen. Außerdem bietet die Website vorgefertigte Übungen zu Rhythmus, Melodie, Akkorden, Harmonik und Intonation an. Man erhält Zugriff auf die Übungen, nachdem man einen Account erstellt hat. Die Übungen sind dann unterteilt in einzelne Reiter in die einzelnen Lektionen.

Die Website bietet allerdings nur die Aufgaben an, das heißt es gibt keine eingebaute Überprüfung, weder als Texteingabe noch als Audioinput vom Nutzer. [Man09]

Musictheory.net und Tenuto

Die Website Musictheory bietet eine Vielzahl von Übungen in verschiedenen Disziplinen an. Es existieren theoretische Lektionen, welche das fundamentale Wissen herstellen sollen, sowie Übungen, um dieses abzufragen und zu testen. Die Lektionen umfassen die Grundlagen der Musik, wie etwa die verschiedenen Notenschlüssel oder das Notenlesen, bis hin zu verschiedenen Arten von Akkorden und wie diese

aufgebaut werden. Die Übungen umfassen dementsprechend einen ähnlichen Wissensumfang, es wird beispielsweise das Notenlesen abgefragt oder die Tonart, aber man kann auch Gehörbildung üben, wo man das Erkennen von Tönen bis hin zu Akkorden üben kann. Bei diesen Übungen kann man allerdings nur das Gehörte aus einer Liste auswählen. Ein Unterschied zu der Website der HfMDK Mannheim besteht darin, dass Musictheory.net Antwortmöglichkeiten gibt und bei diesen ein Feedback gibt, ob die Antwort richtig war. Auch hier gibt es kein Feature, welches gesungene Intervalle überprüft. Musictheory.net hat eine App für iOS entwickelt, welche die Aufgaben, welche auch auf der Website zur Verfügung stehen, mit erweiterten Funktionen bereitstellt. Die Funktionen der App umfassen zusätzlich zu den Aufgaben der Website eine Möglichkeit Aufgabentypen anzupassen, umso Aufgabentypen spezifisch zu lernen. Desweiteren wurde ein Challenge Modus in der App hinzugefügt, welcher den Nutzer herausfordert unter Zeitdruck Aufgaben richtig zu beantworten und den Highscore zu verbessern. Das User Interface der App wird extra beworben auf der Website als einfach zu bedienen und mit einer klaren Benutzer Erfahrung. Die App kostet 3.99\$, wohingegen die Website kostenfrei ist und ohne Benutzeraccount verwendet werden kann. Es ist noch anzumerken, dass sowohl die Website, als auch die App nur auf Englisch verfügbar sind, was eine Behinderung für den Nutzer darstellen kann. [mus09]

Teoria

Teoria bietet, ähnlich wie Musictheory, einen Theorieteil und einen Praxisteil an. Im Theorieteil werden auch hier die Grundlagen der Musik vermittelt, wie beispielsweise das Notenlesen oder was ein Intervall ist und wie diese aufgebaut werden. Im praktischen Teil ermöglicht Teoria es dem Nutzer seine eigenen Aufgaben selbst zu gestalten, indem man die abzufragenden Intervalle einstellen kann oder auch den Grundton. Es ist außerdem möglich das sogenannte vom Blatt Singen zu trainieren, was nichts anderes heißt als direkt die Noten und den Rhythmus richtig zu singen, ohne es vorher gehört oder geübt zu haben. Das Singen wird allerdings auch hier nicht überprüft, es ist nur ein Angebot gegeben sich selbst überprüfen zu können. Es werden weiterhin Übungsangebote in den Aufgabenbereichen der Intervalle, Akkorde, des Rhythmus und der Tonarterkennung. Die Aufgaben werden durch den Nutzer voreingestellte Parameter zufällig generiert und können somit auch an die Bedürfnisse des Nutzer angepasst werden. Die Website hat ein relativ einfaches und intuitives Design, ist allerdings nur auf Englisch verfügbar, was beispielsweise die Namen der Akkorde betrifft und durchaus eine Umstellung für den Nutzer darstellen kann. [Alv09]

JKG Neigungskurs Musik

Die Website des Justus-Knecht-Gymnasium Bruchsaal bietet ein kleines vordefiniertes Set an Aufgaben, für die Vorbereitung auf die Musik Abiturprüfung, an. Zu den Aufgabentypen zählen Rhythmusdiktate, Melodiediktate, sowie Intervalle und Akkorde erkennen. Wie bereits erwähnt sind die Aufgaben vordefiniert und bieten keinerlei Möglichkeit sie anzupassen nach den Bedürfnissen des Nutzers. Die Aufgaben bilden eine grobe Übersicht über die möglichen Aufgabentypen und dienen nur als Lernhilfe oder zur Überprüfung, sind jedoch nicht geeignet als alleiniges Lernmittel oder zur täglichen Festigung der Fähigkeiten. Desweiteren ist die Bedienbarkeit und Nutzerfreundlichkeit auf der schwächeren Seite, da die Soundbeispiele über Soundcloud zwar in die Website eingebunden sind, jedoch die Lösungen in PDFs stehen, welche zunächst heruntergeladen werden müssen. [Bru21]

Musikgrad

Musikgrad einige simple Aufgabentypen und Trainingsmethoden zu Intervallbestimmung, aber auch zu der Bestimmung von Akkorden und den Grundlagen der Musik, wie etwa das Notenlesen. Da einige Aufgaben allerdings leider Flash Player benötigen, sind diese nicht mehr zugänglich. Ich beschränke mich im

Folgenden nur auf die verfügbaren Aufgaben, welche allerdings immernoch die wichtigsten Funktionen abdecken. Es wird die Möglichkeit geboten Intervalle zu erhören, dabei werden zwei Töne gleichzeitig abgespielt und der Nutzer muss aus allen Intervallen entscheiden, welches erklingen ist. Es wird dabei die Möglichkeit gegeben, die abzufragenden Intervalle zu konfigurieren, umso auf eigene Schwächen genauer einzugehen. Es gab offensichtlich mal die Möglichkeit auch zu einem gegebenen Grundton ein Intervall zu vervollständigen in einem Notensystem, diese Funktion ist allerdings mittlerweile veraltet und nicht mehr Verfügbar. Auch bei Musikgrad gibt es keine Möglichkeit, dass der Nutzer singen kann und dieser Gesang automatisch überprüft und angezeigt wird. Die Website an sich ist simpel aufgebaut, sodass man durch eine einfache Auswahl zu den verschiedenen Übungen kommt. Die Übungen sind verfügbar innerhalb einer eingebundenen Anwendung im Browser, sodass kein Vollbildmodus verfügbar ist. [Rie]

Earmaster

Earmaster ist eine professionell Entwickelte Gehörbildungssoftware, welche von dem gleichnamigen Unternehmen entwickelt wird. Earmaster existiert seit den 1990er Jahren und wird seit dem immer weiter entwickelt und deckt somit einen Großteil der Bedürfnisse der Nutzer ab. Die Software kann online für 4\$ im Monat gekauft werden und bietet neben Kapitelweise aufgebauten Modulen auch die Möglichkeit benutzerdefinierte Übungen zu generieren. Die Funktionen umfassen eine modulweise Einführung in die Gehörbildung, wobei Themenbereiche von der Tonhöhe bis hin zu Akkord Fortschreitungen abgedeckt werden. Weiterhin bietet Earmaster neben diesem sogenannten Einsteigerkurs auch Workshops, welche das erlangte Wissen aus dem Einsteigerkurs weiter festigen sollen. Diese Workshops gehen dabei auch Modulweise vor und fragen dabei die gewünschten Inhalte gezielter ab. Betrachtet man hierbei beispielsweise den Intervall singen Workshop, so werden dort einzelne Herangehensweisen des Intervallsingens behandelt und abgefragt. Die Software bietet für ihre Zahlreichen verschiedenen Aufgabentypen eine Vielfalt an Arten der Eingabe der Lösungen. Es ist möglich Intervalle selbst zu singen, Töne einzeln auf das Notensystem zu platzieren, sowie aus einer Auswahl an Antworten zu wählen. Die Tonerkennung ist konfigurierbar, es werden dem Nutzer mehrere Tonlagen zur Auswahl gestellt, sowie die Möglichkeit gegeben seine eigene Tonlage zu definieren. Weiterhin werden nicht nur die tatsächlichen Tonhöhen als richtige Antwort gewertet, sondern auch die jeweiligen Oktaven des gesungenen Tons, sodass der Nutzer immer in der angenehmsten Tonlage singen kann. Außerdem gibt es einige extra Jazz Workshops, welche die besonderen Eigenschaften der Akkorde und Akkordfolgen der Jazzmusik behandeln. Der Nutzer kann außerdem seine bereits vollendeten Lektionen in einer Statistik einsehen, wobei dort auch der Gesamtfortschritt der Übungen eingesehen werden kann. Bei dem Absolvieren der Übungen selbst bekommt der Nutzer direktes Feedback in Form von 5 möglichen zu erreichenden Sternen, welche je nach Genauigkeit und Intonation des Tons vergeben werden. Weitere Gamification Elemente, wie etwa eine das Aufzeichnen einer Übungsstreak, welche angibt wie viele Tage man täglich geübt hat, gibt es nicht.

3.1.2 Analyse der vorhandenen Features in State of the Art(?)

- Auf Pitch Detection achten - Übungsarten - Einführung in Thema - Wieder(spielbarkeit)

3.2 Analyse von Methoden u. Konzepten zur echtzeit Erkennung und Verarbeitung von Audio Input vom Anwender mit Unity

- Echtzeit Audio einlesen - Auf Audio zugreifen (GetSpectrumData, GetData, mehrere Ansätze -> Ringbuffer, direkte Verarbeitung) - PDA - Pitch Detection Algorithm (Welche Methoden gibt es warum für was entschieden) - Genauigkeit verbessern (Noise Filter, Threshold, Interpolation) - Noise Filter welche ansätze gibt es, warum für einfache Avg Lösung entschieden - Threshold für Ausschläge geben - Interpolation warum Newton und warum nur 3 Punkte

3.2.1 Recherche

Ich bin bei der Recherche schrittweise vorgegangen und habe zunächst generell nach der Tonverarbeitung in Unity recherchiert. Die erste Quelle für diese Informationen ist in der Regel die Dokumentation, dementsprechend habe ich mir die Unity Dokumentation angesehen und nach allem gesucht was Audio, Sound oder Frequency erwähnt. Dabei bin ich zunächst auf die allgemeine Informationen Seite von Unity zu Audio gestoßen [Tec09], diese leitete mich weiter zu detaillierteren Artikeln zu Audio Verarbeitung in Unity. Zu diesen Artikeln gehörten Audio Source, Audio Listener, Audio Mixer, Audio Effects und Reverb Zones, wobei sich die Reverb Zone nach kürzerer Recherche als irrelevant herausstellte. Die restlichen Artikel waren hingegen sehr von Nutzen für mein weiteres Vorgehen. Die wichtigste Information habe ich aus der Skripting Dokumentation der Audio Source erhalten. Die Audio Source besitzt in der Skripting API eine statische Methode, welche es mir ermöglicht auf die Spektrum Daten des dazugehörigen AudioClips zuzugreifen. Bevor ich mich allerdings mit Spektrum Daten befassen konnte, musste ich erst herausfinden, wie ich Audio Daten vom Nutzer einlese, mein nächster Schritt war also nach einer Mikrofon Klasse in der Unity Dokumentation zu suchen. In der Unity Dokumentation habe ich die Microphone Klasse gefunden, über diese fand ich heraus, dass es möglich ist die Position der aktuell ausgelesenen Daten zu erhalten. Mithilfe der Position der Daten liese sich also eine Art Ringbuffer implementieren, indem man die aktuelle Position des Mikrofons als Pointer auf einen Ringbuffer zeigen liese. Mein nächster Gedanke war daraufhin, einen eigenen Ringbuffer zu implementieren. Ich habe mir also Gedanken gemacht, welche weiteren Funktionen ich benötigen würde, wenn ich diesen Ansatz weiter verfolgen würde. Da ich über diesen Ansatz nur die Audiorohdaten speichern könnte, müsste ich außerdem eine geeignete Schnelle Fourier Transformation implementieren bzw. finden. Ich stellte diesen Ansatz erstmal hinten an, um zunächst Ansätze zu erforschen, welche näher an der Unity Engine liegen und Diese mehr ausnutzen. Ich recherchierte also weiter in der Unity Dokumentation und fand nach kurzer Zeit über das Schlagwort Audio Source die Funktion GetSpectrumData. Die Funktion ermöglicht es direkt die Audiodaten einer AudioSource mithilfe einer schnellen Fouriertransformation zu erhalten. Ich habe mich nun also mehr mit den mir zur Verfügung stehenden schnellen Fouriertransformationen von Unity befasst. Die möglichen schnellen Fouriertransformationen, welche Unity zur Verfügung stellt, sind folgende: Rectangular, Triangle, Hamming, Hanning, Blackman und BlackmanHarris. Ich gehe genauer auf die Fouriertransformationen und die Arten der Verarbeitung in einer späteren Sektion ein.

Ich testete nun mit der Fouriertransformation einige Ansätze aus, ob und wie gut sich aus einer einfachen Fouriertransformation der erklingende Ton herausfiltern lässt. Ein naiver Ansatz war zunächst das absolute Maximum der Fouriertransformation zu wählen und diesen als erkannten Ton weiter zu verarbeiten. Dieser Ansatz funktionierte zunächst sehr gut mit Sinuswellen förmigen Tönen. Als ich anfang die Erkennung mit meiner eigenen Stimme zu testen, wurde mir jedoch schnell klar, dass die *Obertöne* der menschlichen Stimme einen komplexeren Ansatz der Tonerkennung benötigen würde. Ich setzte mich als nächstes also mit einigen Pitch Detection Algorithmen bzw. Ansätzen auseinander.

Außerdem wusste ich bereits, dass es in den Projekteinstellungen zu einem Unity Projekt einige Audio Einstellungen gibt und habe mir daher auch diese genauer angesehen und wie sich diese auf das Projekt auswirken.

Als alle Unity Referenzen zu einem Großteil abgedeckt waren, weitete ich die Suche aus nach allgemeineren Informationen zu Pitch Detection Algorithmen (PDA). Dabei wurde schnell deutlich, dass es viele Ansätze gibt um die erklingende Note heraus zu filtern, sodass ich mich entschied einen der performatesten PDAs zu implementieren. Es blieb nur noch die Frage wie sich dieser am Besten in Unity integrieren lassen würde.

Analyse der Fouriertransformationen

Analyse der Pitch Detection Algorithmen

3.2.2 Analyse der Ansätze zur echtzeit Audio Verarbeitung

Wie bereits in der Recherche erwähnt gab es mehrere Ansätze, um echtzeit Audio Daten vom Nutzer zu lesen. Ich möchte in dem folgenden Abschnitt meine Handlungsweise und Entscheidung erläutern. Die Ansätze umfassten einen eigenen Ringbuffer zu programmieren und mithilfe von diesem die Daten sobald sie ankommen weiterverarbeiten zu können, dieser Ansatz hätte außerdem eine eigene Implementation einer Schnellen Fourier Transformation benötigt. Ein weiterer Ansatz wäre gewesen das Verhalten von GetData als Ringbuffer zu verwenden, jedoch nur für neu angekommene Daten.



4 Konzeption eines spielerischen Trainingsprogramms zur Gehörbildung

- Grundkonzepte (geleitet / einföhrung in das Thema) und erweiterbare Module (level Basiert) - Tutorial / Lernkapitel - Endlosübungen zur Festigung - Geföhrte Einföhrung in Benutzung? - Flow
 - User Centered Design -> Kontakte: Befragung was wichtig ist (UI, UX) - Studierende befragt (Eva, Pia, Kommilitonen Eva)
 - Basierend auf den Erkenntnissen aus Kapitel 2 -> Aus der Analyse muss klar werden, was wichtig ist

4.1 Aufbau für so ein Programm

- Design Pattern?

4.2 Systemsicht: I/O Verarbeitung, Gui wie wird interagiert

4.3 Evaluationskonzept

- Hobby, Profi, Studierende - Unterschiedliche Fragen für jede Gruppe? oder Unterschiedliche Wahrnehmung der Gruppen? - Hobby - Dazu gelernt? Hilfreich? Richtige Kompetenz oder zu schwer? Weiter benutzen? - Profi - Soetwas gewünscht in der Ausbildung? Weiterhin praktisch zum fit bleiben? Empfehlung an Kollegen? - Studierende - Richtige Ansätze für Uni? Hilfreich beim Lernen? Alle wichtigen Features dabei? -allgemeine - Wie war die Erkennung der Stimme - Nutzbarkeit der Anwendungen



5 Prototypische Realisierung in Unity3D (nicht zu ausführlich schreiben)

- Töne u. Intervalle vorspielen; Erkennung / Einordnung von Tönen - Intervalle ergänzen, selbst singen u. dessen Verarbeitung - Adäquate GUI mit Feedback - Wie wurde das eingebundene realisiert - Warum Unity und kein andere Engine - Typisches Ergebnis ist UML Diagram - Laufzeitanalyse (wegen echtzeit aspekt)



6 Evaluation / Validierung der erarbeiteten Methoden und Konzepte

- Mehrstufige Validierung mit Musikverein, Studenten, ... - Unterschiedliche Validierungen vergleichen - Konzept in 3 Stufen – Vorbereitung, Durchführung, Nachbereitung -> Wie gemacht: Was will ich messen (ERkennungsrate, UX Bewertung)



7 Zusammenfassung

- Zusammenfassung - Ausblick – Offene Fragen? (Real time Erkennung, ...) — Umsetzung für Chor?
- Machine Learning zur Unterscheidung? AI Dirigent?
- Literaturverzeichnis (APA)



Literaturverzeichnis

- [Alv09] José Rodríguez Alvira. Teoria. Online, 2021-02-09.
- [Bru21] Justus-Knecht-Gymnasium Bruchsal. Jkg bruchsaal musik. Online, 2016-12-21.
- [Man09] Musikhochschule Mannheim. Gehörbildung online. Online, 2021-02-09.
- [mus09] musictheory.net. musictheory. Online, 2021-02-09.
- [Rie] Walter Riedinger. Musikgrad. Online.
- [Sen01] UdK Berlin Sengpiel. Harmonische, partialtöne, teiltöne und obertöne. 2002-11-01.
- [Tec09] 2020 Unity Technologies. Unity3d dokumentation audio overview. Online, 2021-02-09.