Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

InIT Institut für angewandte
Informationstechnologie

Projektarbeit (Informatik)

Automatische Erkennung der Akkordfolge in Popmusik

Autoren	Fabian Helg Fiona Waser
Hauptbetreuung	Thilo Stadelmann
Nebenbetreuung	Sigisbert Wyrsch
Interne Partner	Zentrum für Signalverarbeitung und Nachrichtentechnik (ZSN)
Datum	17.12.2015

Zusammenfassung

Für Musiker ist es interessant, Audiodateien in eine Software laden zu können und anschliessend die Akkordfolge des Musikstücks zu erhalten. So können die Musiker das Musikstück ohne grossen Aufwand nachspielen. Die Implementation einer solchen Software bedingt einer automatischen Akkorderkennung. In diesem Bereich gibt es durchaus noch Forschungsbedarf. Diese Arbeit legt den Grundstein, für die Implementation einer Akkorderkennungsapplikation.
Die Fachbereiche Musiktheorie, Mathematik, digitale Signalverarbeitung und Music Information Retrieval (MIR) sind für diese Forschung konstitutiv. Aufgrund von Referenzdaten, die bereits annotierte Akkordlabels enthalten, werden Akkorderkennungsmethoden evaluiert. Dazu wird ein System entwickelt, welches Akkorde aufnehmen kann und diese aufgrund von Referenzdaten, auf ihre Exaktheit prüft. Anschliessend werden Methoden evaluiert, um Empfehlungen geben zu können, wie eine solche Software implementiert werden kann. Die Evaluation erfolgt mithilfe prototypisch implementierter Methoden.
Es wurde ein Ansatz mithilfe von Templates und MIDI-Repräsentation umgesetzt. Dieser ist mit elektrisch selbstgenerierten Musikstücken sehr treffsicher, jedoch bei realen Popmusik, wie der Musik von Queen, gar nicht zufriedenstellend.
Recherchen haben ergeben, dass aktuelle Algorithmen neben der reinen Zuordnung von Frequenzen zu Akkorden auch zusätzliche Methoden, wie zum Beispiel Takterkennung in ihre Berechnungen einfliessen lassen. Dies bringt weitaus bessere Resultate. Eine weitere Möglichkeit ist der Ansatz des maschinellen Lernens. Das System wird mithilfe von Trainigsdaten geschult, um dann bei neuen Daten selbst Vorhersagen treffen zu können. Auch Hidden Markov Modelle (HMM) können zusammen mit anderen Methoden einen guten Ansatz darstellen. Auch der Ansatz der Zeit-Frequenz-Neuzuordnung verlangt noch weitere Recherche. Hinsichtlich der Problemstellung wurde ein Fortschritt gemacht. Es braucht jedoch noch weitere Recherche, um eine gut funktionierende Akkorderkennungs-Software zu implementieren.

Abstract

It is interesting for musicians to have a software, which analyses audiofiles and returns the chords. That way, the musicians are able to easily reenact the composition. The implementation of such a software requires a automated solution for chord detection. There is potential for further research in this sector. This thesis sets the foundation for the implementation of a chord detection application.
The following scientific fields are involved in this matter: mathematics, musictheory, digital signal processing and music information retrieval (MIR). Procedures will be evaluated with reference data including in advance annotated chord labels. For that purpose, a system is developed, which collects the generated and reference chords and compares them for matches. Afterwards the methods will be evaluated to give suggestions on how to implement such a software. The evaluation is done by prototypical method implementation.
An approach with MIDI-representation and templates was implemented. It is very precise with self made electrical audio files but works reall bad with real world popmusic such as from the band Queen.
Research has shown, that the latest algorithms are using additional method to enhance the process. For example using beat detection. This apporach yields much better results. A different approach is to use machine learning, in this case supervised learning, to make better predictions. Another approach could be the Hidden Markov Models (HMM) in combination with another method. Also the approach of time-frequency reassignment needs further research. There was certainly some progress regarding the problem, but still there is even more research needed to develop a well functioning Chord detection software.

Vorwort

Eine unserer grössten Leidenschaften ist die Musik. Wir als Musiker sind schon lange daran interessiert, ob Akkorderkennung automatisiert werden kann, damit aus Audiodateien Akkorde möglichst akkurat erkannt werden können. Als Informatiker wollten wir eine Projektarbeit bearbeiten, welche gleich mehrere unserer Leidenschaften verbindet. Wir beabsichtigten herausfinden, wie die Akkorderkennung von Popmusik funktioniert um für uns selbst dieses Wissen zu erlangen und später möglicherweise Applikationen zu programmieren, um automatisierte Akkorderkennung für die breite Masse zugänglich zu machen. Wir sind sicher, dass es für solch eine Applikation viele Interessenten gibt, uns eingeschlossen.

Akkorderkennung ist keine leichte Aufgabe und ist noch kein ausgeschöpftes Gebiet. Wir hatten vor dieser Projektarbeit lediglich unser Musikwissen und mussten uns das fehlende Wissen in den Bereichen Mathematik, digitale Signalverarbeitung und Music Information Retrieval aneignen. Doch wir scheuten uns nicht diese Arbeit dennoch anzupacken, auch wenn das am Anfang hiess, dass wir viel lesen und wenig programmieren mussten.

Obwohl wir sehr grosses Interesse daran haben, am Ende eine praktische Applikation in diesem Bereich zu entwickeln, wussten wir, dass der Rahmen dieser Projektarbeit nicht reicht, um solch eine Applikation noch während der Projektarbeit zu entwickeln. Dies muss auf später verschoben werden. Dennoch glauben wir, dass uns dieses Wissen für unser weiteres Berufsleben helfen wird, nicht nur im Bereich Akkorderkennung, sondern auch in den anderen genannten Bereichen.

Auch wollen wir erreichen, dass diese Arbeit später Anderen, welche auch in dieses Gebiet einsteigen wollen, nützlich sein wird. Bei so einer anspruchsvollen Arbeit war uns wichtig, dass wir auf Betreuer zurückgreifen können, welche bereits viel Erfahrung in diesem Gebiet haben und uns bei Problemen unterstützen können.

Wir danken unseren Betreuern, die uns bei Problemen unterstützt haben und uns mit hilfreicher Literatur und Quellen versorgt haben. Auch danken wir für die Geduld, die sie uns entgegen gebracht haben, wenn wir in Durststrecken geraten sind und es nicht so optimal lief wie erhofft.

Zürcher Hochschule
für Angewandte Wissenschaften

School of
Engineering

Erklärung betreffend das selbständige Verfassen einer Projektarbeit an der School of Engineering

Mit der Abgabe dieser Projektarbeit versichert der/die Studierende, dass er/sie die Arbeit selbständig und ohne fremde Hilfe verfasst hat. (Bei Gruppenarbeiten gelten die Leistungen der übrigen Gruppenmitglieder nicht als fremde Hilfe.)

Der/die unterzeichnende Studierende erklärt, dass alle zitierten Quellen (auch Internetseiten) im Text oder Anhang korrekt nachgewiesen sind, d.h. dass die Projektarbeit keine Plagiate enthält, also keine Teile, die teilweise oder vollständig aus einem fremden Text oder einer fremden Arbeit unter Vorgabe der eigenen Urheberschaft bzw. ohne Quellenangabe übernommen worden sind.

Bei Verfehlungen aller Art treten die Paragraphen 39 und 40 (Unredlichkeit und Verfahren bei Unredlichkeit) der ZHAW Prüfungsordnung sowie die Bestimmungen der Disziplinarmassnahmen der Hochschulordnung in Kraft.

Ort, Datum:

……

Unterschriften:

……

……

……

Das Original dieses Formulars ist bei der ZHAW-Version aller abgegebenen Projektarbeiten zu Beginn der Dokumentation nach dem Abstract bzw. dem Management Summary mit Original-Unterschriften und -Datum (keine Kopie) einzufügen.

Inhaltsverzeichnis

1. Einleitung

1.1. Motivation

Für Musiker, Entwickler von Software im Zusammenhang mit Musik und Forscher im Bereich Music Information Retrieval (MIR) ist es sehr interessant, möglichst viele Eigenschaften von Audioaufnahmen in Popmusik automatisch bestimmen zu können. Dies eröffnet viele Möglichkeiten für interessante Applikationen. Akkorde sind sehr informativ, weil mit diesen Angaben und einem Musikinstrument bereits sehr gut Musiziert werden kann.

Die interessante Aufgabe hierbei ist die Implementierung eines Algorithmus, welche eine möglichst genaue Akkordfolgenerkennung erzielt. Es gibt bereits umfangreiche Forschungen in diesem Bereich und auch einige gute Lösungen. Ein Beispiel ist der Music Information Retrieval Evaluation eXchange (MIREX), welcher jedes Jahr Wettbewerbe im Bereich von MIR veranstaltet (IMIRSEL 2015). Dies ist ein weiterer Indikator, dass Bedarf besteht und das Gebiet nicht ausgeschöpft ist.

Westliche Popmusik ist sehr unvorhersehbar. Neben dem Gesang spielen viele verschiedene Instrumente eine Rolle. Heutzutage kommt es oft vor, dass Instrumente anderer Stile, wie das Klavier, in die Popmusik eingebracht werden.

Alle diese Argumente machen klar, dass eine solche Projektarbeit seinen Sinn hat. Es soll der Grundstein gelegt werden, um automatische Akkordfolgenerkennung mithilfe dieser Arbeit umsetzen zu können.

1.2. Problemstellung

Musiker wünschen sich manchmal, sie könnten ein Musikstück einfach in eine Software laden und diese gibt die Akkordfolge des Stücks zurück. Dies soll in Echtzeit oder nach einer Rechenphase geschehen. So kann der Musiker das Musikstück ohne langes Probieren nachspielen.

Es wird also eine Software benötigt, die Poplieder auf ihre Akkorde hin untersucht und diese ausgibt. Unter allen möglichen Musikrichtungen wird in dieser Arbeit lediglich die Akkorderkennung von westlicher Popmusik erforscht. Zusammenhänge zu anderen Musikrichtungen können zwar möglich sein, werden jedoch in dieser Arbeit weder zu erzielen versucht noch thematisiert.

Die Erkennung der Akkorde soll automatisiert sein und mit möglichst vielen verschiedenen Popliedern funktionieren. Es soll nach Methoden in den wissenschaftlichen Gebieten Musiktheorie, Mathematik, Digitale Signalverarbeitung und Music Information Retrieval (MIR) gesucht werden. Diese Methoden sollen auf ihre Effektivität untersucht werden und Empfehlungen extrahiert werden. Diese Erkenntnisse sollen dem Leser Hilfestellung bei der Implementierung einer solchen Software bieten. Die Methoden werden prototypisch mit Code umgesetzt, wobei die Prototypen softwaremässig nicht elaboriert sein müssen. Mithilfe dieser Prototypen werden die Methoden auf ihre Effektivität getestet.

1.3. Gliederung des Dokuments

Zu Beginn ist das nötige Fachwissen für diese Arbeit zusammengefasst. Neben den Grundlagen der Musiktheorie, wie die Darstellung und der Zusammenhang mit den Frequenzen, werden die Analyse von Signalen und Methoden zur Akkorderkennung beschrieben.

Anschliessend geht es zur eigentlichen Arbeit über. Im Kapitel Konzept sind die Gedankengänge vor der Implementierung des System niedergeschrieben. So wurde es vor der Implementierung entworfen. Darauf folgend ist das implementierte System mit seinen Komponenten beschrieben.

Im Kapitel 5 werden die Ergebnisse dargestellt und andere Algorithmen vorgestellt. Im Kapitel 6 wird ein Ausblick auf die Zukunft gegeben, was noch zu verbessern wäre.

Anschliessend folgen Verzeichnisse und Anhang. Im Anhang befindet sich der geplante Projektplan, eine Auflistung des Datenträgerinhalts und die offizielle Aufgabenstellung.

2. Theoretische Grundlagen

Das erste Unterkapitel befasst sich mit Musiktheorie. Dabei werden alle nötigen Grundlagen erklärt, die für die Akkorderkennung nützlich sind. Dieses Unterkapitel soll jedem Laien in Musik ermöglichen, den Gedanken und Vorgängen der Akkorderkennung zu folgen. Im zweiten Unterkapitel wird die Fourier-Analyse angeschaut, ein wichtiges Werkzeug in der digitalen Signalverarbeitung und ein essentielles Instrument zur Akkordfolgenerkennung. Auch hier wird lediglich auf die nötigen Funktionen eingegangen, die bei der Akkorderkennung erforderlich sind. Zuletzt werden Möglichkeiten zur Akkorderkennung besprochen. Dies beinhaltet sowohl die Basismethoden als auch die Vor- und Nachbearbeitungsmethoden um die Erkennung zu verbessern.

Die hier behandelte Theorie und Methoden, speziell die Kapitel Musiktheorie und Akkorderkennung, beziehen sich ausschliesslich auf Popmusik. Es könnten zwar Gemeinsamkeiten in anderen Musikstilen auftreten, es wird jedoch keine Garantie gegeben, dass die aufgeführten Methoden und Entscheidungen auch mit anderen Musikstilen funktionieren und bei ihnen angewendet werden sollen.

2.1. Musiktheorie

Zuerst wird Musik auf der Ebene der Schwingungen und Frequenzen erklärt. Anschliessend wird genauer auf Musiknoten eingegangen und wie sie zu interpretieren sind. Als nächstes wird erklärt, was Intervalle sind und sie ausmachen. Anschliessend wird das wichtigste zum Thema Akkorde erläutert, mit Gewichtung auf Dreiklänge. Zuletzt folgt eine kurze Einführung in die MIDI Repräsentation, welche in dieser Arbeit auch als Werkzeug der Akkorderkennung genutzt wurde.

Dieses Kapitel ist eher kurz gehalten und ist bei weitem nicht abschliessend. Neben Google sind Lehrklaenge.de (Gorski o. J.) und Musiklehre Online (Kaiser-Kaplaner o. J.) nützliche Webseiten, um sich in Musiktheorie einzulesen und nachzuschlagen, falls die Erklärungen hier nicht reichen.

2.1.1. Musik als Schwingungen und Frequenzen

Ein Ton wird durch ein vibrierendes Objekt erzeugt, wie beispielsweise die Stimmbänder eines Sängers oder die Saite einer Gitarre. Diese verändern die Moleküle in der Luft und lassen sie schwingen. Die verschiedenen Drücke gehen als Schwingungen durch die Luft. Beim Ziel werden diese vom menschlichen Gehirn interpretiert oder mithilfe eines Mikrofons in ein elektrisches Signal umgewandelt. Beim Menschen wird die Schwingung zur Gehörtrommel weitergeleitet, welche dann die Schwingungen ihrerseits wiederholt. Am Ende werden diese durch Mittel- und Innenohr prozessiert, an das Nervensystem weitergegeben und so vom Gehirn interpretiert. Grafisch kann die Veränderung vom Luftdruck an einem Zeitpunkt in einer Druck-Zeit-Kurve, also einem Schwingungsverlauf, dargestellt werden. Der Schwingungsverlauf zeigt die Differenz vom Luftdruck zum durchschnittlichen Luftdruck. (vgl. Müller 2015, 19)

Wenn hoher und tiefer Luftdruck sich regelmässig und alternierend wiederholen, so ist der Schwingungsverlauf periodisch. In diesem Fall ist die Periode der Schwingung durch die benötigte Zeit um einen kompletten Zyklus zu durchlaufen definiert. Die Frequenz, in Hertz (Hz) gemessen, steht Reziprok zur Periode. Der Sinusoid ist der simpelste Typ eines periodischen Schwingungsverlaufes. (vgl. ebd., 21)

In der Abbildung 2.1 dauert eine Periode ein Viertel einer Sekunde, das entspricht einer Frequenz von 4 Hz. Ein Sinusoid kann vollständig durch seine Frequenz, seine Amplitude (der Punkt der höchsten Differenz zum Mittelwert) und seiner Phase (an welcher Stelle des Zyklus der Sinusoid am Nullpunkt ist) beschrieben werden. Diese drei Attribute sind wichtig, um Audiosignale zu analysieren. (vgl. ebd., 21)

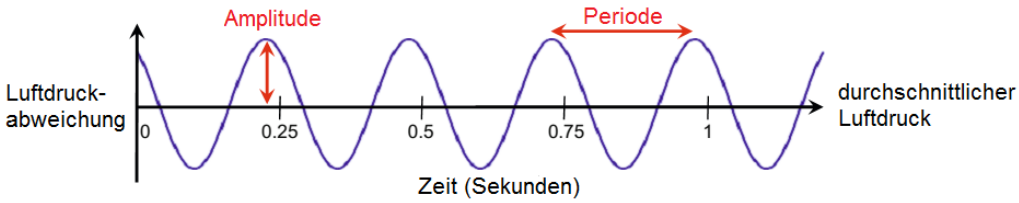


Abbildung 2.1.: Schwingungsverlauf eines Sinusoid mit einer Frequenz von 4 Hz.

Je höher die Frequenz einer sinusoidalen Schwingung ist, desto höher klingt der Ton. Die hörbare Frequenz für den Mensch liegt zwischen zirka 20 Hz und 20'000 Hz (20 kHz). Für andere Lebewesen ist dies unterschiedlich: Hunde können mehr als doppelt so hohe Frequenzen wahrnehmen, Fledermäuse sogar fünfmal so hoch wie der Mensch. Der resultierende Ton eines Sinusoid wird auch" harmonischer Klang" oder "purer Ton" genannt. Die Tonhöhe (engl. pitch) ist eigentlich nur eine subjektive Wahrnehmung. Bei puren Tönen ist die Beziehung zwischen Frequenz und Tonhöhe klar. Beispielsweise wissen wir, dass ein Sinusoid mit der Frequenz 440 Hz der Tonhöhe A4 entspricht, dem sogenannten Kammerton (engl. concert pitch). Diese wird als Referenztonhöhe genutzt, um eine Gruppe musikalischer Instrumente für einen Auftritt zu stimmen. (vgl. ebd., 21)

Frequenzen werden als ähnlich wahrgenommen, wenn sie sich um den Faktor 2 unterscheiden. Beispielsweise klingen die Töne A3 (220 Hz), A4 (440 Hz) und A5 (880 Hz) ähnlich. Auch wird die Distanz zwischen A3 und A4 gleich wahrgenommen wie die Distanz zwischen A4 und A5. Also kann man folgern, dass die menschliche Wahrnehmung der Tonhöhe logarithmischer Natur ist. Man kann eine Skala erstellen, welche eine Oktave in 12 Halbtöne aufteilt und auf einer logarithmischen Frequenzachse basiert. Jeder Tonhöhe kann eine Mittenfrequenz (engl. center frequency) $F_{Tonhöhe}(p) = 2^{(p-69)/12} \times 440$ [Hz] zugeordnet werden, wobei p die Tonhöhe ist und $p \in [0:127]$ entsprechend der MIDI Notennummer (siehe Kapitel MIDI Repräsentation). Diese Formel ergibt $F_{Tonhöhe}(69) = 440$ für die Referenztonhöhe A4. Um die Tonhöhe um eine Oktave zu erhöhen wird p um 12 erhöht, was zu einer Erhöhung um den Faktor 2 führt: $F_{Tonhöhe}(p+12) = 2 \times F_{Tonhöhe}(p)$. Auch ist das Verhältnis von zwei Tonhöhen p und $p+1$ konstant: $F_{Tonhöhe}(p+1)/F_{Tonhöhe}(p) = 2^{1/12} \approx 1.059463$. Dies bedeutet, die Tonhöhe wird um einen Halbton (engl. semitone) erhöht sobald die Mittenfrequenz um diese Konstante multipliziert wird. Die Masseinheit Cent ist eine logarithmische Einheit, die für musikalische Intervalle Verwendung findet. Per Definition ist eine Oktave in 1200 Cent unterteilt, wobei ein Halbton 100 Cent entspricht. Hier ist auch der Unterschied von einem Cent mit einer konstante $2_{1/1200} \approx 1.0005777895$. Der Unterschied in Cent zwischen zwei Frequenzen ω_1 und ω_2 ist gegeben durch $log_2(\frac{\omega_1}{\omega_2}) \times 1200$. Das Intervall von einem Cent ist zu klein um von Menschen wahrgenommen zu werden. Die Schwelle des Wahrnehmbaren wird "differentielle Wahrnehmbarkeitsschwelle" (oder auch JND von engl. just noticeable difference) genannt und variiert von Person zu Person sowie anderen Aspekten, wie musikalischem Kontext oder Klangfarbe. Als normaler Erwachsener können Unterschiede von 25 Cent erkannt werden, trainierte Hörer erkennen sogar 10 Cent weniger. (vgl. ebd., 22)

Töne im echten Leben sind jedoch weit weg von simplen Tönen mit einer wohldefinierten Frequenz. Wenn eine Note auf einem Instrument gespielt wird, so kann das in einem komplexen Ton resultieren. Dies ist ein Mix von verschiedenen Frequenzen, welche sich in fortlaufender Zeit noch verändern. Ein solcher musikalischer Ton (engl. musical tone) kann als eine Überlagerung von puren Tönen, also Sinusoiden, beschrieben werden, definiert durch die Frequenz, der Amplitude und der Phase. Ein Partialton (engl. partial) kann jeder Sinusoid sein mit dem ein musikalischer Ton beschrieben ist. Die Frequenz des tiefsten verfügbaren Partialtons ist die fundamentale Frequenz (engl. fundamental frequency). Ein harmonischer Partialton (engl. harmonic oder harmonic partial) ist ein ganzzahliges Vielfach der fundamentalen Frequenz. Da Partialtöne in der Frequenzachse hochgezählt werden, ist die fundamentale Frequenz der erste Partialton sowie der erste harmonische Partialton. Die Abbildung 2.2 zeigt eine Reihe von harmonischen Partialtönen. (vgl. ebd., 22-23)

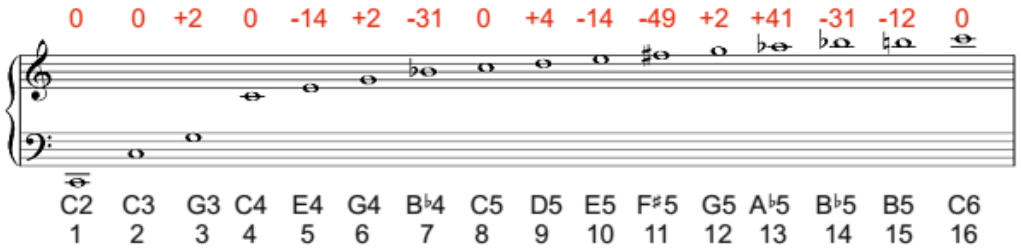


Abbildung 2.2.: Illustration von harmonischen Partialtönen (Müller 2015, 24). Angefangen bei der Note C2 wird hier für jeden der 16 ersten harmonischen Partialtöne die nächste musikalische Note gezeigt. Die Zahlen oben in rot beschreiben den Unterschied in Cent zwischen der Frequenz des harmonischen Partialtons und der Mittenfrequenz der nächsten Note.

2.1.2. Musiknoten

Als Musiknote (engl. note) kann sowohl das Symbol auf dem Notenblatt als auch der Ton als Audiorepräsentation gemeint sein. Jede Note hat Attribute, die einen Musiker wissen lassen, wie lange und welche Tonhöhe er spielen soll. Bei einem Piano zum Beispiel, sagt die Tonhöhe aus, welche Taste gespielt werden soll, die Dauer sagt aus, wie lange der Musiker die Taste drücken soll. (vgl. Müller 2015, 3)

Zwei Noten mit fundamentalen Frequenzen, die im Verhältnis gleich einer beliebigen Quadrierung (zum Beispiel halb, zwei oder vier Mal), werden als ähnlich wahrgenommen. Noten mit dieser Beziehung können einer Tonklasse (engl. pitch class) zugeordnet werden. Dies führt auch zur Beschreibung der Oktave, welche als Intervall zwischen einer Musiknote und einer anderen mit halber oder doppelter fundamentalen Frequenz definiert ist. Wenn wir diese Definition verwenden, so ist die Tonklasse eine Reihe von Noten, die je acht Schritte voneinander entfernt sind. Um Musik mit einer finiten Anzahl an Symbolen beschreiben zu können, müssen alle Tonhöhen in einer musikalischen Skala (engl. musical scale) beschrieben werden. Es wurden bereits viele verschiedene Skalen beschrieben und es gibt viele Diskussionen darüber welche besser geeignet sind. Die Gültigkeit einer Skala wird entschieden durch die Art der Musik, die benutzten Instrumente, das Genre oder der kulturelle Hintergrund. Eine Skala, die beispielsweise für westliche Pianomusik geeignet ist, muss nicht unbedingt für indische Sitar Musik passen. Es gibt keine allgemeingültige Skala und die Wahl einer Skala geht mit Vereinfachungen aus praktischen Gründen einher. Wir gehen hier von einer Skala mit zwölf Tönen aus, in der eine Oktave in zwölf Skalenschritte unterteilt ist (engl. twelve-tone equal-tempered scale). Die fundamentalen Frequenzen dieser Schritte sind auf einer logarithmischen Frequenzachse gleich aufgeteilt. Der Unterschied zwischen den fundamentalen Frequenzen zweier Skalenschritte heisst Halbton, welches das kleinste mögliche Intervall in dieser Skala ist. (vgl. ebd., 3-4)

In unserer Skala gibt es zwölf Tonklassen. Diese werden in der westlichen Notation durch einen Buchstaben und einem Vorzeichen (engl. accidental) dargestellt. Sieben der Tonklassen werden beschrieben durch die Buchstaben C, D, E, F, G, A, B. Diese beschreiben die weissen Tasten eines Klaviers. Die fünf restlichen sind die schwarzen Tasten des Klaviers (siehe Abbildung 2.3), welche beschrieben werden durch den Buchstaben und einem Vorzeichen (♯ oder ♭). Das Kreuz (♯, engl. sharp) erhöht die Note um einen Halbton. Die Erniedrigung um einen Halbton wird mit ♭ (engl. flat) symbolisiert. Die restlichen Tonklassen können folglich beschrieben werden durch $C^{\sharp}$, $D^{\sharp}$, $F^{\sharp}$, $G^{\sharp}$, $A^{\sharp}$ oder $D^{\flat}$, $E^{\flat}$, $G^{\flat}$, $A^{\flat}$, $B^{\flat}$. Um die Noten zusammen mit der Tonhöhe zu symbolisieren, wird nach der wissenschaftlichen Tonhöhen-Notation (engl. scientific pitch notation) jede Note mithilfe dem Namen der Tonklassen gefolgt von der Nummer der Oktave geschrieben. Die Note A4 hat eine fundamentale Frequenz von 440 Hz und gilt als Referenz. Die Nummer der Oktave addiert um eins, bei einer Erhöhung der Note der Tonhöhe B, resultiert in einer Note der Tonhöhe C. Das bedeutet zum Beispiel, dass nach Erhöhung der Note B4 die Note C5 folgt. Ähnlich dazu verhält sich die Erniedrigung, bei der die Nummer der Oktave um eins subtrahiert, von einer Note der Tonhöhe C, zu einer Note der Tonhöhe B wird. Die tiefste Note dieser Notation ist C0, welche eine fundamentale Frequenz von 16 Hz hat, was bereits unter der von Menschen wahrnehmbaren Grenze liegt. (vgl. ebd., 4)

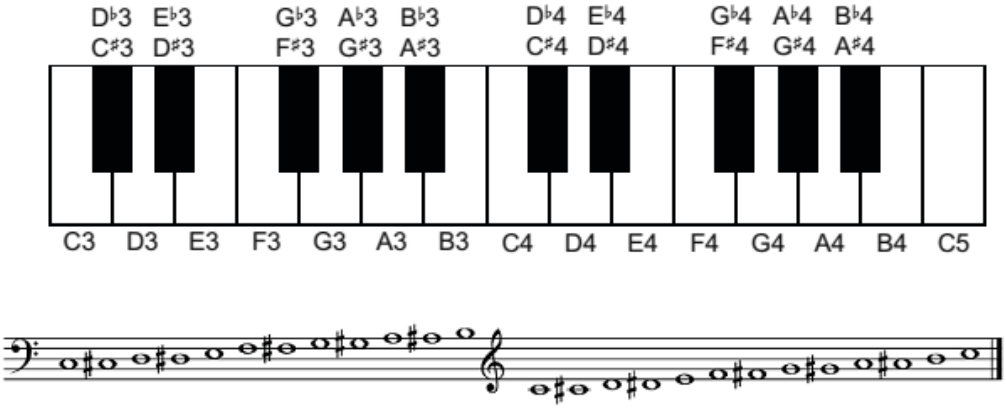


Abbildung 2.3.: Illustration einer Pianonotation (Müller 2015, 4). Oben sieht man ein Teil eines Pianokeyboards mit Tasten von C3 bis C5. Unten sieht man die zugehörigen Noten in westlicher Notation.

Wenn man alle Noten dieser Skala nach ihrer Tonhöhe ordnet, so erstellt man eine chromatische Skala (engl. chromatic scale). Vom griechischen Wort Chroma, was Farbe bedeutet, so wird auch hier jede Tonklasse die Noten enthalten, welche als ähnlich wahrgenommen werden. Zum Beispiel haben die Noten C3 und C5 den gleichen Chromawert, diese gehören folgend auch zur gleichen Tonklasse. So könnte man bei Noten, die nicht in der gleichen Tonklasse sind, sagen, dass sie eine andere "Tonfarbe" haben, daher sich auch nicht ähnlich anhören. (vgl. ebd., 4-5)

2.1.3. Intervall

Ein Intervall in der Musik kann vereinfacht als Differenz zwischen zwei Tonhöhen bezeichnet werden. Die Oktave ist ein Intervall, welche definiert ist als die Distanz zwischen einer Tonhöhe und einer anderen, mit halber oder doppelter fundamentalen Frequenz. Mit diesem Intervall können andere Intervalle mithilfe der Frequenzverhältnisse der Harmonien (physikalische Herangehensweise), geometrischen Verhältnisse (mathematische Herangehensweise) oder Notenrelationen (musische Herangehensweise) bestimmt werden. Dies führt zu leicht verschiedenen Begriffen des Intervalls, welche aber die gleichen Namen haben. Abbildung 2.4 zeigt eine Reihe von Intervallen. (vgl. Müller 2015, 239)

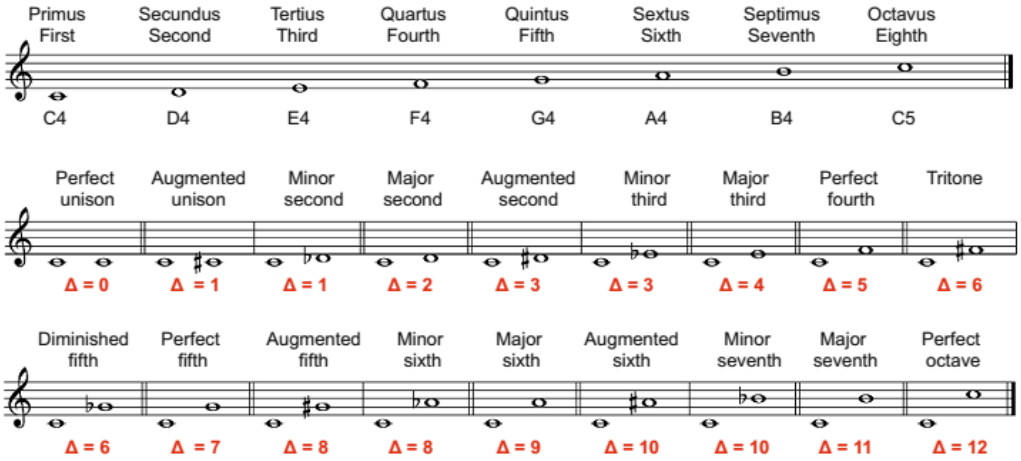


Abbildung 2.4.: Illustration von Intervallen (Müller 2015, 240). Die oberste Notenzeile zeigt die C-Dur Skala mit ihren bildenden Noten. Die zweite und dritte Notenzeile zeigt die Repräsentation von verschiedenen Intervallen, wobei Δ die Distanz in Halbtönen definiert.

Bei der musischen Herangehensweise für westliche Musik wird von einer Skala ausgegangen, bei der die Oktave in zwölf Skalenschritte aufgeteilt ist, und zwar gleichmässig verteilt auf einer logarithmischen Frequenzachse (engl. twelve-tone equal-tempered scale). Der kleinste mögliche Intervall wird ein Halbton (engl. semitone) genannt, welcher die Differenz zwischen zwei Skalenschritte ausmacht. Bei Frequenzen beschreibt der Halbton eher ein Verhältnis als eine Differenz. Aufgrund des Begriffs eines Halbtons, können nun andere Intervalle definiert werden, welche in der westlichen Musiktheorie benutzt werden. Die Abbildung 2.5 zeigt die am häufigsten vorkommenden Intervalle. (vgl. ebd., 240)

Δ	Interval name	Interval	JI ratio	Pyt. ratio
0	(Perfect) unison	C4 – C4	1:1	1:1
1	Minor second	C4 – D♭4	15:16	$3^5:2^8$
2	Major second	C4 – D4	8:9	$2^3:3^2$
3	Minor third	C4 – E♭4	5:6	$3^3:2^5$
4	Major third	C4 – E4	4:5	$2^6:3^4$
5	(Perfect) fourth	C4 – F4	3:4	$3:2^2$
6	Tritone	C4 – F♯4	32:45	$2^9:3^6$ or $3^6:2^{10}$
7	(Perfect) fifth	C4 – G4	2:3	2:3
8	Minor sixth	C4 – A♭4	5:8	$3^4:2^7$
9	Major sixth	C4 – A4	3:5	$2^4:3^3$
10	Minor seventh	C4 – B♭4	5:9	$3^2:2^4$
11	Major seventh	C4 – B4	8:15	$2^7:3^5$
12	(Perfect) octave	C4 – C5	1:2	1:2

Abbildung 2.5.: Liste von Intervallen (Müller 2015, 241). Die erste Spalte ist die Differenz in Halbtönen, die zweite der Name des Intervalls, die dritte der Intervall mit C4 als Grundnote, die vierte das Verhältnis mit JI-Notation (engl. just notation) und die fünfte die pythagoreischen Verhältnisse.

Wie schon beschrieben, können Intervalle auch mithilfe von Physik bestimmt werden. Dabei werden die Frequenzverhältnisse betrachtet, die in harmonischen Partialtönen eines Tones auf natürliche Weise vorkommen. Die harmonischen Partialtöne sind die ganzzahligen Multiplikatoren einer fundamentalen Frequenz, welche die harmonische Serie eines Tones formen. Man errechnet die Intervalle aufgrund der Frequenzverhältnisse zwischen Partialtönen, die in der gleichen harmonischen Serie vorkommen. Dies zeigt Abbildung 2.6. Beispielsweise ist die Oktave ein Intervall zwischen den beiden ersten harmonischen Partialtönen, die Quinte (engl. fifth) ein Intervall zwischen dem zweiten und dritten Partialton, die Quarte (engl. fourth) ein Intervall zwischen dem dritten und vierten Partialton, und so weiter. So werden die Intervalle aufgrund von Verhältnissen definiert. Intervalle, welche auf diese Art definiert werden, werden pure oder reine Intervalle (engl. just interval) genannt. Ähnlich dazu ist auch die Stimmung basierend auf Harmonien bekannt als pure oder reine Stimmung (engl. just intonation). Die jeweiligen Frequenzverhältnisse mit reiner Stimmung sind in Abbildung 2.5 aufgeführt. (vgl. ebd., 241)

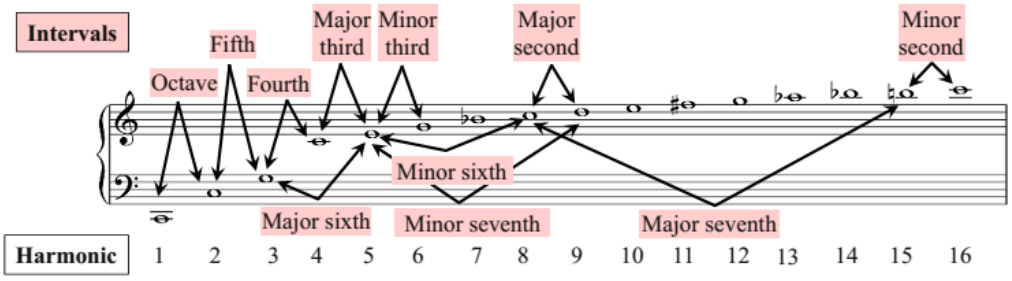


Abbildung 2.6.: Illustration einer harmonischen Serie in Musiknotation mit der Grundnote C2. (Müller 2015, 242)

Neben reiner Stimmung gibt es noch weitere Stimmsysteme, das älteste vorgestellt von dem griechischen Philosophen und Mathematiker Pythagoras. Dieses auf Geometrie basierende System ist erzeugt aus dem Frequenzverhältnis 1:2 der Oktave und 2:3 der Quinte. Die Anderen Intervalle sind aus diesen Verhältnissen berechnet, indem Quinten und Oktaven addiert oder subtrahiert werden. Die Intervalle können so nur durch Frequenzverhältnisse beschrieben werden, welche nur Zweier- oder Dreierpotenzen enthalten (siehe Abbildung 2.5). (vgl. ebd., 242)

Intervalle, die mit Verhältnissen von kleinen Ganzzahlen wie 1:1 für Prime (engl. unison), 1:2 für Oktave, 2:3 für Quinte oder 3:4 für Quarte, kommen natürlicherweise im tieferen Teil der harmonischen Serie vor und werden als stimmig und angenehm aufgenommen. Der Begriff des Einklangs (engl. consonance) beschreibt eine Kombination von Noten, die sich für die meisten Leute angenehm anhören, sobald sie zusammen gespielt werden. Die Unstimmigkeit (engl. dissonance) beschreibt den gegenteiligen Effekt, wobei die Kombination als streng oder unangenehm empfunden wird. Die Intervalle Prim, Oktave, Quinte und Quarte werden als perfekt in Stimmigkeit angesehen und werden manchmal auch perfekte Intervalle (engl. perfect intervals) genannt. Die Dur- und Mol-Terzen (engl. thirds) sowie die Dur- und Moll-Sechsten (engl. sixths) werden auch als stimmig empfunden, jedoch zu einem tieferen Grad (engl. imperfect consonance). Die restlichen Intervalle werden typischerweise als unstimmig angesehen. Der Tritonus (engl. tritone) wird als der unstimmigste Intervall bezeichnet. Wie man in Abbildung 2.5 sehen kann, besitzt dieser Intervall beim Frequenzverhältnis in reiner Stimmung (JI ratio), die höchsten Ganzzahlen. Der Tritonus ist somit auch der einzige Intervall, der nicht im tieferen Teil der harmonischen Serie vorkommt (siehe Abbildung 2.6). Wenn Noten miteinander gespielt werden, bezieht sich die Stimmigkeit darauf, wie die Harmonien der Töne zusammenfallen. Das bedeutet, bei der Stimmigkeit kommt es nicht nur auf die Grösse der Intervalle zwischen zwei Noten an, sondern auch auf die kombinierte spektrale Verteilung des resultierenden Tons. (vgl. ebd., 242-243)

2.1.4. Akkorde

Akkorde sind eine Gruppe von mehreren Tönen, welche zusammen gespielt werden. Viele erachten Akkorde als eine Gruppe von mindestens drei Tönen, manche erachten jedoch auch eine Gruppe von zwei Tönen als Akkord. Ein Zwei-Ton-Akkord heisst Dyade (engl. dyad), ein Dreiklang (engl. triad) hat drei Noten, ein Vierklang (engl. tetrad) vier Noten und so weiter. (vgl. Müller 2015, 243)

Die wichtigsten Akkorde der westlichen Musik sind die Dreiklänge. Diese können aufeinander gestapelt werden, wobei die tiefste note als Grundnote (engl. root note) bezeichnet wird. Es gibt sowohl Dur- als auch Moll-Dreiklänge. Es kann also zwischen vier verschiedenen Typen von Dreiklängen unterschieden werden. Abbildung 2.7 zeigt diese Typen. (vgl. ebd., 243)

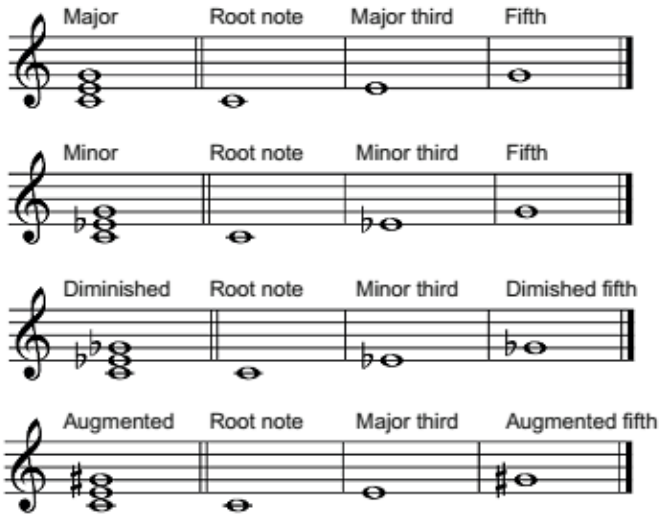


Abbildung 2.7.: Illustration verschiedener Typen von Dreiklängen auf der Grundnote C4 (Müller 2015, 244). Die erste Notenzeile zeigt den Dur-Dreiklang, die zweite Notenzeile den Mol-Dreiklang, die dritte Notenzeile den verminderten Dreiklang (engl. diminished triad) und die vierte Notenzeile den erweiterten Dreiklang (engl. augmented triad).

Jeder dieser Akkordtypen kann mit unterschiedlichen Grundnoten gespielt werden. Egal welche Grundnote genutzt wird, jeder Dreiklang-Typ hat eine andere Qualität im Ohr des Hörers. Der Dur-Dreiklang, welcher aus einem Dur-Terz und einer perfekten Quinte besteht, umfasst stimmige Intervalle. Der Klang dieses Dreiklangs wird meist als einheitlich, angenehm und fröhlich aufgenommen. Wenn die Tonhöhe der zweiten Note um einen Halbton vermindert wird, entsteht ein Moll-Dreiklang. Diese werden zwar auch als übereinstimmend und einheitlich aufgenommen, klingen jedoch eher traurig oder düster. Die anderen zwei Dreiklang-Typen werden als unstimmig und instabil aufgenommen. Verminderte und erweiterte Dreiklänge werden meist bei Übergängen zwischen stabileren Akkorden genutzt. (vgl. ebd., 244)

Da es zwölf verschiedene Grundnoten gibt, können auch je zwölf Dur- und Moll-Dreiklänge gebildet werden. Abbildung 2.8 zeigt eine Übersicht dieser 24 Dreiklänge, wobei die tiefste Note jedes Akkords auch die Grundnote ist. Der Dur-Akkord (engl. major chord) ist normalerweise mit demselben Symbol gekennzeichnet wie die Tonklasse seiner Grundnote. Zum Beispiel wird der Akkord C-Dur als C beschrieben und besteht aus drei Noten der Tonklassen C, E und G. Die Moll-Akkorde (engl. minor chords) werden gleich beschrieben, jedoch wird ein "m" angefügt, welches für das englische "minor" steht. Zum Beispiel wird der C-Moll Akkord als Cm beschrieben, wobei er aus drei Noten mit den Tonklassen C, $E^{\flat}$ und G besteht. Bei einer logarithmischen Frequenzachse (engl. twelve-tone equal-tempered scale) wird bei den Tonklassen nicht zwischen $C^{\sharp}$ und $D^{\flat}$ oder $G^{\sharp}$m und $A^{\flat}$m unterschieden, obwohl diese Akkorde aus musiktheoretischer Sicht unterschiedlich sind. Obwohl Akkorde aus verschiedenen Perspektiven

angeschaut werden können, beschränken wir uns hier darauf, dass ein Dur- oder Moll-Akkord definiert ist durch die Tonklassen oder Chromawerte seiner einzelenen Noten. Aus mathematischer Sicht kann ein Dreiklang als drei Elemente aus der Menge {C, C♯, D, ... B}, bestehend aus den zwölf Chroma-Attributen, definiert werden. So kann eine Untermenge aus drei Elementen als binärer Chromavektor mit drei Einträgen, mit gesetztem Wert 1 bei den jeweiligen Chromapositionen der Untermenge, beschrieben werden. Abbildung 2.7 zeigt das daraus resultierende Chromamuster für die 24 Dur- und Mol-Akkorde. Aufgrund dieses mathematischen Models können die zwölf Dur-Akkorde mithilfe von zyklischem Verschieben des C-Dur Dreiklangs auf zwölf verschiedene Arten generiert generiert. Ähnlich werden auch die Moll-Akkorde aus Cm generiert. Was Tonklasse angeht sind die Dur- und Moll-Akkorde eindeutig bestimmt was bedeutet, dass jede der 24 Dur- und Moll-Dreiklänge zu verschiedenen Untermengen von drei Elementen führt. In Noten jedoch, gibt es viele Alternativen um den gleichen Akkord zu realisieren. Wenn die tiefste Note eines Akkords auch die Grundnote ist, so ist der Akkord in normaler Form. Wenn jedoch die Grundnote nicht die tiefste Note des Akkords ist, so ist dies ein invertierter Akkord. (vgl. ebd., 245-246)

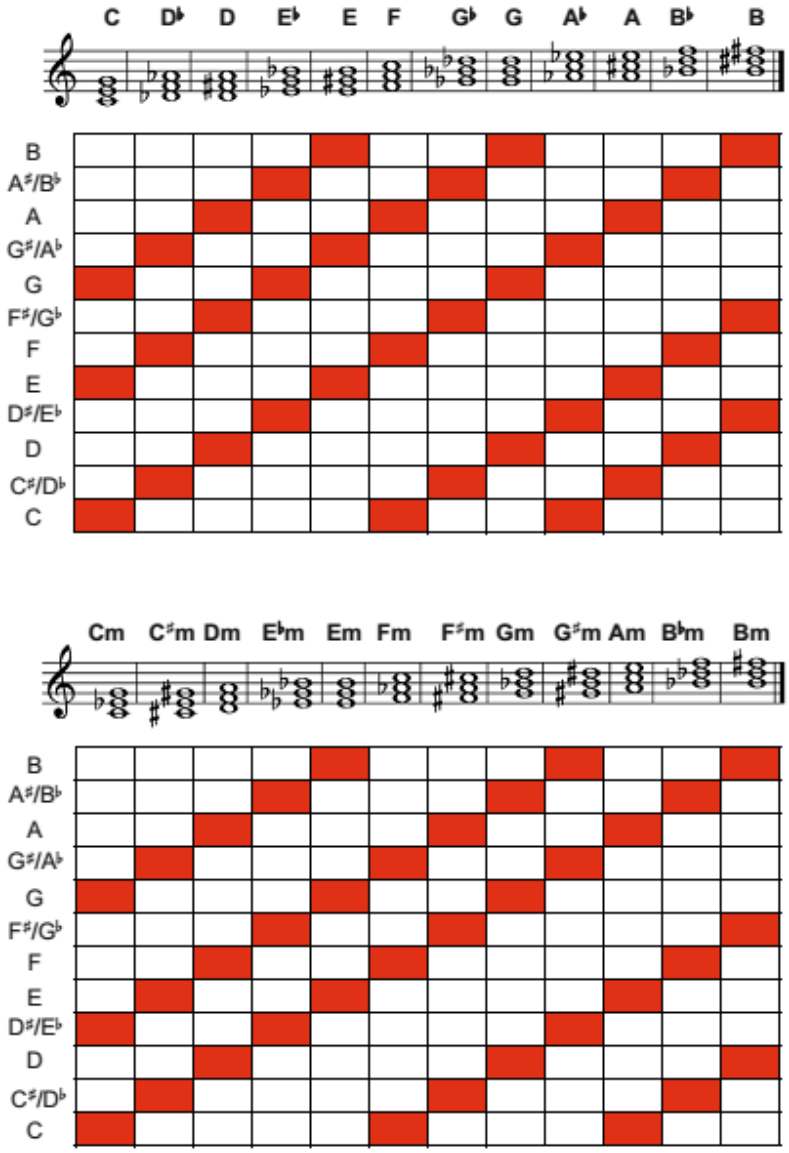


Abbildung 2.8.: Übersicht aller Dur- (oben) und Moll-Akkorde (unten) mit harmonischer Äquivalenz (Müller 2015, 245). Eine Zeile Partiturnotation ist gegeben mit möglichen Noten für jeden Akkord sowie das Chromamuster (engl. chroma pattern), wobei die nötigen Noten rot eingefärbt sind.

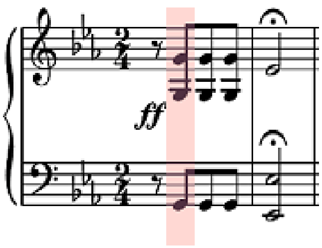
Neben Intervallen und Akkorden gibt es noch musikalische Skalen (oder auch Tonleitern genannt). Eine musikalische Skala kann als eine Menge von Noten gesehen werden, typischerweise sortiert mit aufsteigender Tonhöhe. Wenn ein Akkord als vertikale Struktur bezeichnet wird, so ist eine Skala eine horizontale Struktur. Eine Skala kann als eine Unterteilung einer Oktave in eine Anzahl von Skalenschritte gesehen werden, wobei jeder Skalenschritt eine Intervall zwischen zwei Noten ist. Hier wurde bereits auf die "twelve-tone equal-tempered scale" eingegangen, welche auch chromatische Skala (engl. chromatic scale) genannt wird. Es gibt noch die folgenden Skalen: Dur-Tonleiter (engl. major scale), Moll-Tonleiter (engl. minor scale), diatonische Tonleiter (engl. diatonic scale), pentatonische Tonleiter (engl. pentatonic scale) und die Ganztonleiter (engl. whole tone scale). Dur- und Moll-Tonleiter sind die Untermenge der chromatischen Skala und für die westliche Musik von Bedeutung. Auf die Skalen, wie sie augebaut sind und welche Eigenschaften sie haben, wird hier nicht mehr weiter eingegangen. (vgl. ebd., 246-247)

Musiker nutzen ihre Intuition und Erfahrung um Akkorde in einem harmonischen Fluss schön anzuordnen und zu kombinieren. Solch eine Abfolge von Akkorden in einer Zeitspanne wird Progression (engl. chord progression) genannt. Auch wenn es etliche Akkorde und Akkord-Progressionen geben kann, so gibt es Regeln von typischen Kombinationen für verschiedene Musikstile. Eines dieser Prinzipien, welche auch bei Film oder Literatur angewandt wird, ist das Wechselspiel von Spannungsauf- und Abbau. Manche Akkorde sind eher für den Aufbau und andere eher für den Abbau geeignet. Auch gibt es generelle Begebenheiten bei harmonischen Flüssen wie beispielsweise, dass ein C-Dur am wahrscheinlich eher von einem G oder Am gefolgt wird als von einem $C^\sharp$ oder Gm. Solche Phänomene können bei der Akkorderkennung genutzt werden, um die Erkennung zu verbessern, indem ein Akkord nicht als Einzelereignis, sondern als ein Element einer Progression angesehen wird. (vgl. ebd., 252-253)

2.1.5. MIDI Repräsentation

Eine mögliche symbolische Repräsentation von Musik basiert auf dem MIDI Standard (siehe Abbildung 2.9), was für Musical Instrument Digital Interface steht. MIDI lässt den Musiker in Echtzeit elektronische Instrumente oder digitale Synthesizer steuern. Wenn jemand eine Taste auf einem E-Piano drückt, startet er den zu wiedergebenden Ton, wobei die Tippschnelligkeit die Intensität des Tons entscheidet. Die gleichen Töne können auch direkt per Nachrichten an das Instrument erzeugt werden, welche Informationen beinhalten wie zum Beispiel die Schnelligkeit oder auch einen Zeitstempel wann welche Nachricht an der Reihe ist. Diese Nachrichten können von einem anderen elektronischen Instrument erzeugt werden oder auch mithilfe eines Computers. MIDI repräsentiert nicht musikalische Töne sondern nur die Art wie das Instrument gespielt wird. Die Spezifikation “Standard MIDI File (SMF)“ beschreibt, in welchem Format die Daten gespeichert werden sollen. Dieses Format ermöglicht es, die Daten unabhängig vom Betriebssystem zu verwenden. Es gibt unzählige Webseiten, die Musik dieser Art anbieten oder verkaufen. (vgl. Müller 2015, 13)

Wichtige Werte sind “Note-on“ und “Note-off“, welche Start und Ende einer Note beschreiben. Die MIDI Notennummer (engl. MIDI note number) ist eine Ganzzahl von 0 bis 127 und kodiert die Tonhöhe einer Note. Ähnlich zum akustischen Piano, wo die 88 Tasten den Tonhöhen A0 bis C8 entsprechen, so gehen die MIDI Notennummern von C0 bis $G^\sharp 9$. Zum Beispiel besitzt die Note C4 die MIDI Notennummer 60, der Kammerton A4 hat die Nummer 69. Die Schlüsselschnelligkeit (engl. key velocity) ist auch eine Ganzzahl zwischen 0 und 127 und reguliert die Intensität des Tons. Bei einem Note-on Ereignis beschreibt diese das Volumen, bei einem Note-Off Ereignis den Zerfall beim Beenden des Tons. Die Interpretation einer Schlüsselschnelligkeit wird vom Instrument oder Synthesizer bestimmt. Der MIDI Kanal (engl. MIDI channel) ist eine Ganzzahl zwischen 0 und 15. Dieser beschreibt, welche Teile von welchem Kanal interpretiert werden. Ein Kanal kann mehrere Noten gleichzeitig übernehmen. Der Zeitstempel ist eine Ganzzahl, die bestimmt, wie viele Takte oder Ticks gewartet wird, bevor das nächste Note-On oder Note-Off Ereignis ausgeführt wird. (vgl. ebd., 13-14)



Time (Ticks)	Message	Channel	Note Number	Velocity
60	NOTE ON	1	67	100
0	NOTE ON	1	55	100
0	NOTE ON	2	43	100
55	NOTE OFF	1	67	0
0	NOTE OFF	1	55	0
0	NOTE OFF	2	43	0
5	NOTE ON	1	67	100
0	NOTE ON	1	55	100
0	NOTE ON	2	43	100
55	NOTE OFF	1	67	0
0	NOTE OFF	1	55	0
0	NOTE OFF	2	43	0
5	NOTE ON	1	67	100
0	NOTE ON	1	55	100
0	NOTE ON	2	43	100
55	NOTE OFF	1	67	0
0	NOTE OFF	1	55	0
0	NOTE OFF	2	43	0
5	NOTE ON	1	63	100
0	NOTE ON	2	51	100
0	NOTE ON	2	39	100
240	NOTE OFF	1	63	0
0	NOTE OFF	2	51	0
0	NOTE OFF	2	39	0

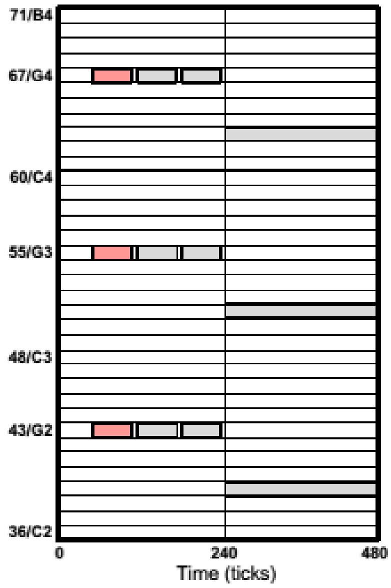


Abbildung 2.9.: Musikalische Repräsentationen der ersten zwölf Noten von Beethovens 5. Sinfonie (Müller 2015, 14). Ganz links als Notenblatt, in der Mitte als vereinfachte MIDI Repräsentation und rechts als Notenrolle eines Pianos (engl. piano-roll).

Ein Vorteil des MIDI Format ist, dass es sowohl die musikalischen als auch die physikalischen Anfangszeiten und Dauer der Noten beschreiben kann. Es werden keine absoluten Zeiteinheiten wie Mikrosekunden definiert, sondern Zeitinformation auf musikalischer Ebene wie bei einem Notenblatt. MIDI teilt eine Viertelnote in Basiszeiteinheiten der Ticks. Die Anzahl der Pulse per Viertelnote (PPQN) wird am Anfang im Header der MIDI Datei bestimmt und gilt dann für alle MIDI Nachrichten dieser Datei. Ein häufig genutzter Wert ist 120 PPQN, welcher die Auflösung der Zeitstempel für die Notenereignisse bestimmt. (vgl. ebd., 14–15)

Die MIDI Repräsentation ist sehr flexibel. Es können auch absolute Zeitinformationen eingegeben werden, um lokale Tempoänderungen möglich zu machen. Mit Temponachrichten kann die absolute Dauer eines Ticks berechnet werden. Bei 120 PPQN und 600000 μs pro Viertelnote, entspricht ein Tick 5000 μs. Von der Temponachricht kann auch die Anzahl der Viertelnoten pro Minute berechnet werden, welche in Beats pro Minute (engl. beats per minute, BMP) angegeben wird. 600000 μs pro Viertelnote entsprechen 100 BMP. (vgl. ebd., 15)

MIDI wurde entwickelt um Performanceprobleme bei elektronischer Musik zu lösen und ist in musikalischer Hinsicht limitiert. Beispielsweise kann nicht unterschieden werden zwischen D$^\sharp$4 und E$^\flat$4 welche beide die MIDI Notennummer 63 haben. (vlg. ebd., 15)

2.2. Fourier-Analyse von Signalen

Musikalische Signale sind komplex und bestehen aus verschiedenen Soundkomponenten. Die Extrahierung von musikalisch relevanten Informationen aus dem Kurvenverlauf alleine ist sehr schwer. Daher muss das gegebene Signal in eine Form gebracht werden, die die Weiterverarbeitung vereinfacht. In diesem Fall nutzen wir sinusoidale Funktionen. Daher heisst der Prozess Fourier-Analyse. Die sinusoidale Funktion ist speziell da sie explizite physikalische Bedeutung besitzt in Hinsicht auf Frequenzen. Die Fourier-Transformation konvertiert ein Signal, das auf Zeit beruht, in eine Repräsentation, welche auf Frequenzen beruht. Als eine der wichtigsten Werkzeuge in der Signalverarbeitung kommt sie in vielen Musikverarbeitungsprozessen vor, natürlich auch in der Akkorderkennung. (vgl. Müller 2015, 39)

Der inverse Prozess der Fourier-Transformation ist die Fourier-Repräsentation, welche ein Signal als gewichtete Überlagerung von unabhängigen elementaren Funktionen präsentiert. Jede der Gewichte zeigt den Grad, zu dem die elementare Funktion zum originalen Signal beiträgt und somit einen gewissen Aspekt eines Signals enthüllt. Sinusoiden sind besonders gut geeignet um als elementare Funktionen zu dienen. Jede der Gewichte wird dann einem Frequenzwert zugeordnet und drückt den Grad aus, in welchem das Signal eine periodische Oszillation zu dieser Frequenz hat. Die Fourier-Transformation kann also als ein Weg bezeichnet werden, um frequenzabhängige Gewichte zu berechnen. (vgl. ebd., 69)

Die Fourier-Transformation erzeugt ein Mittelwert der Frequenzinformationen über die ganze Zeitspanne. Wann aber welche Frequenzen auftreten ist dann versteckt. Um diese versteckte Information zu erhalten, nutzen wir die Short-Time-Fourier-Transformation. Statt das ganze Signal zu verarbeiten, wird auf einmal immer nur ein kleiner Teil des Signals verwendet. Dazu wird die Fensterfunktion (engl. window function) eingesetzt. Diese Funktion ist nur für kurze Zeit nicht Null. Das originale Signal wird dann mit der Fensterfunktion multipliziert, um so ein Fenstersignal (engl. windowed signal) zu bekommen. Um Frequenzinformationen für verschiedene Zeitspannen zu bekommen, wird die Fensterfunktion an der Zeitkomponente des Signals verschoben und eine Fourier-Transformation für jedes der Fenstersignale einzeln durchgeführt. (vgl. ebd., 53)

Die STFT hängt stark von der Fensterlänge ab, welche die Unterteilung bestimmt. Auch hängt die STFT stark von der Form der genutzten Fensterfunktion. Wenn beispielsweise die rechteckige Fensterfunktion genutzt wird, so gibt es typischerweise einen "Rieseleffekt". (vgl. ebd., 53)

Es gibt auch die dreieckige Fensterfunktion. Diese erzielt schon einen kleineren "Rieseleffekt". In der Signalverarbeitung wird of das Hann Fenster (oder auch Hanning Fenster) genutzt. Es kann einige ungewollte Artefakte der Fourier-Transformation weicher machen, führt jedoch auch zu einem Schmiereffekt. Das kann das Fenster glatter aussehen lassen als die Signaleigenschaften andeuten. (vgl. ebd., 98)

Die STFT gibt zu einem Signal für jeden Punkt in der Zeitspanne und der Frequenz eine komplexe Nummer zurück. In einem Spektrogramm werden die Resultate der STFT präsentiert. Bei einem Spektogramm repräsentiert die x-Achse die Zeit, die y-Achse die Frequenz und die Intensität oder Farbe die Dimension des Spektogrammwertes zu einer bestimmten Frequenz und Zeit. Bei der Darstellung gibt es verschiedene Möglichkeiten, wie zum Beispiel die Amplitude als die Höhe der dreidimensionalen Oberfläche darzustellen. Bei musikalischen Anwendungen wird die Frequenzachse oft logarithmisch angezeigt. Die Amplitudenwerte werden oft als logarithmische Skala, zum Beispiel in einer Dezibel-Skala, präsentiert. Abbildung 2.10 zeigt Beispiele von Spektogrammen eines Chirp-Signals. (vgl. ebd., 98-99)

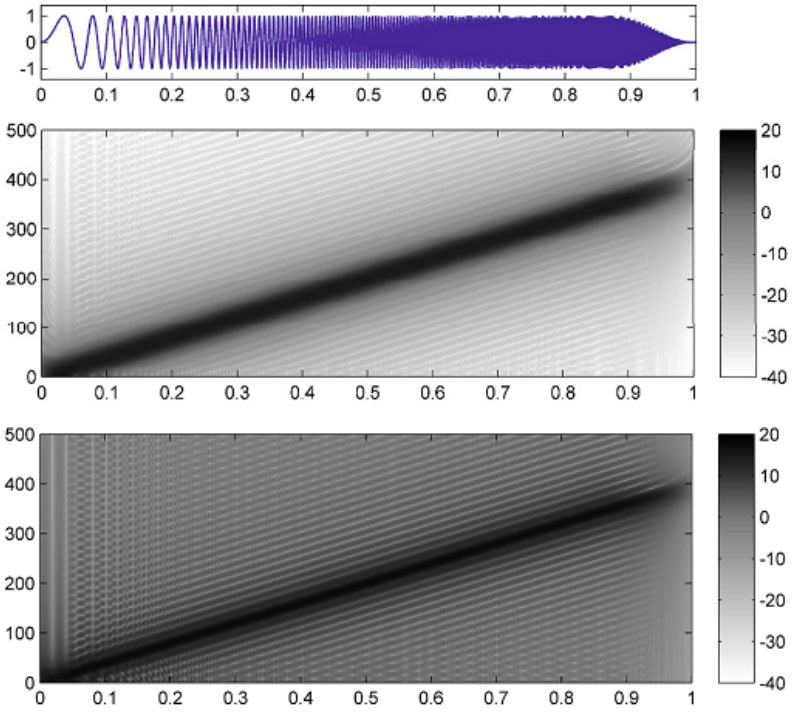


Abbildung 2.10.: Spektogramm von einem Chirp-Signal (Müller 2015, 100). Ganz oben ist das Signal, in der Mitte ein Spektogramm mit Hann Fenster und unten ein Spektogramm mit einer viereckigen Fensterfunktion, beide mit einer Fenstergrösse von 62.5 Millisekunden. Die x-Achse der Spektogramme beschreibt die Zeit in Sekunden, die y-Achse die Frequenz in Hertz.

2.3. Akkorderkennung mithilfe von Templates

In diesem Kapitel wird die Akkorderkennung mithilfe von Templates erklärt. Zuerst wird erklärt, wie so ein typisches Akkorderkennungssystem aufgebaut ist. Dann wird die Basismethode genauer erklärt und evaluiert. Zuletzt werden Verbesserungstechniken zur Template-basierten Akkorderkennung aufgezeigt.

2.3.1. Typisches Template-basiertes Akorderkennungssystem

Das typische Akkorderkennungssystem besteht im Wesentlichen aus zwei Schritten. Als erstes wird die Audioaufnahme in kleinere Frames unterteilt, welche dann in einen passenden Merkmalsvektor (engl. feature vector) übernommen werden. Die meisten Systeme greifen auf chromabasierte Audiomerkmale zurück, welche dabei helfen sollen, versteckte Toninformationen zu erkennen. Im nächsten Schritt werden die Merkmalsvektoren mithilfe von Patternmatching auf vordefinierte Akkordmodelle abgebildet. Das passendste Modell bestimmt den Akkord des jeweiligen Frames. Um bessere Ergebnisse zu erreichen, werden Verbesserungstechniken entweder vor dem Patternmatching (Prefiltering) oder während und nach dem Patternmatching (Postfiltering) angewandt. Eine Übersicht der Schritte eines typischen Akkorderkennungssystems zeigt Abbildung 2.11. (vgl. Müller 2015, 253)

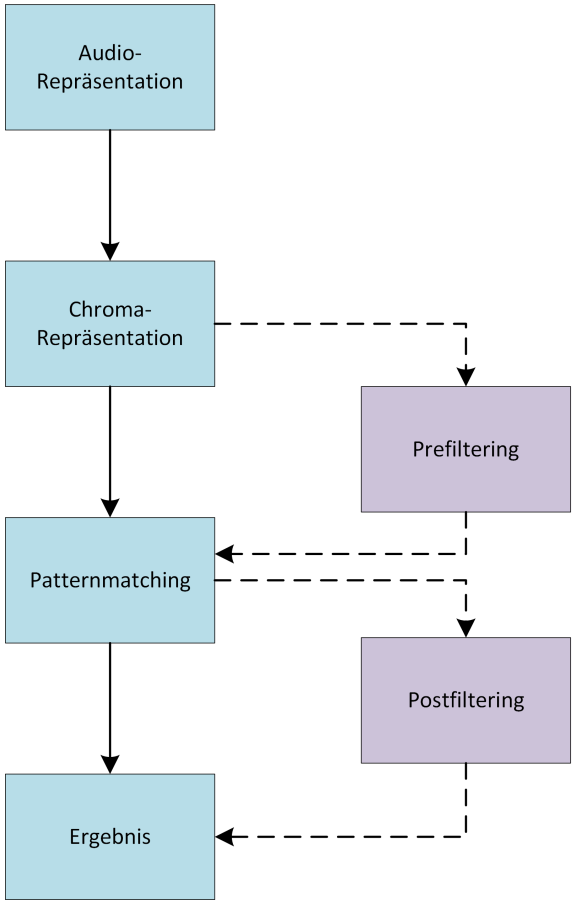


Abbildung 2.11.: Übersicht der Schritte eines typischen automatisierten Akkorderkennungssystems.

Im folgenden Kapitel wird die Basismethodik erklärt, um Akkorde mithilfe von Templates relativ simpel zu erkennen. Im gleichen Kapitel wird noch auf Probleme eingegangen, welche bei solchen Verfahren bestehen. Im anschliessenden Kapitel wird erklärt wie die genannten Probleme mithilfe von Pre- und Postfiltering verhindert werden können.

2.3.2. Basismethode

In einer gegebenen Audioaufnahme muss herausgefunden werden, welcher Akkord wann gespielt wurde. Daher wird die Aufnahme zuerst in eine Reihe von Merkmalsvektoren übernommen. Dann wird jeder Merkmalsvektor auf einen Akkord abgebildet. (vgl. Müller 2015, 254)

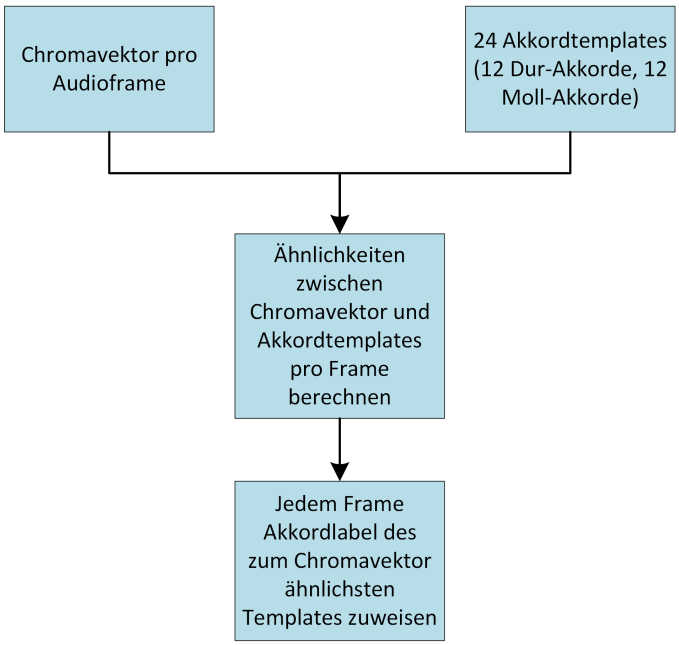


Abbildung 2.12.: Übersicht einer Template-basierten Akkorderkennungsprozedur.

Dafür wird eine Liste von möglichen Akkorden gewählt. In diesem Fall handelt es sich um die zwölf Dur und Moll Dreiklänge A = {C, C$^\sharp$, ... , B,Cm,C$^\sharp$m, ... ,Bm}. Um Merkmale zu extrahieren, wird auf Chromamerkmale gesetzt. (vgl. ebd., 254)

Es wird ein 12-dimensionaler Chromavektor definiert, in welchem alle möglichen Akkorde spezifiziert werden können. Zuerst wird eine vordefinierte Liste an Templates berechnet, welche als prototypische Chromavektoren betrachtet werden können. Diese repräsentieren spezifische Akkorde. Als nächstes wird der Akkord mithilfe des Templates bestimmt, welcher die grösste Ähnlichkeit zum Merkmalvektor hat. (vgl. ebd., 254-255)

Das bedeutet also, jeder mögliche Akkord kann mithilfe eines binären 12-dimensionalen Chromavektors $t = (x(0), x(1), ..., (x(11)))^T$ definiert werden, wobei $x(i) = 1$ nur gilt sobald der Chromawert im Akkord enthalten ist. Hier das Beispiel eines Chromavektors für den Akkord C-Dur $t_c := x = (1,0,0,0,1,0,0,1,0,0,0,0)^T$. Die zwölf Dur- und Moll-Akkorde können dabei durch zyklisches Verschieben abgerufen werden. (vgl. ebd., 255)

Es gibt viele mögliche Wege um Chromamerkmale mit Akkord-Templates zu vergleichen. Eine einfache Variante ist das innere Produkt der normalisierten Vektoren $s(x,y) = \frac{\langle x|y \rangle}{||x|| \times ||y||}$, wobei im Fall $||x|| = 0$ oder $||y|| = 0$ $s(x,y) = 0$ gesetzt wird. Das resultiert in $s(x,y) \in [-1,1]$. Wenn die Vektoren x und y nur positive Einträge haben, gilt $s(x,y) \in [0,1]$. (vgl. ebd., 257)

Um die Prozedur möglichst anschaulich aufzuzeigen, wird als Beispiel der Anfang von "Let It Be" der Beatles angeschaut. Als erstes wurde die Aufnahme in Chroma-Repräsentation abgebildet. Als nächstes wurde jeder Chromavektor mit allen 24 binären Akkordtemplates verglichen, sodass 24 Ähnlichkeitswerte ausgerechnet wurden, in der Abbildung 2.13 in der Form eines Zeit-Akkord-Graphen. (vgl. ebd., 257)

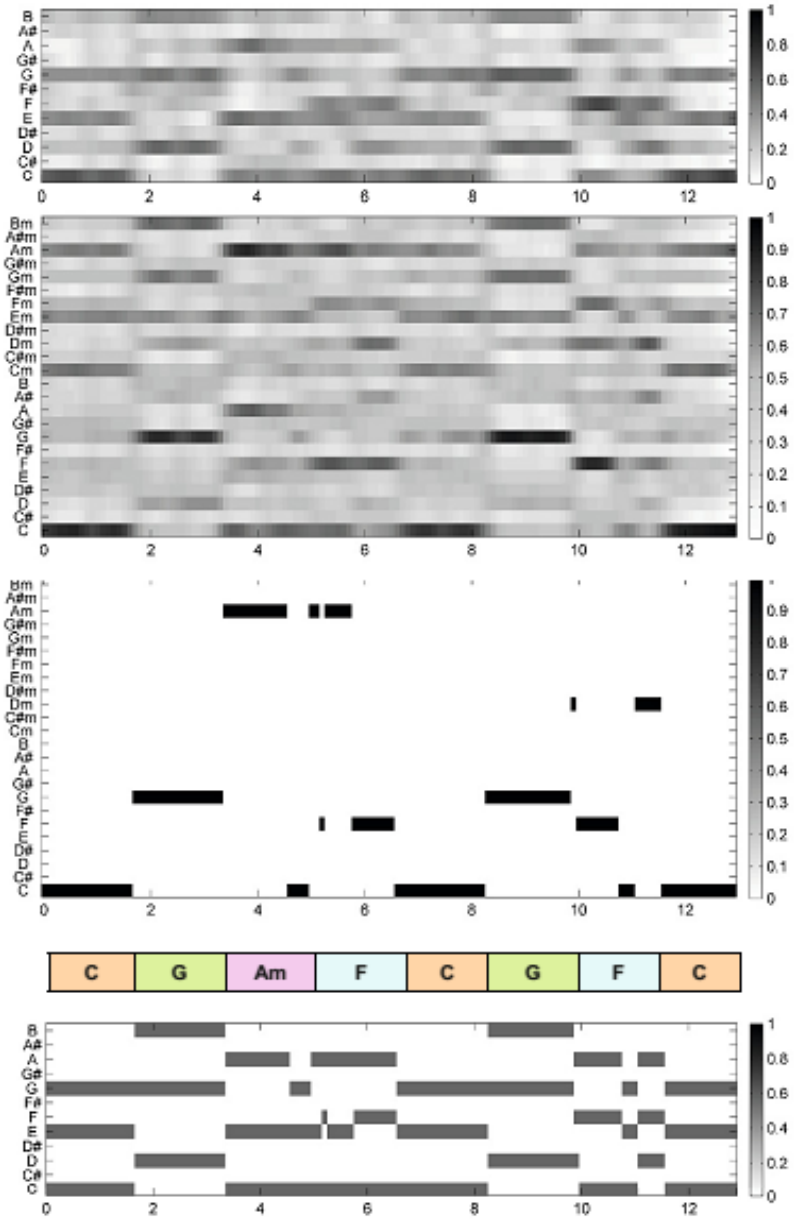


Abbildung 2.13.: Illustration (Müller 2015, 256) einer Template-basierten Akkorderkennung mithilfe von 24 Dur- und Moll-Akkorden der ersten Sekunden von "Let It Be" der Beatles. Ganz oben befindet sich die Chroma-Repräsentation. Als nächstes folgen die Ähnlichkeitswerte zwischen den Chromavektoren und den 24 Akkordtemplates. Dann folgt das Resultat der Akkorderkennung. Weiter unten befinden sich die manuell gesetzten Akkordannotationen eines Musikexperten. Als letztes das normalisierte binäre Template des Resultates der Akkorderkennung. Die x-Achse beschreibt jeweils die Zeit in Sekunden.

Man kann an der Visualisierung sehen, dass die Chromavektoren am Anfang des Musikstücks dem C-Dur C am ähnlichsten sind. Auch gibt es höhere Übereinstimmung mit den Templates für Cm, Em und Am. (vgl. ebd., 257)

Wie man in der Visualisierung sieht, stimmen die meisten Resultate mit den Annotationen des Musikexperten überein. Um aber die Qualität der Erkennung zu evaluieren, werden meist die resultierenden Annotationen mit Referenzannotationen (engl. ground truth) verglichen. Das wirft Fragen auf wie beispielsweise, wie der Vergleich quantifiziert wird, ob die Referenzannotation stimmt und ob die Abmachungen in dem gewählten Modell sinnvoll sind. Trotzdem muss solch ein Vergleich durchgeführt werden, denn sie sind hilfreiche Indikatoren um die Leistung der Akkorderkennung zu veranschaulichen und die Charakteristiken der Daten zu verstehen. (vgl. ebd., 257-258)

Hier wird die Evaluation darauf beschränkt, wie stark die Erkennung der Referenzannotation gleicht. Diese Referenzannotation wird meist von Musikexperten durchgeführt, die die nötige Erfahrung besitzen. Der Experte teilt meist die Partitur (engl. score) in Teile auf und vergibt diesen Akkordlabels, die dem gewählten Teil harmonisch am meisten gleichen. Die Teile können sich je nach Experten unterscheiden in Länge und Granularität. Dies kann also zu unterschiedlichen Akkordlabels, und daher auch unterschiedlichen Referenzdaten je nach Experten, führen. (vgl. ebd., 258-259)

Bei dem Beatles-Beispiel ist die Granularität sehr gering und einige Noten gehören gar nicht wirklich zu den Akkorden und sind nur als Übergänge vorhanden. Die Abbildung 2.14 zeigt die Akkordannotationen in feinerer Granularität. (vgl. ebd., 254)

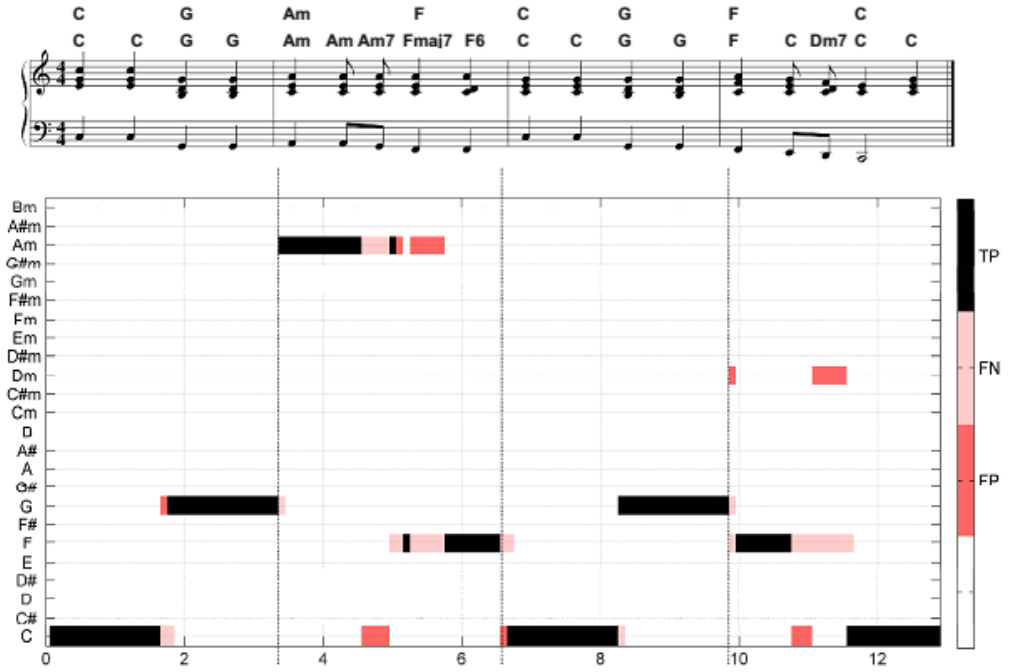


Abbildung 2.14.: Evaluation der Akkorderkennung von "Let It Be" der Beatles (Müller 2015, 258). Oben sind zwei unterschiedliche Akkordannotationen von Musikexperten. Die eine Annotation ist auf Basis jeder zweiten Viertelnote, die andere Annotation ist feiner. Unten sieht man die Evaluation auf Basis jeder zweiten Viertelnote. Die x-Achse der Evaluation beschreibt die Zeit in Sekunden.

Um möglichst gut vergleichen zu können, werden Akkordannotationen auf Basis der Zeitachse der Audioaufnahme benötigt. Dies ist mühsame Arbeit und wird meist von Hand gemacht. Auch wird die Prozedur der Erkennung meist auf Blockbasis gemacht, was zusätzliche Quantisierung benötigt. Es kann auch möglich sein, dass das Akkordmodell der manuellen Annotation nicht mit dem gewählten Akkordmodell der Erkennung übereinstimmt. Zusammenfassend kann man also feststellen, dass nur schon die Erstellung einer Referenzannotation Probleme mit sich bringt. Zum einen können Musiker sich noch nicht einmal einig werden, wie die richtige Annotation aussieht. Zweitens kommt es darauf an, wie die Granularität der Annotation gewählt wurde. Drittens müssen die manuellen Annotationen mit den berechneten Resultaten vergleichbar gemacht werden. (vgl. ebd., 259)

Man wir schnell merken, dass nicht jedes Frame eine Annotation braucht. Wenn die Audioaufnahme beispielsweise mit Stille beginnt oder Applaus endet, so wird diesen Frames ein Symbol zugewiesen, da diese Stellen keinen wirklichen Akkord besitzen. In solch einem Fall wird das Symbol N notiert. So kann das Resultat dann blockweise mit der Referenz verglichen werden. Um den Vergleich zu quantifizieren, werden die drei Fälle TP (true positive), FP (false positive) und FN (false negative) spezifiziert. TP steht für eine Übereinstimmung von Referenz und Resultat. FP steht für eine falsche Übereinstimmung, das bedeutet wenn die Referenz einen Akkord hat aber ein falscher Akkord berechnet wurde. FN steht für den Fall, dass die Referenz keinen Akkord hat aber doch ein Akkord berechnet wurde. Mit diesen Werten können Beurteilungen wie Genauigkeit (engl. precision) als P, Trefferquote (engl. recall) als R und F-Mass (engl. F-measure) als F ausgerechnet werden. Dies sind die Definitionen:

$P = \frac{\#TP}{\#TP+\#FP}$, $R = \frac{\#TP}{\#TP+\#FN}$, $F = \frac{2 \times P \times R}{P+R}$

Bei Abbildung 2.14 sind die Fälle ersichtlich. Wir erhalten hier für P = 0.84, R = 0.79 und F = 0.82. Also stimmen die meisten Akkorde mit der Referenzannotation überein wenn man nur die gröbere Granularität berücksichtigt. (vgl. ebd., 259-260)

Eine weitere grosse Hürde ist die Akkordmehrdeutigkeit. Einige Akkorde bestehen aus mehreren gleichen Noten. Dies führt zu Problemen bei der Klassifikation. Abbildung 2.17 zeigt Beispiele solcher Mehrdeutigkeit. (vgl. ebd., 260)

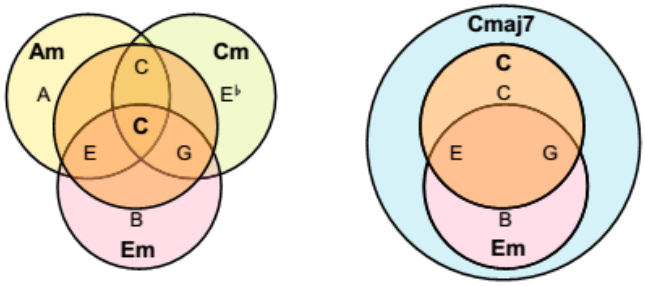


Abbildung 2.15.: Mehrdeutigkeit von Akkorden (Müller 2015, 261). Das linke Bild zeigt die gleichen Noten des Akkords C mit den Akkorden Am, Cm, und Em bei einer Klassifikation mit 24 Dur- und Moll-Akkorden. Rechts wird der Akkord Cmaj7 gezeigt, welcher aus den Noten C, E, G, und B besteht und die Akkorde C und Em beinhaltet.

Die meisten Fehlklassifikationen wurzeln in dem Problem, dass die Akkordmodelle zu stark vereinfacht wurden. Das Problem würde gelöst werden, wenn man die Liste der möglichen Akkorde erweitert, am Beispiel vom Akkord Cmaj7 würde man die Dur 7 Akkorde miteinbringen. Dies würde wiederum die Konfusionswahrscheinlichkeit bei der Klassifikationsphase erhöhen. (vgl. ebd., 261)

Auch akustische Mehrdeutigkeit ist ein grosses Problem. Eine gespielte Note auf einem Instrument ist ein komplexer Mix von Tönen. Wenn man die Noten zusammen spielt, überlagern sich die Harmonien der einzelnen Noten. Die Dur-Moll Verwirrung (engl. major-minor confusion) ist ein häufiges Phänomen in der automatisierten Akkorderkennung. Dies tritt beim Moll-Akkord Cm auf, welcher die Chromawerte C, $E^{\flat}$ und G hat. Neben der Energie in diesen drei Chromabändern, kann der akustische Ton dieses Akkords auch substanzielle Frequenzkomponenten im Chromaband E haben. Dies kann zu Verwirrung zwischen den Akkorden Cm und C führen. Wenn manche Noten leiser gespielt werden als andere, verstärkt sich das Problem noch. So kann ein Moll-Akkord als Dur-Akkord klassifiziert werden. (vgl. ebd., 261)

Eine weitere grosse Fehlerquelle in der automatisierten Akkorderkennung sind die unterschiedliche Stimmung (engl. tuning) von Instrumenten. Orchester sind manchmal unter oder über der üblichen Stimmungsfrequenz von 440 Hz. Auch kann die Stimmung aufgrund der Aufnahme verändert werden. Abbildung 2.16 zeigt dieses Problem am Beatles-Beispiel. (vgl. ebd., 262)

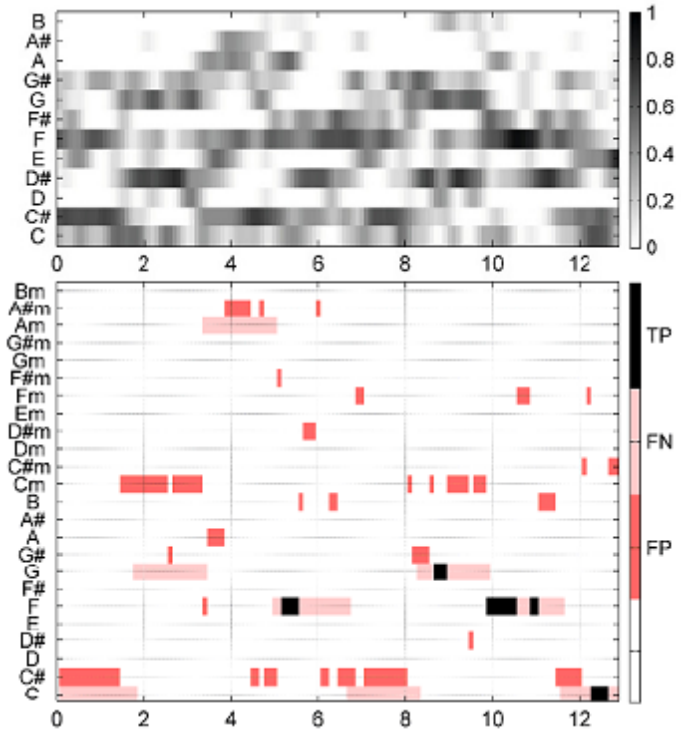


Abbildung 2.16.: Resultat einer Akkorderkennung des Beatles-Beispiels in der die Audioaufnahme um einen halben Halbton (50 Cent) höher gestimmt wurde (Müller 2015, 264). Oben die Chroma-Darstellung und unten die Akkordlabels mit ihren Klassifizierungen (TP, FN, FP) aufgrund von Referenz-Akkordlabels. Die x-Achse beschreibt die Zeit in Sekunden.

Obwohl manche Werte mit Verschieben noch übereinstimmen, sind andere Werte völlig falsch (vgl. ebd., 263).

Zuletzt gibt es noch das Problem der Unterteilungsmehrdeutigkeit. Dies kann am Beispiel des C-Dur Präludium von Johann Sebastian Bach erläutert werden (siehe Abbildung 2.17). Am Anfang startet es mit einer Bass-Note, dann erst setzen die anderen Noten ein und bauen graduell den Ton des ganzen Akkords auf. Dies ist ein sogenannter gebrochener Akkord, der eigentlich als eine einzige harmonische Einheit wahrgenommen werden kann. Um das Problem zu lösen könnte in diesem speziellen Beispiel die Blocklänge erhöht werden. Diese Problemlösung kann jedoch bei anderen Musikstücken zu Problemen führen. Eine Alternative wäre die Filterung vor dem Patternmatching. (vgl. ebd., 264)

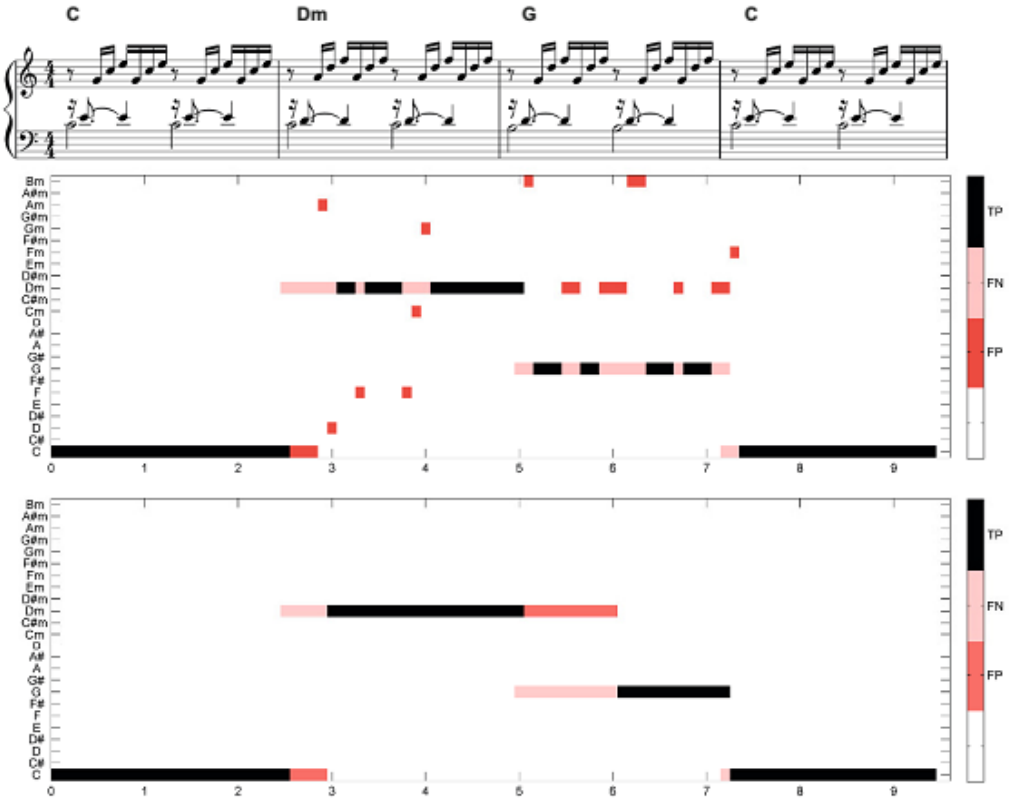


Abbildung 2.17.: Evaluation von Akkorderkennungsresultaten von der Prelude BWV 846 in D-Dur von Johann Sebastian Bach (Müller 2015, 265). Oben die Referenzannotation, in der Mitte das Erkennungsresultat bei einer Blocklänge von 200 Millisekunden und einer Hop-Grösse von einer halben Fensterlänge (Feature-Rate von 10 Hz) und unten das Resultat nach Prefiltering mit 20 Frames. Die x-Achse beschreibt die Zeit in Sekunden.

2.3.3. Verbesserungsmöglichkeiten

Die Template-basierte Methode reicht nicht, um gute Ergebnisse zu erzielen. Es gibt noch viele Unstimmigkeiten zwischen den Experten, wie musikalische Phänomene interpretiert werden können. Deshalb werden im Folgenden Verbesserungstechniken erklärt, die ausnutzen, auf was sich Musikexperten Heute einigen können.

Die erste Methode besteht darin, Templates mit Harmonien zu verwenden. In der Basismethode wurden idealisierte binäre Templates benutzt. Bei echten Audioaufnahmen können jedoch die Chromamerkmale aufgrund von Harmonien und anderen Komponenten unstrukturierter sein, also nicht binär. Zum Beispiel wird eine Note mit Chroma C genommen. Die ersten acht harmonischen Partialtöne entsprechen den Chromawerten C, C, G, C, E, G, B$^\flat$ und C. Dabei zerfällt die Energie bei den harmonischen Partialtönen oft exponentiell. Dann wäre also eine Template mit Harmonien für das Chroma C $t_C^h = (1+\alpha+\alpha^3+\alpha^7,0,0,0,\alpha^4,0,0,\alpha^2+\alpha^5,0,0,\alpha^6,0)^T$, wobei die Energie des k-ten Partialtons α^{k-1} für $\alpha \in [0,1]$, $k \in \mathbb{N}_0$. Das Akkord-Template mit Harmonien für C-Dur kann erhalten werden, indem die Templates der Chromaklassen des Akkords summiert werden $t_C^h = t_C^h + t_E^h + t_G^h$. Die Akkord-Templates können mithilfe zusätzlicher Parameter, wie der Gewichtung der Noten eines Akkords, verbessert werden. (vgl. Müller 2015, 266)

Eine nächste Methode ist, Chromamuster mithilfe bereits beschriebener Akkord-Templates von Trainingsdaten erlenen zu lassen. Dies kann mithilfe maschinellen Lernens (engl. machine learning), genauer überwachtes Lernen (engl. supervised learning), erzielt werden. Es gibt also eine Reihe von Trainingsbeispielen. Jedes Trainingsbeispiel besteht aus einem Inputobjekt und einem Outputwert. Hier sind die Inputobjekte Chromavektoren und die Outputwerte Akkordlabels. Das überwachte Lernen versucht, ein Klassifikationsschema zu erstellen, welche für undefinierte Chromavektoren richtige Akkordlabels bestimmen soll, mithilfe der bereits erlernten Akkorde im Modell. Der Vorteil dieser Methode ist, dass die erlernten Akkord-Templates die musikalischen und akustischen Begebenheiten automatisch miteinbeziehen. Der Nachteil ist aber, dass die Trainingsdaten stimmen müssen und für alle benötigten Fälle vorhanden sein müssen. Also werden sehr viele Trainingsdaten für jeden möglichen Fall benötigt. (vgl. ebd., 267-268)

Eine weitere Methode ist die Spektrale Anreicherung. Dabei werden die extrahierten Chromafeatures modifiziert. Es gibt verschiedene Chromavarianten mit unterschiedlichen Eigenschaften. Der benutzte Typ hat grossen Einfluss auf die Resultate. Eine erste Anreicherungsstrategie ist die logarithmische Kompression (engl. logarithmic compression). Diese Strategie macht die Chromaverteilung des Signals uniformer, also einheitlicher, dabei werden kleineren Komponenten eine grössere Gewichtung gegeben, relativ zu den stärkeren Komponenten. Es zeigte sich in einigen Experimenten, dass Anreicherungsmethoden, wie die logarithmische Kompression oder auch die spektrale Aufhellung (engl. spectral whitening), ein essentieller Schritt in Akkorderkennungsprozeduren sind. (vgl. ebd., 269-271)

Wenn logarithmische Anreicherung als eine Art von spektraler Glättung (engl. smoothing) angeschaut wird, so könnte man auch an eine zeitliche (engl. temporal) Glättung denken. Da Glättung immer vor dem Patternmatching angewendet wird, gehört dieser Schritt zum Prefiltering. Glättungsoperationen können gut sein, um den Effekt von lokalen irrelevanten Variationen zu vermindern. Ein Weg ist es, einen Durchschnittsfilter (engl. averaging filter) auf einzelne Komponenten der Chromafeatures anzuwenden. Auch dies ist ein guter Weg, um framebasierte Akkorderkennung zu verbessern. Die optimale Glättungslänge hängt jedoch von den Daten ab. Ein anderer Weg wäre, musikalische Aufteilung zu nutzen. Dabei könnte beispielsweise der Beat herbeigezogen werden, da Akkordwechsel meist auch mit Beatpositionen zusammenfallen. So könnte das Filtern zwischen Beats vorgenommen werden, wobei ein Fenster zwischen zwei Beats liegt. Leider ist die automatische Erkennung von Beatpositionen selbst keine einfache Aufgbe. (vgl. ebd., 271-272)

3. Konzept

3.1. System

Das System ist eine einfache Applikation ohne GUI. Die Steuerung erfolgt über die Kommandozeile, um den Arbeits- und Evaluationsprozess zu vereinfachen. Abbildung 3.1 zeigt ein mögliches Schema des Systems.

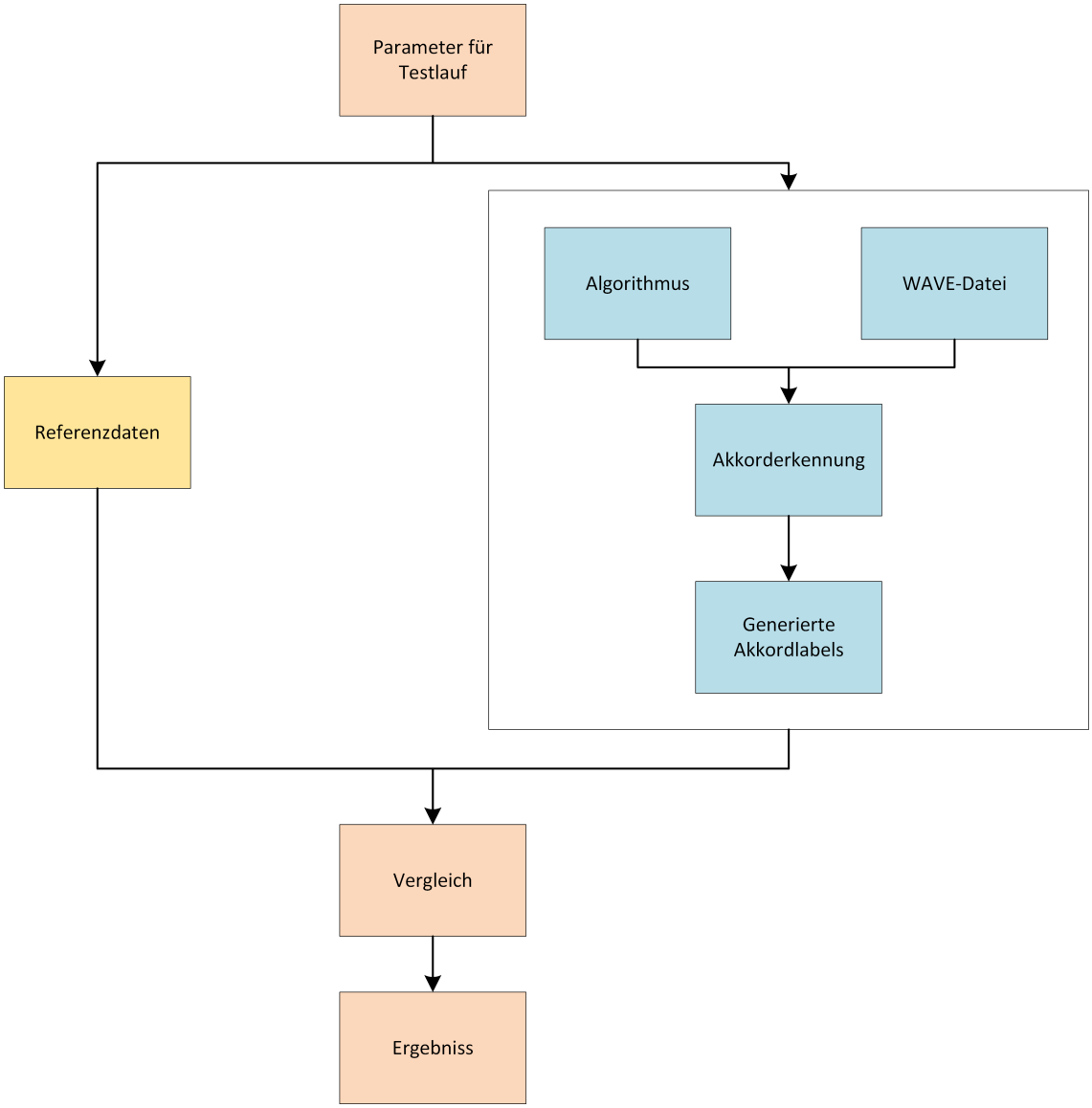


Abbildung 3.1.: Darstellung des Systems mit seinen Komponenten.

Das System verfügt über eine Datei, in der Abbildung 3.1 als Parameter für Testlauf bezeichnet, in der alle Einstellungen vorgenommen werden. Hier können die genutzten Daten und Parameter definiert und einzelne Komponenten aktiviert oder deaktiviert werden.

Ein Algorithmus wird auf eine Musikstück in Form einer Audiodatei angewendet, um eine Akkorderkennung durchzuführen. Als Resultat werden die erkannten Akkordlabels ausgegeben.

Parallel dazu werden Referenzdaten eingelesen. Diese bieten zur gewählten Audiodatei annotierte Akkordlabels, welche verwendet werden, um die Akkuratheit der durch den Algorithmus generierten Akkordlabels zu prüfen.

Die eingelesenen Referenz-Akkordlabels werden mit den generierten Akkordlabels in einem nächsten Schritt verglichen. Der Vergleichsschritt kann neben einem direkten Vergleich pro Akkord auch die prozentuale Übereinstimmung sowie quantitative Masse wie Genauigkeit, Trefferquote und F-Mass enthalten. Dieser Schritt kann beliebig nach Bedürfnissen mit Analysekomponenten erweitert werden.

Am Ende werden alle, nach Parameter und nach dem Vergleichsschritt spezifizierten, Werte und Darstellungen ausgegeben. Wichtig ist dabei eine Darstellung, welche auch manuell interpretierbar ist, um aus ihnen lernen zu können.

3.2. Referenzdaten

Ein wichtiger Teil dieser Projektarbeit ist das Finden von geeigneten Referenzdaten. Diese werden verwendet, um die Effektivität der Akkorderkennung zu prüfen. Diese können auch selbst erstellt werden, was jedoch sehr mühsam und zeitaufwändig ist.

Folgende Punkte müssen bei der Suche von Referenzdaten beachtet werden:

- Wie vertrauenswürdig sind die Referenzdaten? - Wurden sie von mehreren Personen überprüft und verifiziert? Wie qualifiziert sind diese Personen?
- Steht das passende Musikstück im richtigen Dateiformat zur Verfügung? - Es gibt unterschiedliche Aufnahmen des gleichen Musikstücks. Dies kann zur Folge haben, dass die Zeitangaben der Referenzdaten nicht mit der Audiodatei übereinstimmen.
- Haben die Referenzdaten alle nötigen Informationen? - In diesem Fall wann welcher Akkord spielt. Die Referenzdaten können auch andere Informationen wie beispielsweise die Tonhöhe enthalten.
- Sind die Referenzdaten in einem geeigneten Format, um sie einlesen zu können? - Eine klare und einfache Struktur, die leicht eingelesen werden kann.

Isophonics ist eine Webseite, auf der Software sowie Ressourcen des Centre of Digital Music (C4DM) der Queen Mary Universität London, angeboten werden. Sie beschäftigen sich unter Anderem selbst mit Music Information Retrieval (MIR) und bieten somit gute Referenzdaten sowie hilfreiche Software dazu an. Die verwendeten Referenzdaten stammen alle von dort. Abbildung 3.2 zeigt eine Darstellung von Akkordannotationen. Im gleichen Paket gibt es auch Annotationen zu Songsegmenten wie Refrain, Instrumentalsolo und Tonart. Einfachheit halber wurden nur die Akkordannotationen genutzt.

Die Referenz-Annotationen für die Greatest Hits Alben von Queen sind hier erhältlich:
http://isophonics.org/content/reference-annotations-queen

```
0.000   0.259   N
0.259   1.648   N
1.648   3.601   C:min
3.601   5.517   Bb/2
5.517   7.396   C:min
7.396   9.335   Bb/2
9.335   11.244  C:min
11.244  13.144  Bb/2
13.144  15.061  C:min
15.061  17.016  Bb/2
17.016  18.978  Eb
18.978  20.867  Ab/5
20.867  22.841  Eb
22.841  24.727  Ab/5
24.727  25.693  Eb
25.693  26.660  Bb/3
26.660  27.600  C:min
27.600  28.596  F:7
28.596  30.531  Bb
30.531  33.354  Bb:maj(9)
```

Abbildung 3.2.: Illustration (C4DM 2010) der ersten 20 Akkorde im Isophonics-Datensatz von Queens We Are The Champions (Greatest Hits I, Datei: 17 We Are The Champions.lab). Die erste Spalte beschreibt die Startzeit in Sekunden, die zweite Spalte die Endzeit in Sekunden und die dritte Spalte das annotierte Akkordlabel. N wird als Zeichen genutzt, um Stille oder "kein Akkord" zu markieren.

Mithilfe des Sonic Visualizers (C4DM 2010) können die Isophonics-Datensätze mit den passenden Audiodateien zeitlich auf einem Strahl mit den zugehörigen Frequenzdarstellungen sowie den gegebenen Referenzannotationen angezeigt werden (siehe Abbildung 3.3). Praktisch an diesem Tool ist, dass beim Abspielen des Musikstücks Akkordwechsel mit einem kurzen Tick markiert werden. So kann mit etwas Musikgehör gut erkannt werden, ob die Referenz-Annotationen zur Audiodatei passen und zeitlich übereinstimmen.

Der Sonic Visualizer ist hier erhältlich: http://sonicvisualiser.org/

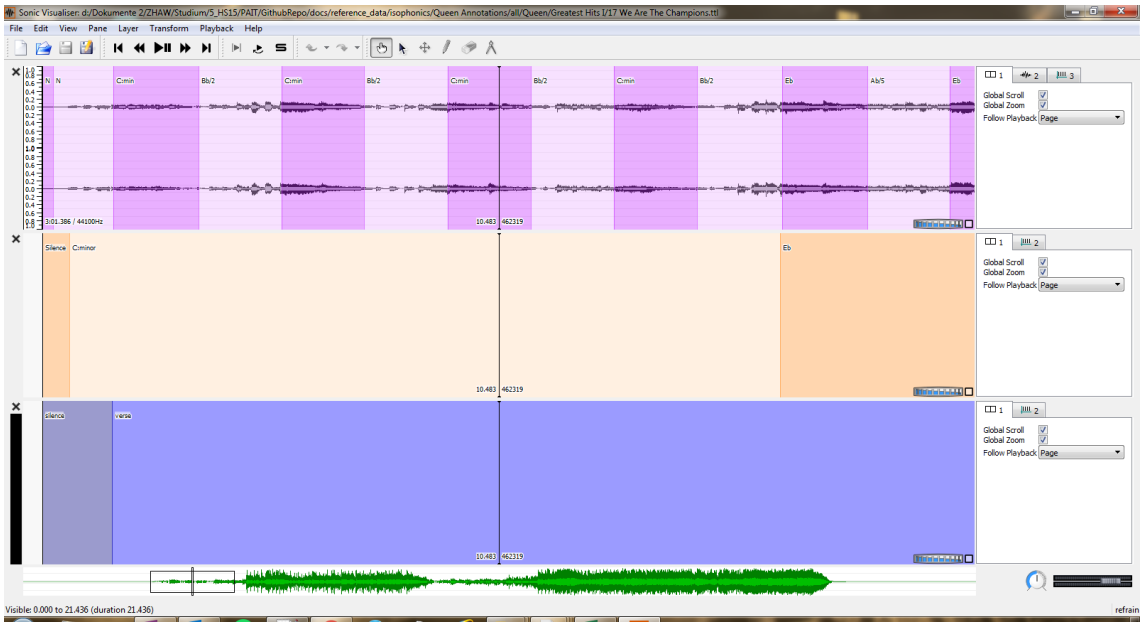


Abbildung 3.3.: Screenshot des Sonic Visualizers (C4DM 2010) mit den ersten Akkorden im Isophonics-Datensatz von Queen's We Are The Champions. Zuoberst sind die Frequenzdarstellungen der Audiosignale beider Seiten (links und rechts da es ein Stereosignal ist). In der gleichen Zeile sind die Akkordannotationen, welche am oberen Rand stehen und mit Trennstrichen abgegrenzt sind. In der Mitte sind die Annotationen zur Tonart und unten die Annotationen der Songsegmente. Ganz unten befindet sich eine Zeitleiste mit einer Frequenzdarstellung für das ganze Audiosignal, wobei die Länge der Audiodatei ersichtlich ist. Die vertikale Gerade in der Mitte markiert die aktuelle Abspielposition.

Die Webseite bietet Akkordannotationen zu einigen Alben der folgenden Bands: Beatles, Queen, Zweieck und Carole King. Die Beatles-Akkordannotationen sind laut Webseite die zuverlässigsten und wurden von mehreren Personen überprüft und verifiziert. Die Queen-Akkordannotationen wurden nur von einer Person überprüft und werden von der Seite selbst als nicht ganz zuverlässig eingestuft. Dies gilt auch für die Zweieck-Akkordannotationen. Die Carole King-Akkordannotationen wurden laut Webseite nicht sorgfältig überprüft.

Diese Arbeit ist auf die Queen-Datensätze beschränkt, da die zugehörigen Audiodateien zur Verfügung stehen. Alle in dieser Projektarbeit eingesetzten Referenzdaten sowie die Audiodateien befinden sich auf dem Datenträger im Anhang.

4. Umsetzung

4.1. Aufbau des Systems

Wie bereits im Konzept beschrieben, ist das System modular aufgebaut. Durch die klare Gliederung bleibt das System übersichtlich und einzelne Komponenten können unabhängig voneinander optimiert werden. Jeder Teilschritt übernimmt die Daten vom letzten Schritt, verarbeitet diese und schreibt die gewonnenen Informationen in eine Textdatei. Die Zwischenresultate können so jederzeit verglichen werden. Zwei grosse Nachteile hat diese Variante aber. Das System benötigt lange Verarbeitungszeiten und der Informationsgehalt, welcher weitergegeben werden kann, ist beschränkt.
Es wurde als Programmiersprache Python verwendet, da hier bereits viele nützliche Bibliotheken mit Funktionen zur Audio-Analyse vorhanden sind und alle nötigen mathematischen Funktionen enthalten sind. Zudem handelt es sich um eine Interpreter-Sprache und kann ohne Neukompilierung auf jedem Betriebssystem ausgeführt werden.

Das System besteht zum Zeitpunkt der Abgabe aus folgenden Komponenten:

- Einlesen von WAVE-Files und generieren eines Spektrograms um die Frequenzen zu extrahieren
- Umwandeln der Frequenzen in MIDI-Noten
- Töne den Akkorden zuordnen
- Referenzdaten für den Vergleich vorbereiten
- Framebasierter Vergleich der Akkordlabels

Abbildung 4.1.: Übersicht der einzelnen Komponenten des Systems. Die einzelnen Funktionen sind in eigene Dateien aufgeteilt.

Die Konfiguration wird zentral in der Datei main.py durchgeführt. Neben dem Selektieren der zu durchlaufenden Komponenten können auch spezifische Parameter angepasst werden. Durch Aufruf von main.py werden die angegebenen Dateien aus der Datei files.txt gelesen und in den einzelnen Komponenten weiterverarbeitet.
Alle mitgegebenen Daten, wie WAVE- und Referendaten, werden im Ordner files.txt abgelegt. Der Dateiname muss jeweils übereinstimmen. Nur die Dateiendung wird für die Unterscheidung verwendet.
Im Ordner data_output werden zwischen- und Endergebnisse gespeichert.
Folgende Dateien können während der Verarbeitung generiert werden:

- FileName-frequency.txt (Auflistung der Frequenzen. 1 Frame pro Zeile, durch Tab getrennt)
- FileName-amp.txt (Amplituden der gefundenen Frequenzen)
- FileName-notes.txt (Auflistung der zugeordneten Frequenzen)
- FileName-chords.txt (Auflistung der zugeordneten Akkorde)
- FileName-result.txt (Framebasierter Vergleich der Labeldaten)
- Plot des Spektrograms und des physischen Audio-Signals
- Referenzdaten im umgewandelten Format (anstelle von Zeitabschnitten ein Label für jedes Frame)

4.1.1. Frequenzerkennung

Die framebasierte Auswertung der Frequenzen und der dazugehörigen Amplituden wird mit der Funktion "specgram" aus der Library "pylab" (Hunter/Dale/Firing/Droettboom et al. 2014) gemacht. Specgram generiert ein fertiges Spektrogram in Form eines zweidimensionalen Arrays, welches weiterverarbeitet werden kann. Alle nötigen Parameter für die STFT können direkt mitegegeben werden.

Laut Müller (Müller 2015, 55-56) wurden die Parameter so gewählt: Fensterlänge: 4096 Samples; Schrittweite: 2048 (Entspricht 50%). Dies ergibt eine Frequenzauflösung von 10.8 Hz und einer Abtastbreite von 46.4 ms. Diese Genauigkeit reicht laut Müller (ebd., 56) aus.

Als Fensterfunktion wurde Hanning eingesetzt.

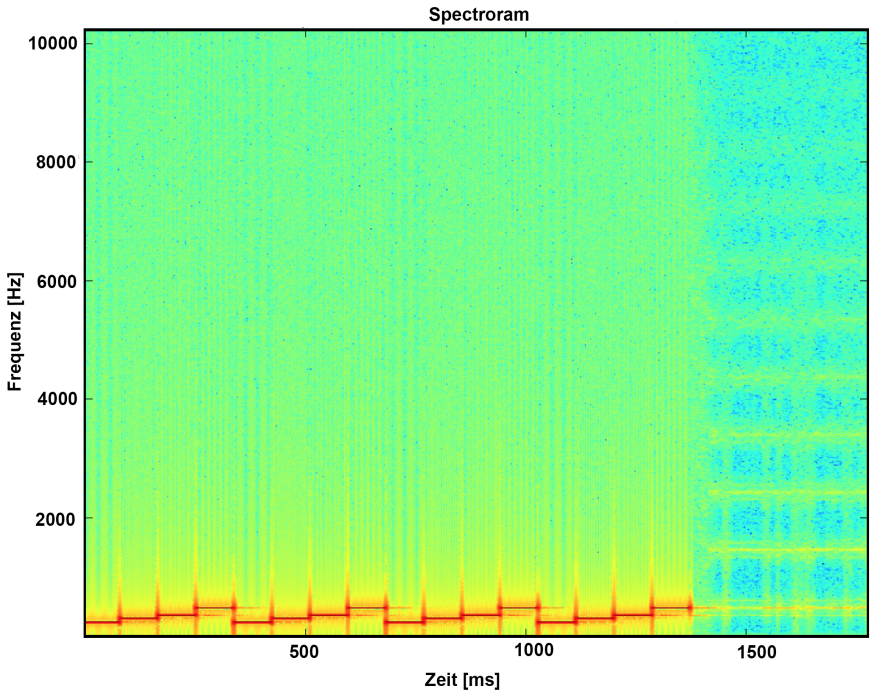


Abbildung 4.2.: Plot der Specgram Daten anhand eines einfachen Audio-Beispiels

4.1.2. Töne zuordnen

Die Frequenzen werden mit der Formel $n = runden(12 * log2(fn/440Hz))$ (Wolfe o. J.) in die jeweiligen MIDI-Notennummer umgewandelt .

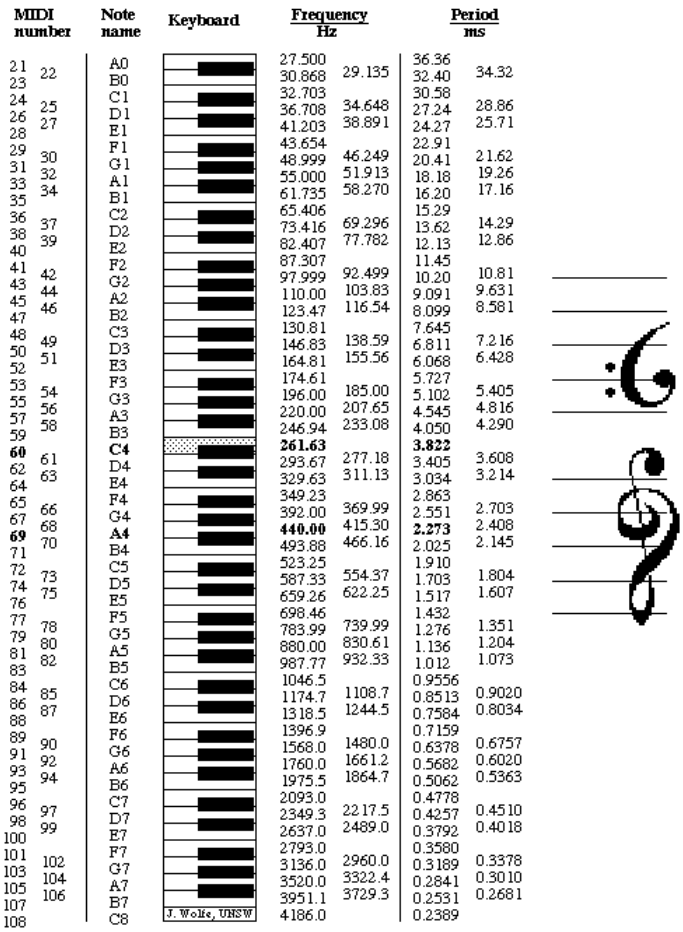


Abbildung 4.3.: Übersicht der einzelnen Komponenten des Systems (Wolfe o. J.). Die einzelnen Funktionen sind in eigene Dateien aufgeteilt.

Die Noten sind nun Nummern, die, wie in Abbildung 4.2 zu sehen ist, eindeutig einer Frequenz und einem Notennamen zugeordnet werden können. Dies wurde so gewählt, damit nahe beieinander liegende Frequenzen automatisch dem richtigen Notennamen zugeordnet werden. Ausserdem kann anschliessend die Akkordanalyse vereinfacht werden.

4.1.3. Akkordanalyse

Die unterschiedlichen Akkordtypen sind durch die Abstände der Noten definiert. (Piano Chord Dictionary.com 2010)

Durch die vorherige Konvertierung von Frequenzen in MIDI-Notennummern, müssen die Akkorde nun ebenfalls mit dieser Repräsentationsform kompatibel sein. Wie in Abbildung 4.4 zu sehen ist, können die musikalischen Formeln auf Abstände zwischen den MIDI-Noten abgebildet werden. Anhand der Abstände zwischen den Noten werden nun extrahierte Frames des Musikstücks mit den Abstandsmerkmalen der unterschiedlichen Akkordtypen verglichen.

Auf eine vertiefte Analyse, welche weitere Akkorde mit mehr als drei Noten berücksichtigt, wurde im Rahmen dieser Arbeit verzichtet.

Akkord Typ	Töne	Musikalische Formel	Schritte in MIDI
Major	C-E-G	1 3 5	4 3
Major 1. Inversion	E-G-C	3 5 1	3 5
Major 2. Inversion	G-C-E	5 1 3	5 4
Minor	C-Eb-G	1 3b 5	3 4
Minor 1. Inversion	Eb-G.C	3b 5 1	3 5
Minor 2. Inversion	G-C-Eb	5 1 3b	5 3
Augmented	C-E-G#	1 3 5#	4 4
Augmented 1. Inversion	E-G#-C	3 5# 1	4 4
Augmented 2. Inversion	G#-C-E	5# 1 3	4 4
Diminished	C-Eb-Gb	1 3b 5b	3 3
Diminished 1. Inversion	Eb-Gb-C	3b 5b 1	3 6
Diminished 2. Inversion	Gb-C-Eb	5b 1 3b	6 3

Abbildung 4.4.: Darstellung von Akkorden und deren Informationen. Zu sehen sind die Töne der Akkorde, die musikalische Formel (Piano Chord Dictionary.com 2010) und die Abstände der MIDI-Repräsentation. Es handelt sich um einen C-Dreiklang.

4.2. Ueberprüfen der Ergebnisse

Für die MIREX (IMIRSEL 2015) haben Pauwels und Peeters im Jahre 2013 eine Software entwickelt, welche Akkorderkennungs-Algorithmen auf ihre Genauigkeit prüft.

Hier eklären sie, wie die Software erstellt wurde:
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=6637748

Sie berechnet den CSR (engl. chord symbol recall), der Prozentwert der Übereinstimmung mit den Referenzdaten, welcher sich wie folgt berechnen lässt:

$CSR = \frac{totaldurationofsegmentswhereannotationequalsestimation}{totaldurationofannotatedsegments}$ (IMIRSEL 2015)

Die Auswertung wird dabei für die unterschiedlichen Akkord-Typen einzeln erstellt.

Der Quelltext steht unter folgender URL zur Verfügung: https://github.com/jpauwels/MusOOEvaluator

Aus technischen Gründen war die Installation dieser Software während der Arbeit nicht möglich. Dadurch wurde ein eigenes System zusammengestellt, welches die Daten nicht mit Zeitbereichen, sondern die einzelnen Frames vergleicht.

Dies hat den Vorteil, dass es einfach gelöst werden kann. Die Referenzdaten sind schnell ins neue Format umgewandelt und die Durchsicht der Übereinstimmung, ohne die Zeit zu prüfen, ist auch von Hand noch möglich.

Unser System macht aber keine Unterscheidung für als Statistik (Dur, Moll, etc.). Es gibt nur folgende Werte: Akkord stimmt (TP), Akkord stimmt nicht (FP) und es ist kein Akkord vorhanden (FN).

5. Ergebnisse

Es hat sich herausgestellt, dass die reine Extraktion der Frequenzen und anschliessende Akkord-Zuordnung mit Template-basierter Methode und Zuhilfenahme von MIDI nicht oder nur sehr bedingt auf reale aufgenommene Pop- und Rockmusik angewendet werden kann.

Einfache Audiodateien mit reinen Sinus-Schwingungen werden mit einer sehr hohen ($> 90\%$) Genauigkeit erkannt. Bei den verwendeten Aufnahmen von Queen konnte nur eine sehr geringe Übereinstimmung ($< 20\%$) erzielt werden. Die meisten Übereinstimmungen wurden dort erzielt, wo jeweils keine Musik gespielt wird und dementsprechend auch kein Akkord erkannt werden konnte.

Zur Veranschaulichung ist in Abbildung 5.1 ein Screenshot der Software "Overtone Analyzer" der Firma Sygyt (Sygyt Software 2015) zu sehen.

Die Software ist hier erhältlich: http://www.sygyt.com/de/overtone-analyzer

Die Akkordlabels der Isophonics-Referenzdatensätze wurden mit Frequenz-Markern eingetragen. Es ist ersichtlich, dass bereits in diesem kurzen Abschnitt von ca. 8 Sekunden viele Lücken oder ungenaue Frequenzbilder vorhanden sind. Wird auf dieses File ein Algorithmus angewendet, wie er in dieser Arbeit implementiert wurde, so werden sehr oft falsche oder keine Akkorde erkannt, was zu einem schlechten Ergebnis führt.

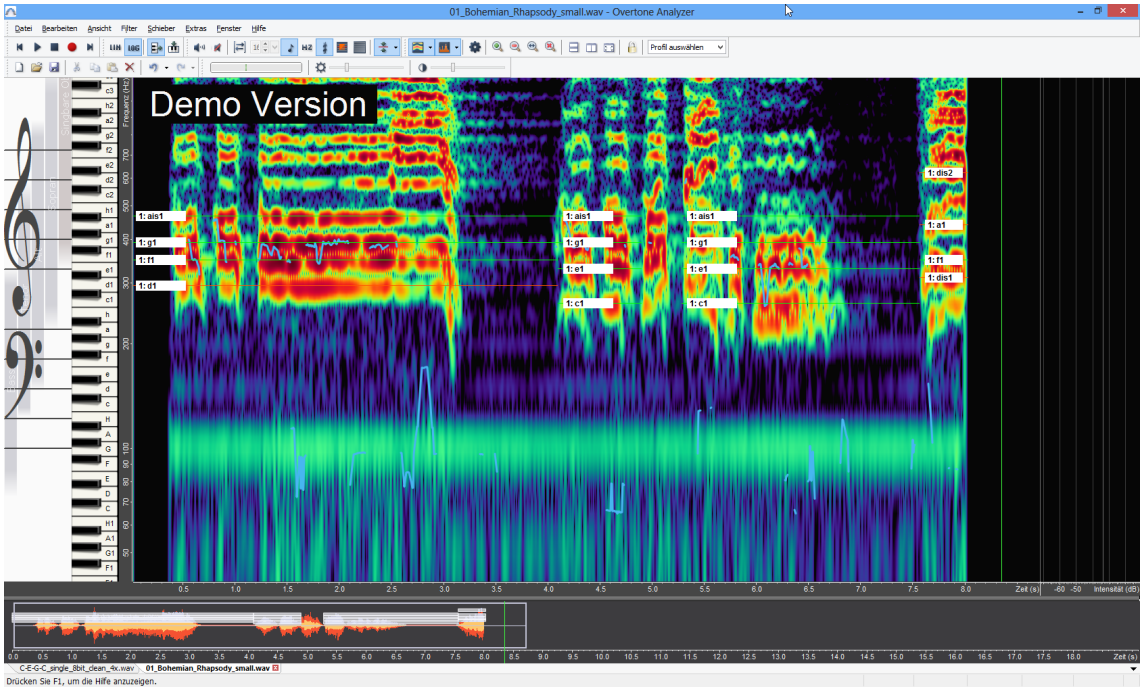


Abbildung 5.1.: Mit dem "Overtone Analyzer" generiertes Spektrogram mit eingezeichneten Frequenzmarkern.

Da die Software von Pauwels und Peeters zur Evaluation der Resultate nicht installiert werden konnte, ist ein direkter Vergleich mit anderen Algorithmen, welche bei MIREX (IMIRSEL 2015) eingereicht wurden, nicht möglich.

Bei einem Blick auf die Teilnahmen des MIREX Wettbewerbs für Akkorderkennung von 2013 (IMIRSEL 2015) sieht man, dass ganz andere Ansätze genutzt wurden als in dieser Arbeit.

Der Ansatz, der nach MIREX die beste Übereinstimmung erzielt hat (Cho/Bello 2010), ist vortrainiert und benutzt dazu noch einen Multistrom-HMM (Hidden Markov Modell). Bedenklich ist die Tatsache, dass die Trainingsdaten bereits Queen und Beatles Daten enthalten. Dies macht die Testdaten zu den Wettbewerbsdaten sehr ähnlich. Interessant wäre zu wissen, wie gut die Methode mit anderer Popmusik funktioniert.

Der zweite Ansatz (Khadkevich/Omologo 2011) beruht auf Zeit-Frequenz-Neuzuordnung (engl. time-frequency reassignment, TFR). Dabei geht es darum, die spektrale Energie jeder Spektrogrammzelle neu zu belegen in eine andere Zelle, die näher zur echten unterstützten Region des analysierten Signals ist. Dies führt dazu, dass die unscharfe spektrale Repräsentation wieder genauer wird, um so die spektralen Eigenschaften mit höherer Zeit und Frequenzauflösung zu erlangen.

Der nächste Ansatz (Ni/Mcvicar/Santors-Rodriguez/De Bie 2013) nutzt maschinelles Lernen und Harmonien. Die genutzte Software erkennt neben Akkorden auch noch Tonart und Bassnoten. Das System nutzt verbesserte Chromagramme, die die menschliche Wahrnehmung von Lautheit miteinbeziehen. Der Ansatz nutzt zusammen mit den Chromagrammen nur maschinelles Lernen.

Ein anderer Ansatz (Glazyrin 2013) nutzt eine Kombination mehrerer Methoden. Es wird ein Spektrogramm mit Taktinformation mit hoher Zeit- und Frequenzauflösung berechnet. Die Chromavektoren des Spektrogramms werden mithilfe einer sogenannten “self-similarity“ Matrix geglättet, bevor die Akkorderkennung durchgeführt wird. Zusätzlich werden binäre Akkodtemplates mit drei harmonien genutzt. Auch werden zwei Heuristiken verwendet, um die Dur-Moll-Verwechslung (engl. major-minor confusion) zu vermindern und Ein-Ton-Akkorde auszuschliessen.

Ein weiterer Ansatz (Pauwels/Geoffroy 2013) hat eine Phase der Feature-Extraktion, eine Glättungsphase und ein probabilistisches Modell.

Der Ansatz (Steenbergen/Burgoyne 2010), der beim Wettbewerb am schlechstesten abgeschnitten hat, nutzt HMM und neuronale Netze. Das neuronale Netz ist trainiert auf die Approximation von Tonklassen-Profilen. Auch wird HMM genutzt, um die Akkorde zu klassifizieren. Dabei werden beide Teile separat trainiert und dann zusammengeführt um zu optimieren.

6. Fazit

Wie bereits im Kapitel Ergebnisse beschrieben, kann der in dieser Arbeit entwickelte Ansatz, nicht mit realen Aufnahmen von Pop- und Rockmusik umgehen. Lediglich bei elektronisch generierten Audiodateien werden Akkordfolgen mit hoher Genauigkeit erkannt. Deshalb wird an dieser Stelle von dem hier entwickelten Ansatz, für reale Popmusik, abgeraten.

Ausgeklügeltere Systeme verwenden neben einfachen Akkord-Templates auch mehr Verbesserungsmöglichkeiten, die in der Theorie erklärt werden. Diese könnten die Effektivität der Akkorderkennung unter Umständen noch verbessern. Es wird jedoch immer die Limitierung durch die Templates bleiben. Deshalb ist der Ansatz eher im Mittelmass, wenn keine andere zusätzliche Methode angewandt wird.

Da funktioniert die Akkorderkennung mithilfe von Hidden Markov Modellen (HMM) schon besser. Diese Modelle werden bereits bei der Spracherkennung (engl. speech recognition) und auch in der Akkorderkennung oft benutzt. Sie beziehen die vergangenen Frames mit ein, statt nur das aktuelle Frame anzusehen. Diese Methode nutzt die Begebenheiten von Akkord-Progressionen aus. Die Hidden Markov Modelle bringen kontextsensitive Informationen mit ein, um den nächsten Akkord genauer zu erkennen. (Müller 2015, 273) Aus dem MIREX Wettbewerb geht hervor, dass HMM nie alleine angewendet wurde. Es gab einige Ansätze, die gute Resultate brachten und zum Teil aus HMM bestanden.

Auch vorstellbar wäre eine Variante mit maschinellem Lernen. Beim MIREX Wettbewerb gab es auch ein einige gute Ansätze mit maschinellem Lernen, die Methoden mit maschinellem Lernen waren sogar unter den besten der Ansätze. Bei diesem Ansatz werden immer sehr viele Trainingsdaten benötigt. Genug Trainingsdaten zu bekommen, damit die Akkorderkennung mit vielen verschiedenen Popmusikstücken zurechtkommt, ist jedoch keine leichte Angelegenheit. Dieser Ansatz kann jedoch mit einem oder mehreren anderen Ansätzen kombiniert werden, um das Problem mit den fehlenden Trainingsdaten auszuhebeln. Dies wird auch beim MIREX Wettbewerb angewendet.

Zuletzt gab es beim MIREX Wettbewerb noch den Ansatz mit der Zeit-Frequenz-Neuzuordnung. Dieser Ansatz wurde bei der Recherche zu dieser Arbeit erst am Ende durch den MIREX Wettbewerb gefunden. Es wäre interessant, diesen Ansatz noch genauer zu untersuchen.

7. Verzeichnisse

Quellenverzeichnis

Centre for Digital Music (C4DM) Queen Mary University of London (2010): *isophonics.*
URL: http://isophonics.org/ [Stand: 16.12.2015]

Cho, Teamin / Bello, Juan P. (2010): *MIREX 2013: LARGE VOCABULARY CHORD RECOGNITION SYSTEM USING MULTI-BAND FEATURES AND A MULTI-STREAM HMM.*
URL: http://www.music-ir.org/mirex/abstracts/2013/CB3.pdf [Stand: 16.12.2015]

Glazyrin, Nikolay (2013): *AUDIO CHORD ESTIMATION USING CHROMA REDUCED SPECTROGRAM AND SELF-SIMILARITY.*
URL: http://www.music-ir.org/mirex/abstracts/2013/NG1.pdf

Gorski, Markus (o. J.): *Herzlich willkommen bei LEHRKLAENGE.de, dem Online-Lehrgang für Musiktheorie!.*
URL: http://www.lehrklaenge.de/ [Stand: 16.12.2015]

Hunter, John / Dale, Darren / Firing, Eric / Droettboom, Michael et al. (2014): *Matplotlib.*
URL: http://matplotlib.org/api/mlab_api.html [Stand: 16.12.2015]

International Music Information Retrieval Systems Evaluation Laboratory (IMIRSEL) (2015): *MIREX HOME.*
URL: http://www.music-ir.org/mirex/wiki/MIREX_HOME [Stand: 16.12.2015]

Kaiser-Kaplaner, Johannes (o. J.): *Musiklehre ONLINE.*
URL: http://www.musiklehre.at/ [Stand: 16.12.2015]

Khadkevich, Maksim / Omologo, Maurizio (2011): *TIME-FREQUENCY REASSIGNED FEATURES FOR AUTOMATIC CHORD RECOGNITION.*
URL: http://www.music-ir.org/mirex/abstracts/2013/KO1.pdf

Müller, Meinard (2015): *Fundamentals of Music Processing. Audio, Analysis, Algorithms, Applications.*
Cham, Heidelberg, New York, Dordrecht, London: Springer.

Ni, Yizhao / Mcvicar, Matt / Santos-Rodriguez, Raul / De Bie, Tijl (2013): *HARMONY PROGRESSION ANALYZER FOR MIREX 2013.*
URL: http://www.music-ir.org/mirex/abstracts/2013/NMSD1.pdf [Stand: 16.12.2015]

Pauwels, Johan / Peeters, Geoffroy (2013): *THE IRCAMKEYCHORD SUBMISSION FOR MIREX 2013.*
URL: http://www.music-ir.org/mirex/abstracts/2013/PP3.pdf [Stand: 16.12.2015]

Piano Chord Dictionary.com (2010): *Piano Chord Dictionary Online Piano Chords.*
URL: http://www.pianochorddictionary.com/ [Stand: 16.12.2015]

Steenbergen, Nikolaas / Burgoyne, John Ashley (2010): *JOINT OPTIMIZATION OF AN HIDDEN MARKOV MODEL - NEURAL NETWORK HYBRID FOR CHORD ESTIMATION.*
URL: http://www.music-ir.org/mirex/abstracts/2013/SB8.pdf [Stand: 16.12.2015]

Sygyt Software (2015): *Overtone Analyzer.*
URL: http://www.sygyt.com/de/overtone-analyzer [Stand: 16.12.2015]

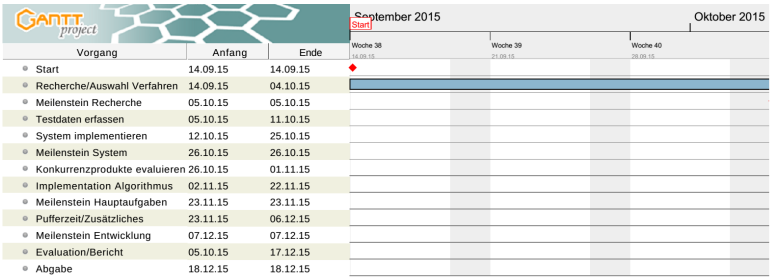
Wolfe, Joe (o. J.): *Note names, MIDI numbers and frequencies.*
URL: https://newt.phys.unsw.edu.au/jw/notes.html [Stand: 16.12.2015]

Abbildungsverzeichnis

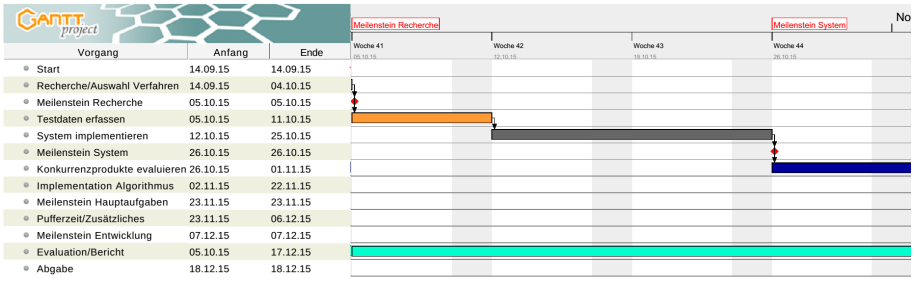
A. Anhang

A.1. Projektmanagement

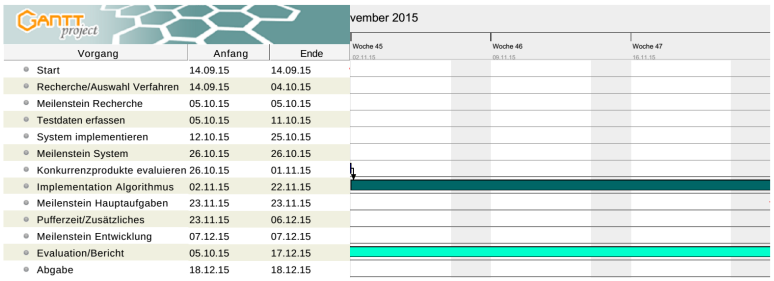
Dies ist der Zeitplan als Gantt-Diagramm, welcher in der ersten Woche der Projektarbeit erstellt wurde.



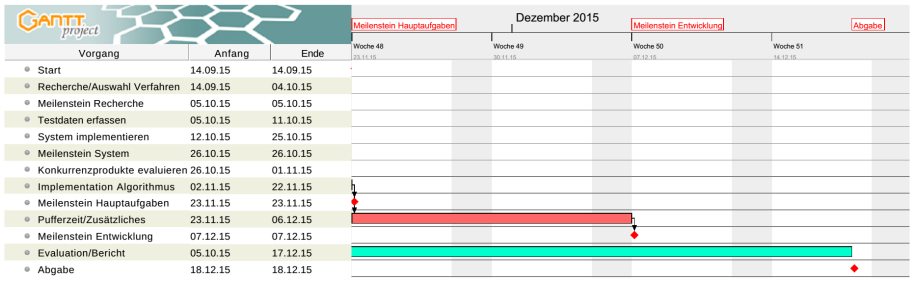
Vorgang	Anfang	Ende
Start	14.09.15	14.09.15
Recherche/Auswahl Verfahren	14.09.15	04.10.15
Meilenstein Recherche	05.10.15	05.10.15
Testdaten erfassen	05.10.15	11.10.15
System implementieren	12.10.15	25.10.15
Meilenstein System	26.10.15	26.10.15
Konkurrenzprodukte evaluieren	26.10.15	01.11.15
Implementation Algorithmus	02.11.15	22.11.15
Meilenstein Hauptaufgaben	23.11.15	23.11.15
Pufferzeit/Zusätzliches	23.11.15	06.12.15
Meilenstein Entwicklung	07.12.15	07.12.15
Evaluation/Bericht	05.10.15	17.12.15
Abgabe	18.12.15	18.12.15



Vorgang	Anfang	Ende
Start	14.09.15	14.09.15
Recherche/Auswahl Verfahren	14.09.15	04.10.15
Meilenstein Recherche	05.10.15	05.10.15
Testdaten erfassen	05.10.15	11.10.15
System implementieren	12.10.15	25.10.15
Meilenstein System	26.10.15	26.10.15
Konkurrenzprodukte evaluieren	26.10.15	01.11.15
Implementation Algorithmus	02.11.15	22.11.15
Meilenstein Hauptaufgaben	23.11.15	23.11.15
Pufferzeit/Zusätzliches	23.11.15	06.12.15
Meilenstein Entwicklung	07.12.15	07.12.15
Evaluation/Bericht	05.10.15	17.12.15
Abgabe	18.12.15	18.12.15



Vorgang	Anfang	Ende
Start	14.09.15	14.09.15
Recherche/Auswahl Verfahren	14.09.15	04.10.15
Meilenstein Recherche	05.10.15	05.10.15
Testdaten erfassen	05.10.15	11.10.15
System implementieren	12.10.15	25.10.15
Meilenstein System	26.10.15	26.10.15
Konkurrenzprodukte evaluieren	26.10.15	01.11.15
Implementation Algorithmus	02.11.15	22.11.15
Meilenstein Hauptaufgaben	23.11.15	23.11.15
Pufferzeit/Zusätzliches	23.11.15	06.12.15
Meilenstein Entwicklung	07.12.15	07.12.15
Evaluation/Bericht	05.10.15	17.12.15
Abgabe	18.12.15	18.12.15



Vorgang	Anfang	Ende
Start	14.09.15	14.09.15
Recherche/Auswahl Verfahren	14.09.15	04.10.15
Meilenstein Recherche	05.10.15	05.10.15
Testdaten erfassen	05.10.15	11.10.15
System implementieren	12.10.15	25.10.15
Meilenstein System	26.10.15	26.10.15
Konkurrenzprodukte evaluieren	26.10.15	01.11.15
Implementation Algorithmus	02.11.15	22.11.15
Meilenstein Hauptaufgaben	23.11.15	23.11.15
Pufferzeit/Zusätzliches	23.11.15	06.12.15
Meilenstein Entwicklung	07.12.15	07.12.15
Evaluation/Bericht	05.10.15	17.12.15
Abgabe	18.12.15	18.12.15

A.2. Inhalt Datenträger

Bericht

- PDF fertig zum Druck
- LaTeX-Projekt (UTF8-Codierung)

Unser System

Python Skripte zur Akkorderkennung und Auswertung

Test-Audio-Files generierung

Matlab-Skript, welches kurze Audio-Files mit Sinus-Akkorden erstellt.
Einstellungen vornehmen und starten mit dem File "ZZ_generate.m"

Isophonics Referenzdaten

- Queen Daten von den Alben Greatest Hits I und II
(nur Akkordannotationen)

Musikfiles

- Queen Greates Hits I und II
- Kurze Files, welche Sinus Abfolgen mit C, E, G und C2 beinhalten
- von Referenzdatengenerierung
(alle Dateien im WAVE-Format)

Literatur

- Fundamentals of Music Processing - Meinard Müller als PDF

Zürcher Hochschule
für Angewandte Wissenschaften

School of Engineering

Automatische Erkennung der Akkordfolge in Popmusik

PA15_stdm_4

BetreuerInnen:	Thilo Stadelmann, stdm Sigisbert Wyrsch, wyrs
Fachgebiete:	Datenanalyse (DA) Software (SOW)
Studiengang:	ET / IT
Zuordnung:	Institut für angewandte Informationstechnologie (InIT)
Interne Partner:	Zentrum für Signalverarbeitung und Nachrichtentechnik (ZSN)
Gruppengrösse:	2

Kurzbeschreibung:
Selbst Hobbymusiker benötigen zum Spielen eines aktuellen Popsongs keine Noten: Kenntnis der Melodie sowie ein Blatt Papier mit Text und Akkorden reicht, um ein Lied zu singen und etwa auf der Gitarre zu begleiten. Ein solches "Lead Sheet" zu erstellen, ist aber je nach musikalischem Können mühsam - und könnte automatisert werden!

Ziel dieser PA ist es, einen Softwareprototyp zu entwickeln, welcher als Eingabe ein Musikstück (zum Beispiel als WAV oder MP3 Datei) entgegennimmt, und die Folge der Akkorde im Stück als Buchstabenfolge ausgibt (1 Buchstabe pro Akkordwechsel). Hierzu existieren Methoden in den Bereichen Audio Processings und Machine Learning sowie entsprechende Softwarebibliotheken:

Aus dem Audiostrom können geeignete Merkmale extrahiert werden, die es einem nachgeschalteten Algorithmus ermöglichen, zu erkennen, ob z.B. ein "D-Dur" Akkord gespielt wurde. In dieser Arbeit sollen die geeignetsten Verfahren zusammen mit den Betreuern ausgewählt, implementiert und experimentell evaluiert werden.

Inhalt:

- Kennenlernen des Hintergrunds von Audio Processing und Music Analysis
- Auswahl eines Verfahrens zur Erkennung der Akkordfolge (melodische Struktur) in Popsongs anhand der wissenschaftlichen Literatur
- Implementierung des Verfahrens (z.B. in Matlab, Python oder Java)
- Finden oder Zusammenstellen geeigneter Daten zum Evlauieren und ggf. Trainieren des Systems (Musik und dazugehörige Akkordannotationen)
- Systematische Evaluation des Ansatzes anhand der Daten, ggf. Anpassungen an Code, Parametern etc.
- Evaluation und Präsentation der Ergebnise (Bericht, Demo)

Voraussetzungen:
Die Betreuer haben langjährige Erfahrung mit Audio Processing und stehen mit Rat, Tat und viel Einsatz zu Seite. Aus Machine Learning und Audio Processing Aufgabenstellungen sind in den letzten Semestern bereits mehrere hervorragende studentische Arbeiten hervorgegangen. Bei Interesse ist eine Fortsetzung in BA und Masterstudium möglich.

Vorkenntnisse in Audio Processing, Machine Learning, Musik etc. sind nicht Voraussetzung. Das nötige Anwenderwissen wird im Rahmen dieser PA vermittelt und erarbeitet. Wir erwarten lediglich Freude am Experimentieren und Programmieren.

Weiterführende Informationen:
https://dublin.zhaw.ch/~stdm/?page_id=77