

Genre-Basierte Automatische Komposition von MIDI-Daten

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medieninformatik

eingereicht von

Thomas Adelbauer, BSc

Matrikelnummer 0626593

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung
Betreuer: Ao. Univ.-Prof. Mag. Dr. Horst Eidenberger

Wien, 16.04.2014

(Unterschrift Verfasser)

(Unterschrift Betreuer)

Erklärung zur Verfassung der Arbeit

Thomas Adelbauer, BSc
Semperitstraße 3/1, 2630 Ternitz

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum)

(Unterschrift Verfasser)

Danksagung

Ich danke meinen Eltern, Michael und Sylvia, für die finanzielle und mentale Unterstützung während der gesamten Studienzeit. Ohne eure Hilfe wäre mir diese Ausbildung verwehrt geblieben.

Ganz besonderen Dank gebührt weiter meinem Diplomarbeitsbetreuer Herrn Mag. Dr. Horst Eidenberger, welcher mir mit seinem fachkundigen Wissen stets mit Rat und Tat zur Seite stand.

Außerdem möchte ich mich bei Michael Hanser bedanken, welcher mir beim Kompositionsprozess mit seinem musikalischen Fachwissen weitergeholfen hat.

Ebenfalls danke ich meinen Korrekturlesern Sara, Sylvia, Stephan und Hanna und meinen Studienkollegen und Freunden, welche die letzten Jahre mit mir gemeinsam bestritten haben.

Ein herzliches Dankeschön geht auch an alle Damen und Herren, welche sich die Zeit genommen haben, um an der Evaluierung teilzunehmen.

Abstract

This master thesis dwells upon the creation of musical compositions which can be matched to specific genres. It is assumed that a set of characteristics and musical motifs can be used to obtain this goal. For this purpose different criteria of a group of MIDI data are analysed and monophonic and polyphonic motifs are extracted by the use of pattern recognition techniques. The obtained data and a genetic algorithm are used to create new compositions which are related to the original songs. The developed system uses a beat-based approach. In this thesis several musical and algorithmic concepts are introduced first. Second the analysis of the data and the concept for new compositions is presented. Finally the results received through interviews of a small group of people are shown. More than 50% of the used musical pieces were correctly assigned by the test persons. Also the quality of the compositions were positively rated.

Kurzfassung

Die vorliegende Diplomarbeit befasst sich mit der Erstellung musikalischer Kompositionen, welche einem bestimmten Genre zugeordnet werden können. Es wird angenommen, dass mit Hilfe von Sets an charakteristischen Eigenschaften und Motiven dieses Ziel erreicht werden kann. Zu diesem Zweck werden diverse Kriterien einer Gruppe von MIDI-Dateien analysiert und mit Hilfe von Mustererkennung sowohl monophone als auch polyphone Motive extrahiert. Mit Hilfe dieser Daten und eines genetischen Algorithmus werden neue Kompositionen erstellt, welche Ähnlichkeiten zu den ursprünglichen Liedern aufweisen sollen. Es wird ein Ansatz verfolgt, welcher mit Notenfolgen bestehend aus ganzen Takten arbeitet. Zunächst werden in der Arbeit musikalische und algorithmische Grundlagen erläutert. Anschließend wird die Analyse der Musikstücke und das Konzept der Komposition vorgestellt. Abschließend werden die Ergebnisse präsentiert, welche durch eine Befragung einer kleinen Personengruppe erhalten wurden. Mehr als die Hälfte der verwendeten Musikstücke konnte dabei von den Probanden richtig zugeordnet werden. Auch die Qualität der Kompositionen wurde durchwegs positiv bewertet.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Zielsetzung und Motivation	2
1.2	Forschungsstand	2
1.3	Überblick	6
2	Grundlagen	9
2.1	Musikalische Genres	9
2.2	Motive in der Musik	11
2.3	MIDI - Musical Instrument Digital Interface	13
2.4	Genetische Algorithmen und Komposition	16
2.5	Interessantheitsmaße	18
2.6	PBMC - Pattern-Based MIDI Composer	19
3	Entwurfsprozess	23
3.1	Anforderungen	23
3.2	Anwendungsfälle	26
3.3	Architektur	28
4	Genre-Analyse	33
4.1	Ziel	33
4.2	Methode	34
4.3	Ergebnisse	47
5	Genre-Basierte Komposition	49
5.1	Ziel	49
5.2	Methode	50
5.3	Ergebnisse	56
6	Evaluierung	59
6.1	Ziel	59
6.2	Methode	60
6.3	Ergebnisse	61

7 Schlussfolgerungen und Ausblick	67
Literaturverzeichnis	71



Einleitung

Formale Techniken zur Komposition von Musikstücken gehen zurück bis ins 11. Jahrhundert, als Guido d'Arezzo ein Modell entwickelte, welches jedem Laut eines religiösen Textes eine Tonlage zuwies [22]. Ein weiteres bekanntes Beispiel ist Mozarts musikalisches Würfelspiel, mit dessen Hilfe man Walzer komponieren konnte [5, 10]. Der Ansatz, Musik nach vorgegebenen Modellen zu entwickeln ist daher schon sehr lange üblich. Musik ist eine eigene Sprache, wie viele andere, und kann durch Grammatiken ausgedrückt und erzeugt werden [12]. Viele Arten von Musik entsprechen einem gewissen Schema und deshalb ist die Möglichkeit gegeben, diese Modelle in Computerprogrammen umzusetzen und automatisch neue Musikstücke zu generieren [5].

Computer sind viel besser als Menschen dazu geeignet, Arbeiten immer wieder auszuführen. Daher ist es für einen Computer leicht, eine Vielzahl verschiedener neuer Musikstücke zu erzeugen. Durch die sehr hohe Anzahl an Möglichkeiten ein Musikstück zu modifizieren, gibt es mehr Variationen, als ein Mensch in einer vertretbaren Zeit je komponieren könnte. Da es manchmal beim kreativem Prozess des Komponierens zu Blockaden kommt, welche unzählige Stunden der Frustration bedeuten können, werden Hilfsmittel zur Erzeugung von

Kompositionen verwendet, um den kreativen Prozess wieder in Gang zu bringen und sich neue Inspiration zu holen [17].

1.1 Zielsetzung und Motivation

Das Ziel der vorliegenden Arbeit war es, ein bereits bestehendes System, den Pattern Based MIDI Composer (PBMC) [19], so zu erweitern, dass dieses in der Lage ist, Lieder von so hoher Qualität zu komponieren, dass diese einem bestimmten Genre zugeordnet werden können. Die Erweiterung wird im weiteren Verlauf der Arbeit als Genre Based MIDI Composer (GBMC) bezeichnet. Ein weiteres Anliegen war es den Funktionsumfang des Systems, welches in der ursprünglichen Version mit monophonen Lieder arbeitete, auf Polyphonie zu erweitern, um die Qualität der Kompositionen zu verbessern. Da in den meisten Arbeiten zum Thema automatischer Komposition Musikstücke aus Klassik, J-Pop, Jazz und gregorianischen Korälen vertreten sind, wurde versucht, weitere und auch modernere Genres zu behandeln und für Kompositionen heranzuziehen [7, 38]. In der Arbeit wurden repräsentative Grundmenge diverser musikalischer Genres, basierend auf manuell erstellten umfangreichen Musiksammlungen automatisch analysiert und mit den so gewonnenen Informationen neue Kompositionen erzeugt.

1.2 Forschungsstand

Computer werden schon seit langer Zeit verwendet, um Musik zu komponieren. Im Jahr 1950 entwickelte Max Mathews bei AT&T erstmals ein Programm zur Klangsynthese. Seitdem werden Computerprogramme verwendet, um mit Hilfe von probabilistischen oder stochastischen Methoden Musik zu erzeugen. Bis Mitte der 80er Jahre lag das Hauptaugenmerk auf der Klangfarbe von Tönen. Da Computer zu dieser Zeit noch nicht über die heutige Leistungsfähigkeit verfügten, wurden die strukturellen Eigenschaften von Musik

nicht in den Fokus gestellt. Die meisten Ansätze konzentrierten sich nur auf die aktuelle Note und basierten meistens auf Markov-Ketten, um die Wahrscheinlichkeit der folgenden Noten zu berechnen [18].

In der aktuellen Forschung steht die strukturelle Komposition im Fokus. Ein Großteil aktueller Systeme verwendet Datenbanken mit verschiedensten Stilvorgaben und Regeln, welche Rahmenbedingungen bei der Erzeugung neuer Kompositionen vorgeben. Da der Computer in der Lage ist in kurzer Zeit eine neue Komposition zu generieren, wird er auch häufig verwendet, um einfach eine Vielzahl zufälliger Kompositionen zu erzeugen, welche anschließend sowohl manuell als auch automatisch bewertet werden müssen. Eine weitere Tendenz ist es, den Computer als begleitendes Instrument einzusetzen. In diesem Fall hört der Computer einem gerade gespielten Stück zu und erstellt passende Begleitphrasen in Echtzeit. Ein alternativer Ansatz, welcher auch in dieser Arbeit verfolgt wird, ist es, den kreativen Prozess des Komponierens nachzubilden [18]. Nachfolgend werden einige Beispiele von Anwendungen vorgestellt, welche einen Auszug aus dem aktuellen Stand der Technik darstellen. Teile der vorgestellten Systeme dienten dem Autor als Inspiration und Leitfaden bei der Umsetzung der Erweiterung von PBMC. Bei Unklarheiten diverser Begriffe verweist der Autor auf das 2. Kapitel.

ANTON

ANTON ist ein System zur automatischen Komposition von melodischen, harmonischen und rhythmischen Musikstücken. Weiter kann es Fehler in von Menschen erstellten Kompositionen erkennen und als computergestütztes Kompositionswerkzeug verwendet werden. Auf Grund der Kombination aus harmonischer, rhythmischer und melodischer Komposition in einem Framework hebt sich ANTON von vergleichbaren Systemen ab [6]. Ein verbreiteter Ansatz bei der algorithmischer Komposition besteht darin harmonische Klänge zu einer Melodie hinzuzufügen. Dazu werden Noten, deren Klang als angenehm und passend

empfunden wird, zur gleichen Zeit wie die Melodie gespielt. Die meisten Ansätze beginnen daher mit einer Melodie, für die zumindest eine passende harmonische Tonfolge existiert, welche das System nur finden muss. ANTON hingegen generiert die Melodie gleichzeitig mit einer harmonischen Tonfolge. Weiter gibt der Rhythmus einer Melodie an, wann und wie lange eine bestimmte Note gespielt und wie sehr sie hervorgehoben wird. Für ANTON ist ein eigenes Modell zur Rhythmuserzeugung erstellt worden, welches auf Grund des komplexen Ansatzes hier nur kurz beschrieben wird. Laut den Entwicklern von ANTON beruht ihre Rhythmuserzeugung auf der Wechselwirkung zwischen der Wirkung (*impact*) und anschließenden Entfaltung (*resolution*) der Töne. Die Wirkung eines Tons setzt sich aus der musikalischen Energie der vorherigen Töne und die Entfaltung aus der Freigabe der zuvor aufgebauten Energie zusammen [6].

CAUI

Die Erfinder von CAUI (Constructive Adaptive User Interface) befassen sich mit der Thematik, Musik basierend auf menschlichen Gefühlen zu komponieren. Im Speziellen beobachten Sie, welche Gefühle durch bestimmte Kompositionen ausgelöst werden, indem sie einer Testperson zuerst verschiedene Musikstücke vorspielen, die von dieser dann einer Kategorie von Gefühlen zugeordnet werden müssen. Mit Hilfe von Machine-Learning-Methoden werden Bedingungen für neue Kompositionen definiert. Zum Beispiel wird definiert, dass eine bestimmte Testperson Stücke favorisiert, welche aus der Tonart E-Dur oder E-Moll, mit Tempo Allegretto, einem 4/4-Takt, aus einer speziellen aufeinanderfolgender Akkordfolge bestehen und von einem Piano gespielt werden. Hierfür benutzen sie Inductive Logic Programming. In weiterer Folge verwenden sie ein System, welches eine bestehende Komposition neu arrangiert, indem es minimale Änderungen an der Akkordfolge vornimmt. Weiter verwenden sie für die Komposition einen genetischen Algorithmus, dessen Tauglichkeits-Funktion auf den zuvor bestimmten Bedingungen basiert. Ihr System

verwendet zur Generierung neuer Stücke ein Programm, welches eine Melodie auf Basis einer Akkordsequenz und einigen Regeln für die gewünschte Dauer erstellt. Ihr System unterscheidet sich grundlegend von anderen automatischen Kompositionssystemen, da es neue Stücke anhand eines personalisierten Modells generiert [30].

Variations

Variations ist ein System, welches versucht das Model der *harten Arbeit* im Kreativitätsprozess nachzubilden. Darunter verstehen die Entwickler den Prozess des Ausprobierens verschiedener Kombinationen von Tönen und Klängen und deren Auswertung. Da dieser Prozess als iterativ angesehen wird, wurde versucht, ihn als Algorithmus nachzubilden. Um dem Problem der Nächsten-Noten-Auswahl auszuweichen, arbeitet der Algorithmus dabei nicht auf Notenebene, sondern auf der Ebene der Motive (charakteristische Tonfolge), um mehr Augenmerk auf die Harmonie zu legen [18]. Die Erzeugung eines Musikstücks verläuft in folgenden Schritten:

1. Zuerst wird eine Grundmenge an Primärmotiven ausgewählt, welche später in der Komposition verwendet werden.
2. Danach werden kurze Phrasen mit Hilfe der zuvor ausgewählten Motive erzeugt und eine nach der anderen zu einer Melodie zusammengefügt. Nach jedem Hinzufügen wird das Ergebnis vom Ear-Modul evaluiert und bei einem nicht zufriedenstellendem Ergebnis die zuletzt hinzugefügte Phrase wieder entfernt.
3. Weitere Motive werden zufällig aus den Primärmotiven und erzeugten Motiven ausgewählt und daraus Variationen erstellt.
4. Bei einer ausreichend großen Anzahl an Phrasen werden diese dann zu einer Einheit zusammengefügt.

Die primären Softwarekomponenten sind der COMPOSER- und das EAR-Modul. Der Composer erzeugt Musikstücke und das Ear-Modul filtert *schlechte* Musik wieder heraus. Abbildung 1.1 zeigt den Ablauf dieses Prozesses. Der Composer erzeugt verschiedene Varianten von Musikstücken wie vorhin beschrieben und übergibt sie dem Ear-Modul, welches als Antwort *Ja* oder *Nein* zurückliefert. Wenn das Ear-Modul mit einem *Ja* antwortet wird mit der Komposition weiter gemacht. Bei einem *Nein* wird das zuletzt erzeugte Stück gelöscht und eine neue Variante ausprobiert [18].

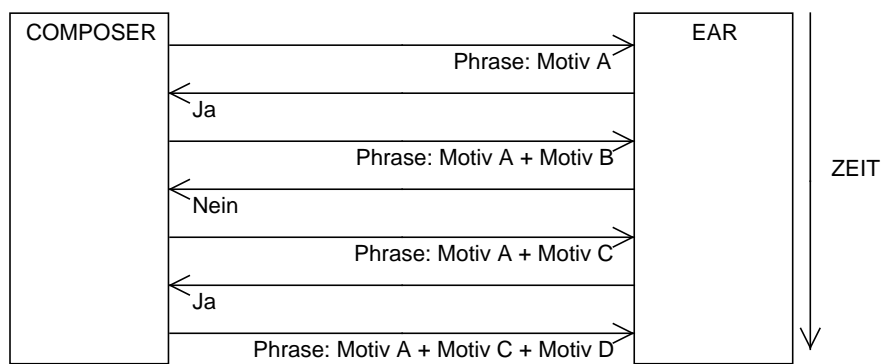


Abbildung 1.1: Prozessablauf zwischen Composer- und Ear-Modul

1.3 Überblick

Die folgenden Kapitel der Arbeit behandeln zuerst einige Grundlagen, gefolgt vom Entwurfsprozess, bestehend aus der Analyse der Genres, der Komposition neuer Lieder und der Evaluierung mit Hilfe einiger durch die Software erzeugter Stücke.

- Kapitel 2 befasst sich mit den musikalischen und technischen *Grundlagen* und gibt einen Überblick der Funktionen des PBMC.

- Kapitel 3 beschreibt die zusätzlichen Anforderungen und den *Entwurfsprozess*, welche an die Software GBMC (*Genre Based MIDI Composer*) gestellt wurden.
- Kapitel 4 ist der *Analyse* der Musikstücke gewidmet. Das Ergebnis dieser Analyse stellt ein Set an Regeln für jedes Genre dar, welches das Genre ausreichend beschreibt und es von den restlichen Genres unterscheidet beziehungsweise Gemeinsamkeiten verschiedener Genres aufdeckt.
- Kapitel 5 beschreibt den Prozess der Entstehung einer neuen *Komposition* auf Grundlage der zuvor erhaltenen Eigenschaften.
- Kapitel 6 präsentiert die *Ergebnisse* der Befragung einer Kontrollgruppe von 22 Probanden, welche die Qualität und die erkennbare Zugehörigkeit zu einem Genre behandelt.
- Kapitel 7 erläutert gewonnene Erkenntnisse und *Schlussfolgerungen* und zeigt weitere mögliche Erweiterungen des GBMC auf.

Grundlagen

In diesem Kapitel werden die grundlegenden Definitionen, Techniken und Funktionsweisen aus den Fachbereichen der Musik und der Informatik erläutert.

2.1 Musikalische Genres

Musikalische Genres entstehen auf Grund eines komplexen Zusammenspiels von kulturellen, künstlerischen und musikalischen Einflüssen um Ähnlichkeiten zwischen Musikern, Kompositionen oder Musiksammlungen zu charakterisieren. Ein musikalisches Genre dient daher der Kategorisierung und Organisation von Musik. Eine der grundlegenden Fragen bei der Zuordnung von Musik zu einem Genre ist, auf welcher Basis man die Klassifizierung anwenden soll: pro Stück, pro Künstler oder pro Album. Wenn man annimmt, dass ein Stück einem bestimmten Genre zugeordnet werden kann, wobei dies schon in Frage gestellt wird, ist es für ein ganzes Album nicht mehr so leicht, da dieses aus heterogenen Stücken bestehen kann. Das gleiche Problem besteht auch bei der Zuordnung von Künstlern zu einem Genre, da deren Musikstil oftmals einer Bandbreite von Genres zugerechnet werden kann [35]. In

Abbildung 2.1 sieht man eine mögliche Einteilung von Genres und Subgenres der Kategorie *Dance & Electronic*, verwendet von Amazons Musikladen. Als exemplarische Erweiterung wurde vom Autor eine weitere Ebene mit einer noch feineren Unterteilung des Subgenres *Drum & Bass* hinzugefügt, um aufzuzeigen, dass eine feinere Granularität möglich ist. In der Musikwelt herrscht über die erste Ebene der dargestellten Genres ein gewisser Grad der Einigkeit. Doch schon bei der Ebene der Subgenres ist es nicht mehr trivial klare Grenzen zu ziehen [2]. Ein weiteres Problem bei der Kategorisierung von Stücken sind Kooperationen zwischen Musikern, welche unterschiedlichen Genres angehören oder die Kombination von verschiedenen Stilrichtungen. Beides kann in neuen Subgenres resultieren.

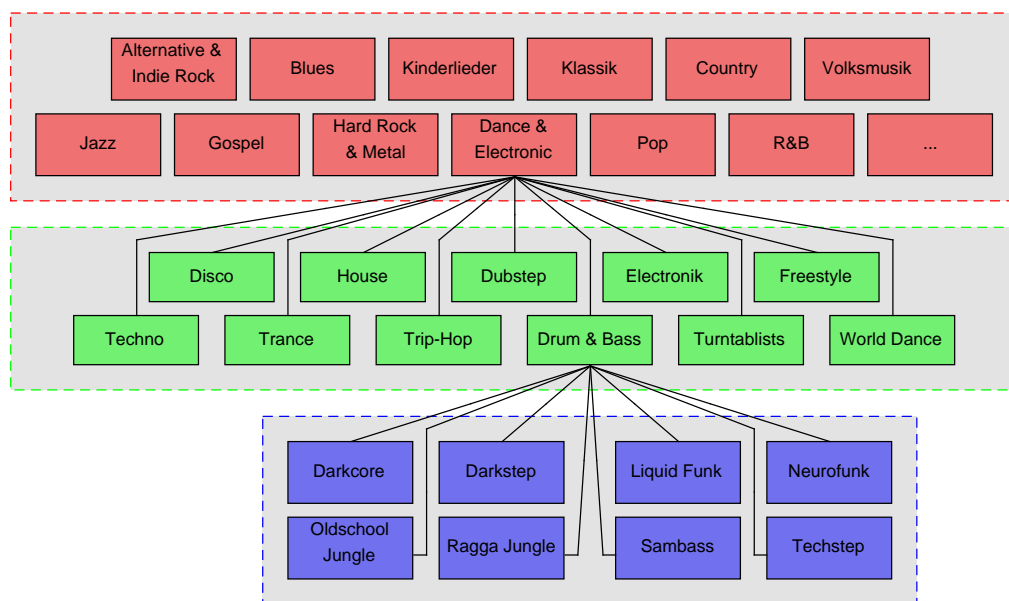


Abbildung 2.1: Mögliche Einteilung von Genres

2.2 Motive in der Musik

Eine Melodie besteht aus mehreren Motiven, welche sich wiederholen und weiterentwickeln. Man kann ein Motiv als kleinste sinntragende musikalische Einheit definieren. Es ist gleichbedeutend mit einem Wort einer Sprache. So wie ein Satz aus mehreren Wörtern besteht, bilden Motive musikalische Phrasen [21, 29]. Ein Motiv ist daher einfach ausgedrückt ein häufig erscheinendes Muster, das aber in unterschiedlichen Formen auftreten kann [24]. Beim Hören eines Musikstücks wird der Hörer durch das wiederholte Auftreten von Motiven vertraut mit dem Stück. Aus diesem Grund ist es üblich, dass sich Komponisten das Motiv als wiederkehrendes musikalisches Element zu Nutze machen [33, 16]. Dies zeigt sich auch im Phänomen der Erwartung des Zuhörers. So ist es einem Hörer meistens noch möglich eine Melodie fortzusetzen, auch wenn das Lied schon zu Ende ist [26, 4]. Motive erscheinen im Verlauf eines Stücks in unterschiedlichen Formen. Die Erkennung dieser Motive ist im Gebiet der automatischen Komposition von großer Bedeutung. Nachfolgend und in Abbildung 2.2 werden einige dieser Motivvariationen vorgestellt:

1. Die Wiederholung, oder auch Repetition, ist die einfachste Form der Variation. Sie besteht aus einer exakten Kopie des ursprünglichen Motivs [19].
2. Die Sequenz ist eine transponierte Wiederholung eines Motivs. Sie tritt um einige Halbtonschritte vom ursprünglichen Motiv versetzt auf [19].
3. Der Krebs, oder auch Reversion, ist ein Motiv, welches von hinten nach vorne gespielt wird [19].
4. Der Spiegel, oder auch Inversion genannt, ist ein Motiv, dessen Intervalle zwischen den Noten gespiegelt sind. Ein Halbtonschritt nach oben im ursprünglichen Motiv resultiert in einem Halbtonschritt nach unten im Spiegel [19].

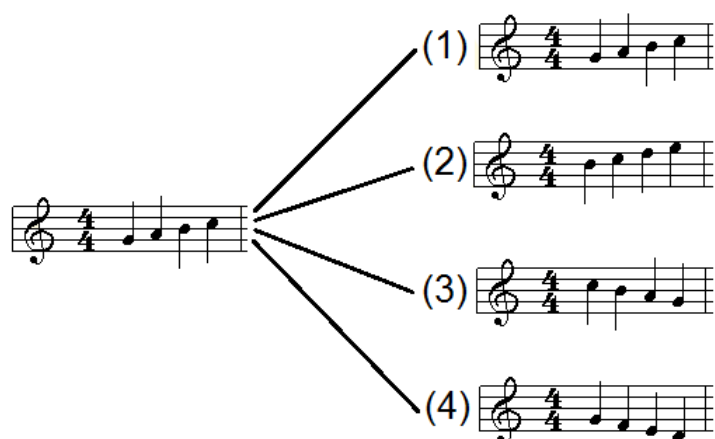


Abbildung 2.2: (1) Repetition, (2) Sequenz, (3) Reversion, (4) Inversion

Es existieren noch weitere Formen von Motiven, wie die Augmentation oder die Diminution, welche hier aber nicht weiter behandelt werden, da diese Motivarten im Verlauf dieser Arbeit nicht verwendet werden.

Monophonie versus Polyphonie

Polyphonie, oder auch als Mehrstimmigkeit bezeichnet, ist das gleichzeitige Erklängen mehrere verschiedener Töne. Im Gegensatz dazu steht die Monophonie (*Einstimmigkeit*), bei der nur eine Stimme zu einem Zeitpunkt gespielt wird. Auf Grund der polyphonen Natur von Musik ist es beim Automatic Composing üblich, ebenfalls polyphone Stücke zu erstellen. Die Verarbeitung eines polyphonen Signals ist rechnerisch wesentlich aufwendiger als die Bearbeitung eines monophonen Datenstroms, welcher nur aus linearen Notensequenzen besteht [32]. Die Analyse monophoner Musik ist ein gut exploriertes Forschungsgebiet. Eine Vielzahl von Algorithmen wurde in der Vergangenheit veröffentlicht und steht zur kommerziellen oder freien Nutzung zur Verfügung. Diese Algorithmen arbeiten in der Regel in Echtzeit. Im Gegensatz dazu ist die Forschung im Bereich der polyphonen Musik, welcher eine größere Herausforderung darstellt, noch nicht so weit fortgeschritten. Hier mangelt es

noch an diversen computergestützten Modellen [27, 25, 34].

2.3 MIDI - Musical Instrument Digital Interface

MIDI wurde in den 1980ern als standardisierte Schnittstelle zur Steuerung und Kopplung von Synthesizern und Klangmodulen entwickelt. Im MIDI-Standard werden drei verschiedene Aspekte der Schnittstelle spezifiziert [1]:

- Das Kommunikationsprotokoll, welches die Steuerung der MIDI-Geräte mit Hilfe der verschiedenen Typen von MIDI-Nachrichten spezifiziert. Die wichtigsten Nachrichten sind die *Note On*- und *Note Off*-Nachrichten, welche den Beginn und das Ende einer Note angeben.
- Die Hardwareschnittstelle zur Kommunikation zwischen diversen MIDI-Geräten. MIDI-Geräte verfügen meistens über drei Buchsen für Input, Output und *Thru*. Die *Thru*-Buchse wird verwendet, um mehrere MIDI-Geräte zu verbinden und mit dem gleichen MIDI-Datenstrom zu versorgen.
- Das binäre Dateiformat, welches für die Speicherung von Sequenzen vorgesehen ist. Hierbei handelt es sich um MIDI-Nachrichten, welche mit einem Zeitstempel versehen sind. Ein eigener Typ von MIDI-Nachrichten beinhaltet Meta-Ereignisse wie zum Beispiel die Tonart oder das Tempo des Stücks.

Weiter gibt es drei verschiedene Typen von MIDI-Dateien:

- Typ 0 besteht lediglich aus einer einzelnen Sequenz.
- Typ 1 dient zur Speicherung eines einzelnen Stücks, welches aus mehreren Spuren bestehen kann.
- Typ 2 beinhaltet mehrere Stücke wobei pro Spur ein Stück gespeichert wird.

Tempo und Zeit

Üblicherweise definieren Musiker das Tempo eines Stücks durch die Anzahl der Viertelnoten pro Minute (Beats per Minute - BPM). Ein Tempo von 100 BPM bedeutet daher, dass in einer Minute 100 Viertelnoten vorkommen. MIDI hingegen verwendet den genau umgekehrten Ansatz um das Tempo auszudrücken, die Zeit pro Viertelnote. Der Grund hierfür ist, dass es so um einiges leichter ist präzise Tempoangaben zu machen, da man feinere Abstufungen erzeugen kann. Ein weiterer wichtiger Wert bei der Berechnung des Tempos im MIDI-Standard sind die *Pulses Per Quarter Note*, kurz PPQN. Dieser Wert beschreibt, nach wievielen Ticks einer internen Uhr eine Viertelnote vorbei ist. Bei einem PPQN von 96 ist nach 96 Ticks eine Viertelnote vorbei, nach 48 Ticks eine Achtelnote, nach 24 Ticks eine Sechzehntelnote und so weiter.

Aufbau einer MIDI-Datei

MIDI-Datenströme enthalten im Gegensatz zu digitalem Audio (z.B. WAV oder MP3) keine tatsächlichen Klänge, sondern nur die notwendigen Kontrollinformationen zur Steuerung des Klangmoduls. Dadurch ist es relativ einfach MIDI-Ströme programmgesteuert zu analysieren und zu modifizieren. In Abbildung 2.3 sieht man den grundlegenden Aufbau einer MIDI-Datei. Eine MIDI-Datei, welche ebenfalls als Score bezeichnet wird, besteht aus mindestens einem Part. Ein Part kann wiederum aus einer oder mehreren Phrasen bestehen, welche die einzelnen Noten der Spur beinhalten. Jedem Part kann ein Instrument zugewiesen werden, welches die Notenfolgen spielen soll. Alle in diesem Part befindlichen Phrasen verwenden daher dasselbe Instrument. Jedem Part wird ein Kanal zugewiesen, wobei der zehnte Kanal eine besondere Rolle einnimmt. Dieser ist ausschließlich für Percussions reserviert. Auf den ersten Blick besteht dieser Part ebenfalls aus Phrasen mit unterschiedlichen Noten. Der Unterschied hierbei ist aber, dass jede Note als ein anderes

Percussioninstrument (z.B. Becken, Tom, Snare-drum, Base-drum etc.) interpretiert wird. So wird die Note mit der Tonhöhe 60, welche üblicherweise dem mittleren C entspricht, in dieser Spur als ein hohes Tom interpretiert. Auf Grund des einfachen und klaren Aufbaus

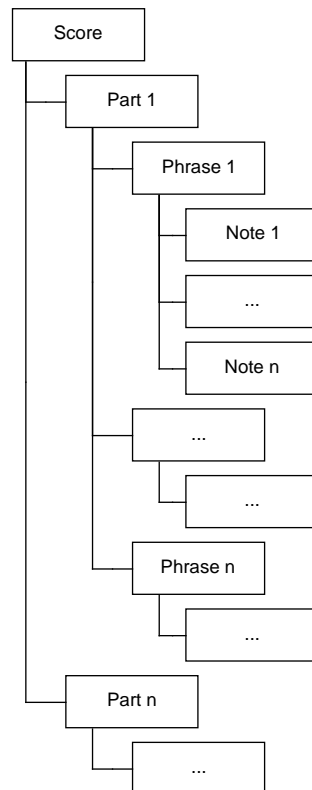


Abbildung 2.3: Aufbau einer MIDI-Datei

einer MIDI-Datei ist es sehr einfach, diese mit Hilfe von Computerprogrammen zu analysieren, zu verändern oder zu erstellen. Durch Echtzeitverarbeitung ist es auch möglich, MIDI-Signalströme dynamisch während des Abspielen des Stücks zu steuern.

2.4 Genetische Algorithmen und Komposition

Auf Grund des strukturellen Aufbaus von Musik liegt es nahe, Programme zu verwenden um diese zu erzeugen. Dies kann nach einem vorgegebenen Schema entweder deterministisch oder stochastisch geschehen. Deterministische Algorithmen erzeugen reproduzierbare Kompositionen anhand zuvor definierter Vorgaben [10]. Die Ergebnisse sind dadurch vorhersehbar und nicht so vielfältig. Im Gegensatz dazu ist bei stochastischen Algorithmen, wie zum Beispiel Markov-Ketten oder genetischen Algorithmen, das Ergebnis nicht reproduzierbar, da dieses auf Zufallszahlen und Wahrscheinlichkeitstabellen basiert [22]. Wird ein stochastischer Algorithmus nur minimal abgeändert, werden die erzeugten Stücke charakteristisch anders sein [10]. Die meisten komplexen Systeme zur Erstellung algorithmischer Kompositionen benötigen eine Vielzahl an Parametern. Daher basiert ein Großteil der Arbeiten im Bereich der automatischen Komposition auf genetischen Algorithmen. Auf Grund ihrer Vorgehensweise eignen sich diese besonders gut in diesem Teilgebiet der Informatik. Man kann die benötigten Parameter nach eigenen Vorgaben *züchten*, ohne ein genaues Wissen der darunter liegenden Struktur haben zu müssen, aber dennoch ein hohes Maß an Kontrolle behalten. Man muss lediglich in der Lage sein, Zwischenergebnisse bewerten zu können [15].

Genetische Algorithmen basieren auf dem natürlichen Vorbild der Evolution und sind aus dem Fachgebiet der Biologie abgeleitet. Wie der Name vermuten lässt verwenden sie das biologische Modell der Evolution nach Darwin, welches nach dem Prinzip des 'Überleben des Stärksten/am meisten Angepassten' funktioniert [8]. Durch Konkurrenz, Selektion, Weiterentwicklung und Mutation entstehen neue Musikstücke, basierend auf den zuvor erzeugten Teilen. Genetische Algorithmen werden auch als heuristische Suchalgorithmen bezeichnet. Der größte Nachteil bei genetischen Algorithmen ist das Flaschenhalsproblem auf Grund der benötigten Tauglichkeits-Funktion (Fitness-Funktion), welche bestimmt wie

gut das Ergebnis (die neue Generation) sich für eine Weiterverwendung eignet [31, 22, 3, 9].

Ein genetischer Algorithmus besteht aus folgenden Komponenten [23]:

- Darstellung der Lösungsmenge in Form so genannter Chromosome
- Mutations- und Kreuzungsoperatoren, um neue Individuen zu erzeugen
- Einer Tauglichkeits-Funktion zur Bewertung neuer Individuen
- Eine Selektionsmethode, die sicherstellt, dass geeignetere Individuen eine höhere 'Überlebenschance' haben

Die grundlegenden Schritte des iterativen Prozesses sind [23]:

1. Generierung einer zufälligen Population
2. Evaluierung aller Chromosome mit Hilfe der Tauglichkeits-Funktion und Anwendung von Mutationen und Kreuzungen um neue Chromosome zu erzeugen
3. Entscheidung, welche neuen Chromosome ältere Chromosome ersetzen sollen (mit Hilfe der Tauglichkeits-Funktion)
4. Die Schritte 2 und 3 werden so lange wiederholt, bis ein Endkriterium erfüllt wurde

Als einen Kreuzungsoperator, oder Crossover, bezeichnet man die Erzeugung eines oder mehrerer Nachkommen von zwei oder mehr Eltern. Der einfachste Fall wird als One-Point-Crossover bezeichnet. Bei diesem werden zwei Genome an der jeweils gleichen Stelle gebrochen und die Teilstücke ausgetauscht. Bei einem N-Point-Crossover hingegen unterteilt man die Genome an mehreren Stellen und tauscht diese aus [15]. In den Abbildungen 2.4 und 2.5 sind Beispiele für beide Arten von Kreuzungen dargestellt. Die Mutation ist ei-

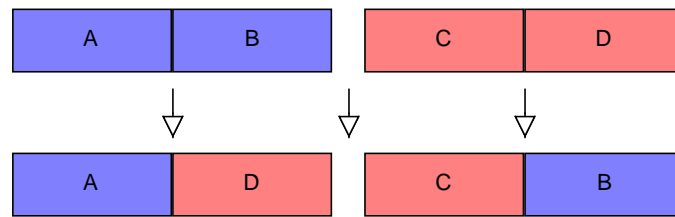


Abbildung 2.4: Beispiel eines One-Point-Crossovers

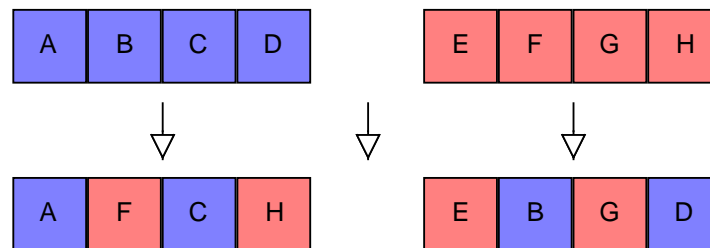


Abbildung 2.5: Beispiel eines N-Point-Crossovers

ne weitere Methode um neue Nachkommen zu erzeugen. Sie ist für jeden Datentyp unterschiedlich definiert. Besteht ein Genom zum Beispiel nur aus Bits, so wie im Beispiel in Abbildung 2.6, kippen bestimmte Bits mit einer gewissen Wahrscheinlichkeit. Es gibt verschiedene Mutationsmechanismen für ein Problem. So wie der Crossover hat die Mutation destruktive und konstruktive Effekte auf die Population. Die Wahrscheinlichkeit einer Mutation sollte im Allgemeinen nicht zu hoch gesetzt werden, da sonst der genetische Algorithmus in eine Zufallssuche abgleitet [15].

2.5 Interessantheitsmaße

Der Begriff des Interessantheitsmaß stammt aus dem Bereich des Data Minings. Diese Maße dienen dazu, die Interessantheit von Aussagen zu bewerten, welche bei der Analyse von

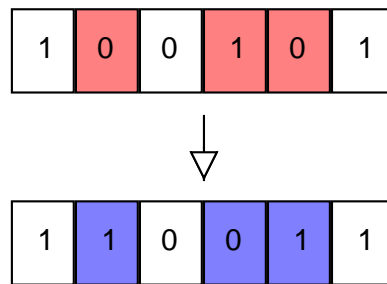


Abbildung 2.6: Beispiel einer Mutation von Bits

Datenbanken erhalten werden. Mit Hilfe eines solchen Maßes kann man die Suche in großen Datenmengen einschränken und so einen effizienteren Zugriff auf die benötigten Informationen erzielen [28]. Sie werden zur Selektion und Reihung von Mustern verwendet, um Benutzern die Suche nach potenziell interessanten Datensätzen zu erleichtern [14]. In der vorliegenden Arbeit wird ein, wie in [14] beschrieben, semantisches Maß verwendet. Die detektierten Motive erhalten auf Grund der im Motiv enthaltenen Noten eine unterschiedliche Gewichtung, anhand welcher die Interessanztheit des jeweiligen Motivs erkennbar ist.

2.6 PBMC - Pattern-Based MIDI Composer

Der Pattern-Based MIDI Composer (PBMC) ist ein in Java entwickeltes Programm, welches 2013 im Rahmen einer Diplomarbeit an der TU Wien von Jakob Fellner mit dem Ziel entwickelt wurde, Algorithmen zur Erzeugung kurzer Melodien zu nutzen. Es ist in der Lage, Musikstücke zu generieren, welche möglichst harmonisch und wohlklingend sind. Das Programm kann in die Sparte des *Automated Composing* eingereiht werden, wobei auch Ansätze des *Assisted Composing* beinhaltet sind. Der PBMC geht, im Gegensatz zu den meisten anderen Composern, nicht von gänzlich neue Kompositionen aus, sondern verwendet für die Erzeugung von Melodien bereits vorhandenes Material. Dieses generiert er aus

der Analyse von Musikstücken, aus welchen er Motive extrahiert, die mit Hilfe von Mustererkennungsalgorithmen detektiert werden. Der PBMC-Composer diente als Grundlage für die vorliegende Arbeit und wurde in diesem Rahmen um mehrere Funktionen erweitert, wie zum Beispiel die Verarbeitung polyphoner Musikstücke. Als Eingabeformat wurde MIDI auf Grund der zuvor erwähnten einfachen Analysierbarkeit und Manipulierbarkeit gewählt.

Der PBMC in seiner ursprünglichen Form arbeitet auf einer monophonen Ebene und erzeugt Kompositionen, welche Ähnlichkeiten zu den zuvor detektierten Motiven aufweisen. Zur Erzeugung neuer Kompositionen kann man zwischen einem genetischen Algorithmus und Markov-Ketten wählen, wobei der genetische Algorithmus, im Sinne der Harmonielehre, die besseren Ergebnisse erzielte. Im Gegensatz dazu ist der Markov-Composer um einiges schneller in der Erzeugung neuer Stücke. Bei der vom Autor durchgeführten Evaluierung wurde die Qualität der Kompositionen, welche vom genetischen Algorithmus erzeugt wurden, schlechter befunden als die des Markov-Composers. Da die Kompositionen des genetischen Algorithmus jedoch größere Unterschiede zu den Ausgangsstücken aufweisen und somit eher neue Melodien erzeugt werden, entschied sich der Autor gegen die Verwendung von Markov-Ketten [19]. In Abbildung 2.7 wird das Benutzerinterface des PBMC gezeigt. Im linken oberen Teil des Bilds sieht man die Parts und Phrasen einer geöffneten MIDI-Datei. Darunter werden die detektierten Motive der Phrase dargestellt. Im rechten Teil des Bilds sieht man die Notenfolgen der ausgewählten Phrase und die markierten Vorkommen des ausgewählten Motivs.

The screenshot displays the Pattern-Based MIDI Composer (PBMC) software interface. The main window shows a score editor with a tree view on the left and a data table in the center. The table lists musical notes with their indices, hanson relationships, differences, notes, octaves, durations, and pitches. The bottom section includes tabs for 'Repetitions', 'Variations', and 'Rhythmic', and a property table showing the current motif and tempo.

index	post hanson	pre hanson	post diff	pre diff	note	octave	duration	pitch
0	par1.0		0	5	D	4	0.5	62
1	loo1.0	par1.0	5	0	G	4	0.5	67
2	loo1.0	loo1.0	0	0	G	4	0.5	67
3	dal1.0	loo1.0	0	-1	G	4	0.5	67
4	nar1.0	dal1.0	-1	3	F#	4	0.5	66
5	del1.5	nar1.0	3	-7	A	4	0.5	69
6	loo1.5	del1.5	-7	0	D	4	1.0	62
7	mar1.0	loo1.5	0	4	D	4	0.5	62
8	nar1.0	mar1.0	4	3	F#	4	0.5	66
9	nar1.0	nar1.0	3	3	A	4	0.5	69
10	dal1.0	nar1.0	3	-1	C	5	0.5	72
11	nar1.0	dal1.0	-1	3	B	4	0.5	71
12	del1.5	nar1.0	3	-7	D	5	0.5	74
13	mar1.5	del1.5	-7	4	G	4	1.0	67
14	loo1.0	mar1.5	4	0	B	4	0.5	71
15	dar1.5	loo1.0	0	1	B	4	0.5	71
16	nal1.5	dar1.5	1	-3	C	5	1.0	72
17	loo1.0	nal1.5	-3	0	A	4	0.5	69
18	sar1.5	loo1.0	0	2	A	4	0.5	69
19	mal1.5	sar1.5	2	-4	B	4	1.0	71
20	loo1.0	mal1.5	-4	0	G	4	0.5	67
21	sar1.0	loo1.0	0	2	G	4	0.5	67
22	loo1.0	sar1.0	2	0	A	4	0.5	69
23	nal1.0	loo1.0	0	-3	A	4	0.5	69
24	loo1.0	nal1.0	-3	0	F#	4	0.5	66
25	dar2.5	loo1.0	0	1	F#	4	0.5	66
26	pal2.5	dar2.5	1	-5	G	4	2.0	67
27	par1.0	pal2.5	-5	5	D	4	0.5	62
28	loo1.0	par1.0	5	0	G	4	0.5	67
29	loo1.0	loo1.0	0	0	G	4	0.5	67
30	dal1.0	loo1.0	0	-1	G	4	0.5	67
31	nar1.0	dal1.0	-1	3	F#	4	0.5	66
32	del1.5	nar1.0	3	-7	A	4	0.5	69
33	loo1.5	del1.5	-7	0	D	4	1.0	62
34	mar1.0	loo1.5	0	4	D	4	0.5	62
35	nar1.0	mar1.0	4	3	F#	4	0.5	66
36	nar1.0	nar1.0	3	3	A	4	0.5	69

property value
name motif2
tempo 82.0
#motif matches 4

logged in as tom

Abbildung 2.7: Ursprüngliche Version des PBMC



Entwurfsprozess

Nachdem im letzten Kapitel ein Einblick in die Grundlagen des Forschungsbereichs gegeben wurde, folgen nun eine Beschreibung des Konzeptes und des Aufbaus zur Erstellung des Systems, welches sich aus den Modulen der Genre-Analyse, der genre-basierten Komposition und der Evaluierung zusammen setzt. Diese werden in Abbildung 3.1 dargestellt. Im ersten Teil werden die Anforderungen, welche an die Software gestellt werden beschrieben. Im zweiten Teil folgen die wichtigsten Anwendungsfälle und der Systemaufbau des GBMC.

3.1 Anforderungen

Eine wichtige Anforderung bestand darin, dass die Basissoftware, welche in [19] erstellt wurde, weiterhin funktionsfähig bleibt. Somit war es notwendig, die Erweiterungen so in das System zu integrieren, dass es zu keinen Komplikationen bei der Ausführung kommen kann. Durch die Integration der Erweiterungen als entkoppeltes Unterprogramm innerhalb der ursprünglichen Version des PBMC wurde letztendlich dazu übergegangen, die neue

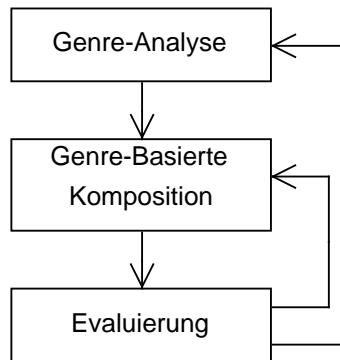


Abbildung 3.1: Entwurfsprozess

Version als eigenständiges Programm zur Verfügung zu stellen. Es werden aber weiterhin bereits existierende Funktionen und Datenstrukturen übernommen, sofern diese ohne Änderungen verwendbar sind. Obwohl es sich um die Implementierung eines Prototypen handelt wurde versucht einen ausreichenden Grad an Benutzerfreundlichkeit zu erzielen. Einige weitere Anforderungen waren auf Grund der Erweiterung einer zur Verfügung gestellten Software implizit vorausgesetzt und wurden dementsprechend übernommen.

- Als Input-Format wird auf Grund der Kompatibilität weiterhin auf Musikstücke im MIDI-Format zurückgegriffen.
- Um die Vielfalt der möglichen Musikrichtungen nicht zu beschränken wird die voll-automatische Analyse von selbst erstellten Musiksammlungen im MIDI-Format angedacht, welche als repräsentative Grundmengen (*Ground Truth*) diverser Genres dienen. Die Analyse soll eine ausreichende Anzahl an Eigenschaften bereitstellen, um die so erzeugten Genres von einander abgrenzen zu können, beziehungsweise Ähnlichkeiten in und zwischen Genres zu detektieren. Diese Eigenschaften sollen für eine spätere und mehrmalige Verwendung gespeichert werden können.

- Die Motiverkennung innerhalb der Musikstücke ist ein essentieller Bestandteil für die Erstellung von Kompositionen. Im Gegensatz zur Basisversion des PBMC, welcher ausschließlich mit monophonen Melodien arbeitet, soll die Erweiterung GBMC in der Lage sein polyphone Melodien zu analysieren und in weiterer Folge polyphone Motive aus diesen zu extrahieren. Das Ziel der Motiverkennung besteht nicht darin möglichst alle Vorkommen zu finden, sondern eine *ausreichende Anzahl an Motiven für die Erzeugung neuer Kompositionen* in akzeptabler Prozesslaufzeit bereitzustellen. Auf Grund der großen Datenmengen, welche bei der Analyse von Musiksammlungen aufkommen, wird nicht der übliche Brute-Force-Ansatz verfolgt, um Motive innerhalb der Stücke zu finden. Die Motivlänge beschränkt sich daher auf die Länge eines Takts, um so eine signifikante Leistungsverbesserung zu erzielen.
- Die finale Anforderung an das zu entwickelnde System besteht in der Entwicklung eines neuen Composers, welcher die Genre-Eigenschaften berücksichtigt und in der Lage ist neue Musikstücke zu generieren. Die so entstehenden Kompositionen sollen den Charakter des ursprünglichen Genres widerspiegeln. Es müssen daher neben der generierten Hauptmelodie passende begleitende Tonspuren erzeugt werden. Die Kompositionen sollen innerhalb der Software wiedergegeben werden können. Eine Speicherung der Stücke im MIDI-Format soll ebenfalls möglich sein.
- Eine optionale Anforderung besteht in der Darstellung der Musikstücke als Notenblatt, zur besseren Überprüfbarkeit der Ergebnisse.

Zusätzlich werden verschiedene Anforderungen an die Datenstruktur gestellt:

- Robustheit
- Einfachheit
- Interoperabilität

Die Robustheit wird mit Hilfe einer vor der Analyse durchgeführten Quantisierung der Daten erzielt. Durch die Anlehnung der Datenstruktur an den MIDI-Standard werden die Kriterien der Einfachheit und der Interoperabilität erfüllt.

3.2 Anwendungsfälle

Nachfolgend sind einige Anwendungsfälle der Erweiterungen des PBMC aufgeführt.

3.2.1 Analyse

Wie Abbildung 3.2 zeigt ist es bei der Erstellung einer neuen Kategorie bzw. eines Genres möglich, schon vorsortierte Musiksammlungen als Quellverzeichnis anzugeben. Alternativ können einzelne Stücke ausgewählt und anschließend analysiert werden. Weiter kann

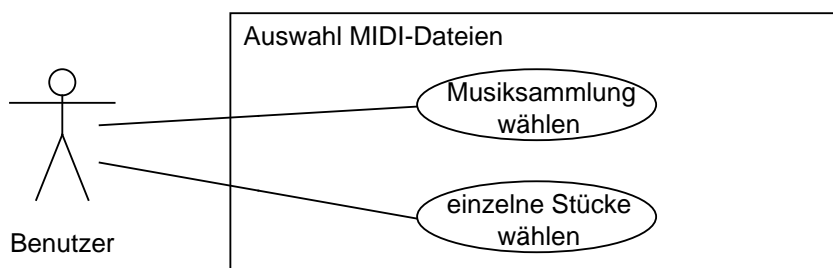


Abbildung 3.2: Auswahl von MIDI-Dateien

man die Extraktion der Eigenschaften einer Musiksammlung, wie die Notenhäufigkeiten, die Instrumentalisierung der Stücke und die Tonarten getrennt von der polyphonen Motivextraktion durchführen und diese anschließend speichern. Alternativ dazu ist es möglich, alle erforderlichen Attribute und polyphonen Motive in einem einzigen Prozess zu extrahieren und automatisch speichern zu lassen. Dies ist in Abbildung 3.3 ersichtlich. Eine Anzeige- und Kontrollmöglichkeit der durch die Analyse erhaltenen Kriterien soll, wie in

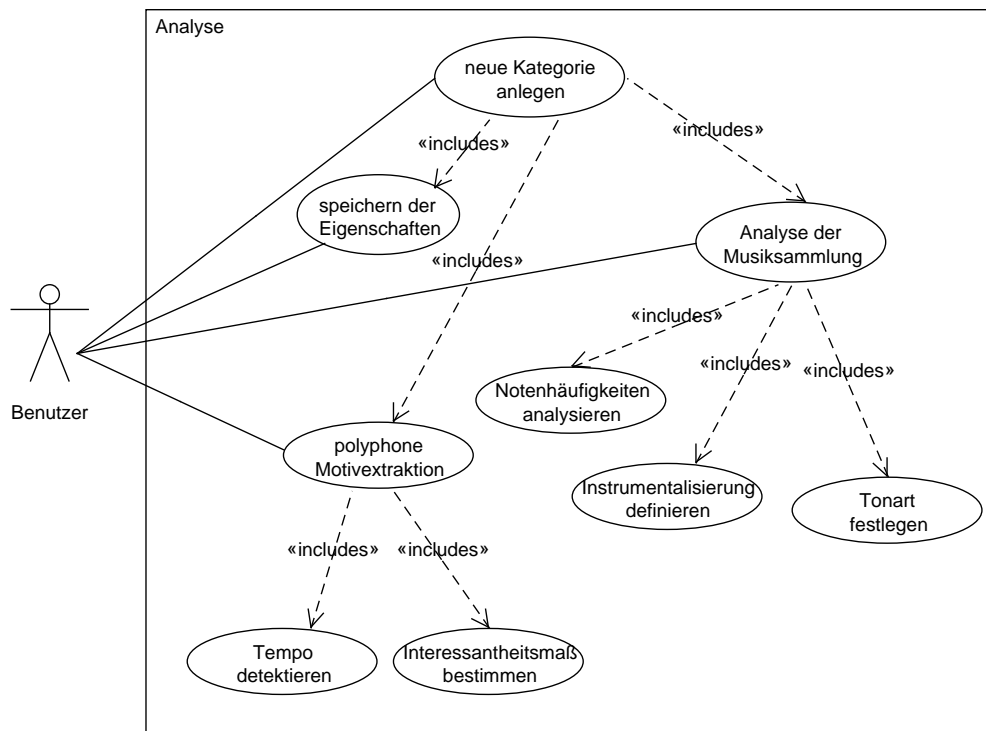


Abbildung 3.3: Analyse von Musiksammlung und Speichern der Eigenschaften

Abbildung 3.4 dargestellt, ebenfalls möglich sein. So soll es zum Beispiel möglich sein extrahierte Motive in Textform und Notendarstellung anzuzeigen und diese auch abzuspielen. Weiter soll der Benutzer die Möglichkeit haben sich Informationen der globalen Attribute anzeigen zu lassen, wie die Tonarten oder die Instrumentalisierung der analysierten Stücke.

3.2.2 Komposition

Vor dem Starten des Prozesses zum Erzeugen einer neuen Komposition soll der Benutzer die Möglichkeit haben, mehrere Parameter zu setzen, um das Ergebnis in eine bestimmte Richtung beeinflussen zu können. Diese umfassen, wie in Abbildung 3.5 dargestellt, un-

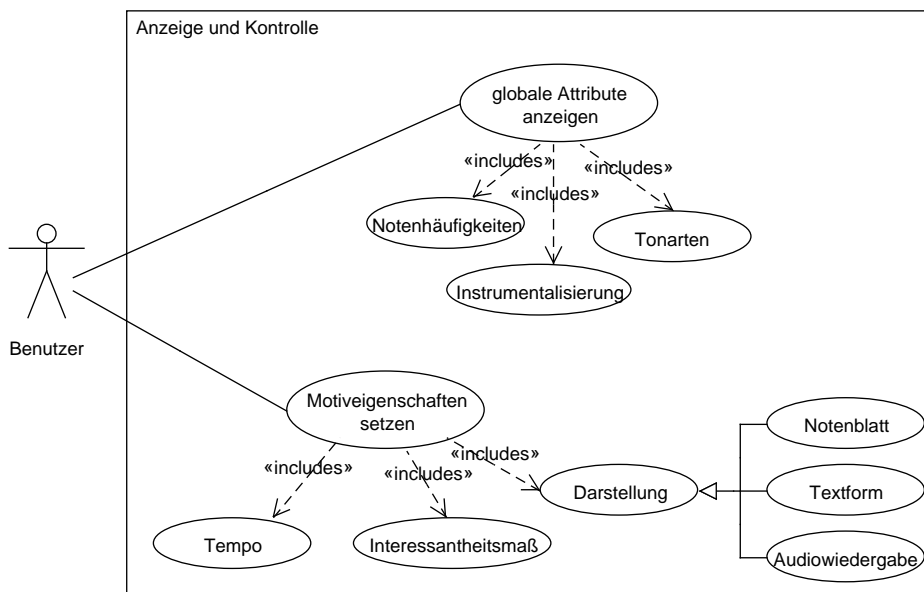


Abbildung 3.4: Anzeige und Kontrolle der Eigenschaften und Motive einer Musiksammlung

ter anderem die Anzahl der zu verwendenden Instrumente, die Instrumentalisierung, die Lautstärke der einzelnen Instrumente, die Dauer, das Tempo und die Tonart des Stücks. Außerdem soll es dem Benutzer möglich sein mit Hilfe der Attribute des Fitnesswerts, der Größe der Grundpopulation und der Art der Takterzeugung Einfluss auf den genetischen Algorithmus zu nehmen. Erzeugte Kompositionen sollen direkt im Programm wiedergegeben werden können. Weiter soll eine Speicherung im MIDI-Format möglich sein, um die Kompositionen zu bewahren und die Möglichkeit zu schaffen, diese in anderen Programmen weiter zu bearbeiten. Dies wird in Abbildung 3.6 veranschaulicht.

3.3 Architektur

Da es sich bei GBMC nur um die Implementierung eines Prototypen handelt ist es wichtig seine Teile in verschiedene Module auszulagern. Diese Module können bei späteren Erwei-

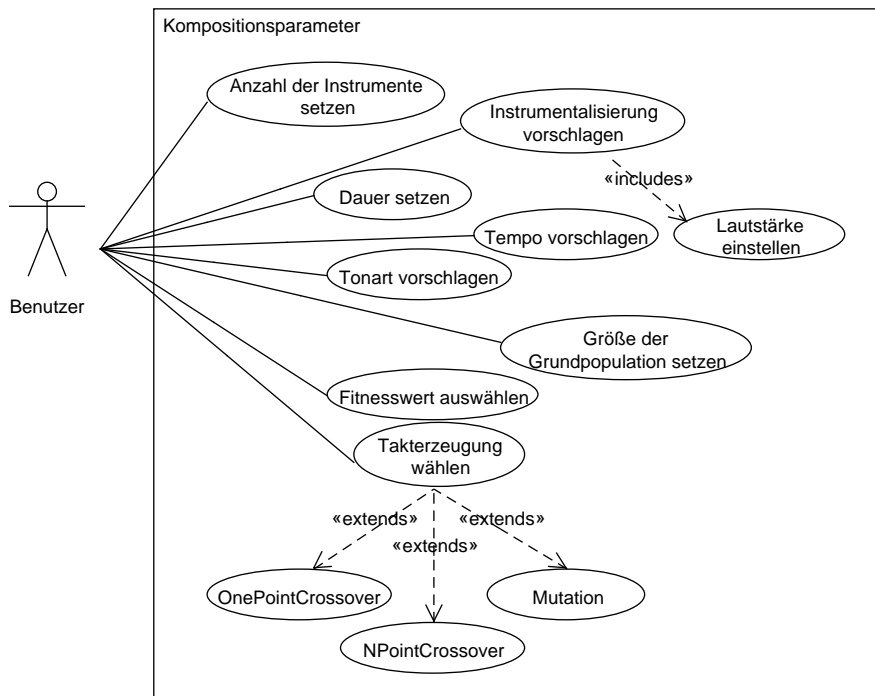


Abbildung 3.5: Parameter der Komposition

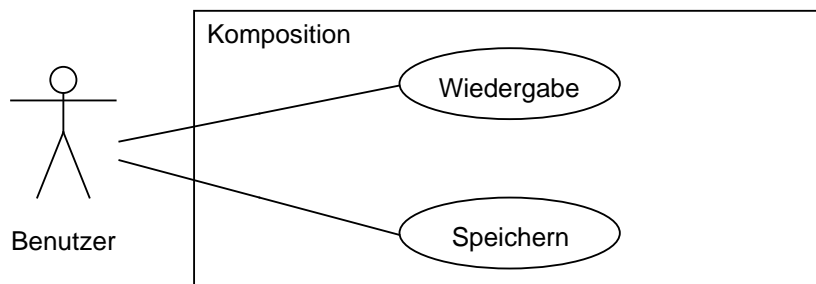


Abbildung 3.6: Weiterverwendung der Komposition

terungen oder Änderungen einfach bearbeitet bzw. ausgetauscht werden, ohne große Änderungen am restlichen System vornehmen zu müssen. Der grundlegenden Aufbau dieser Module ist in Abbildung 3.7 dargestellt. Da der Prozess der Analyse einer Musiksammlung

die meiste Zeit in Anspruch nimmt und diese Daten als repräsentative Quelle eines Genres dienen, wird die Speicherung in einer Datenbank angedacht, um so leichten und effizienten Zugriff der Daten für mehrere Benutzer an verschiedenen Orten bereit stellen zu können. Als Ausgangspunkt für das Erzeugen neuer Kompositionen können entweder MIDI-Dateien oder bereits in der Datenbank vorhandene Genre verwendet werden. Somit ist es jedem Benutzer möglich, neue (Sub-)Genres zu erstellen und diese dann für neue Kompositionen zu verwenden oder direkt die Datenbank als Ausgangspunkt zu benutzen. Abbildung 3.8

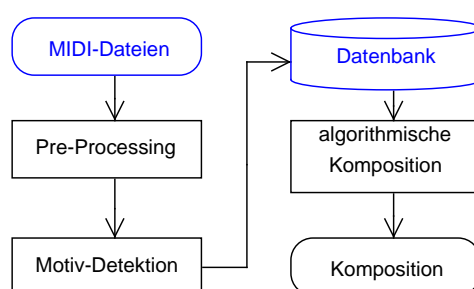


Abbildung 3.7: Module des PBMC

zeigt einen Ausschnitt aus dem Datenbankmodell zur Speicherung der Motive und der relevanten Daten eines Genres. Ein Motiv besteht dementsprechend aus mindestens einem Takt, welcher wiederum aus mindestens einer Note besteht. Motive, welche nur aus einem Takt bestehen sind monophone Motive. Im Gegensatz dazu bestehen polyphone Motive aus mehreren Takten. Zu jedem Motiv wird außerdem die jeweilige Taktart, die Tonart und das ursprüngliche Instrument der Phrase, aus welchem das Motiv extrahiert wurde, gespeichert. Weiter ist jedes Motiv genau einer Kategorie zugeordnet. Außerdem beinhalten Kategorien Listen an Tonarten, Instrumenten, Notenhäufigkeiten und Taktübergängen, welche durch die Analyse der einzelnen Stücke erhalten wurden. Die in diesem Kapitel beschriebenen Konzepte werden in den folgenden Kapitel umgesetzt, beginnend mit der Genre-Analyse, gefolgt von der genre-basierten Komposition.

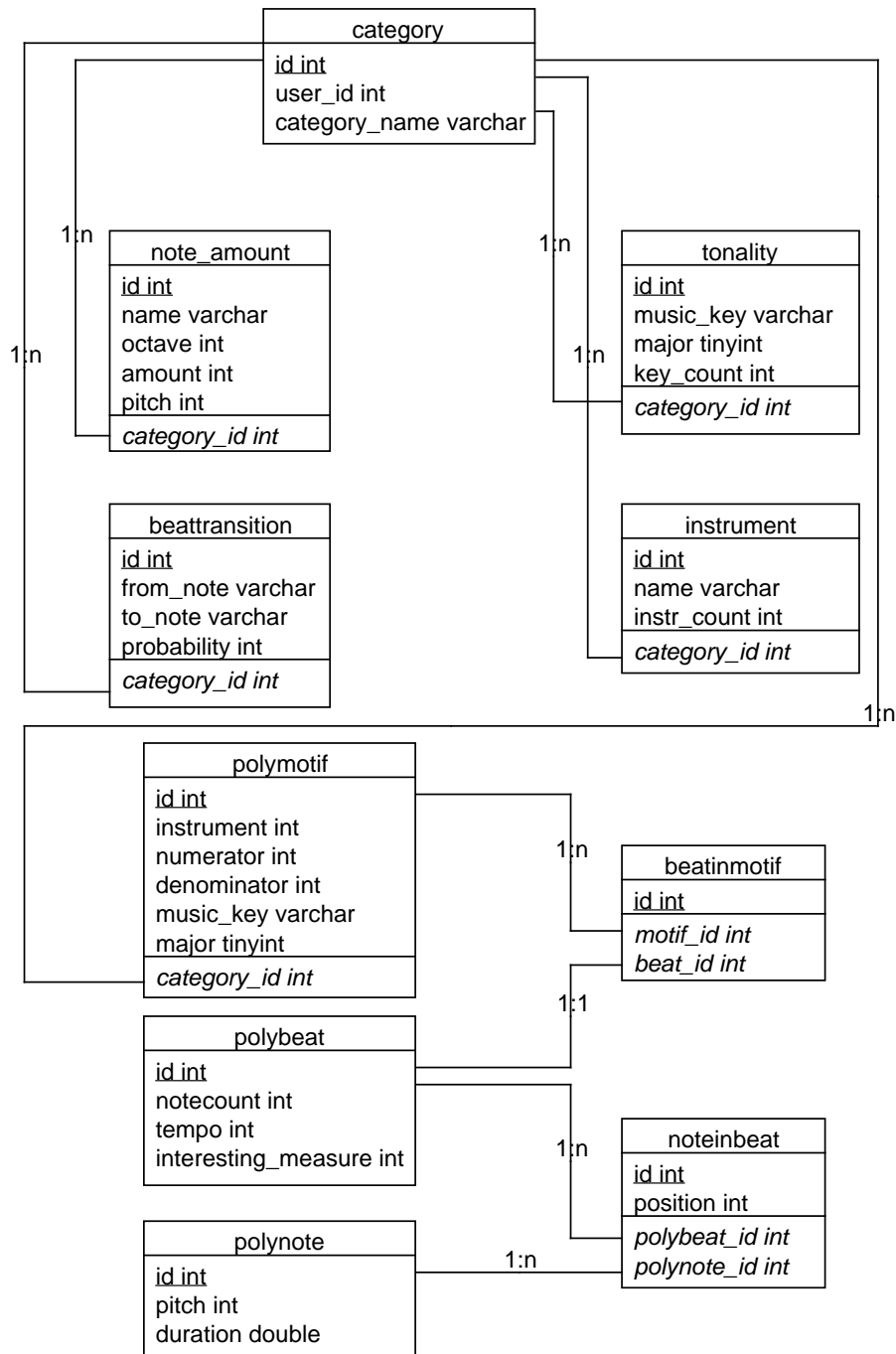


Abbildung 3.8: Ausschnitt aus dem Datenbankmodell



Genre-Analyse

Um genrespezifische Kompositionen erzeugen zu können ist der erste erforderliche Schritt ein Set an Eigenschaften zu generieren, welches ein Genre genauer beschreibt und von anderen Genres abgrenzt. In diesem Kapitel wird genauer auf die Ziele der Analyse der MIDI-Daten eingegangen, gefolgt von einer Beschreibung der diversen Attribute, welche eben diese Abgrenzung ermöglichen. Wie in den Grundlagen bereits erwähnt ist beim MIDI-Standard der Kanal 10 für Percussion reserviert. Da dieser auf Grund seines Aufbaus und seiner Funktionsweise die Ergebnisse der meisten Eigenschaften verfälschen würde ist zu beachten, dass er von der Analyse ausgenommen wurde.

4.1 Ziel

Das Ziel des Analyseprozesses besteht darin, mehrere Eigenschaften zu erhalten, welche ein Genre beschreiben. Es ist hierbei wichtig, nicht zu viele Attribute zu wählen, um die späteren Kompositionen, welche mit Hilfe der gewonnenen Daten erstellt werden, nicht zu sehr einzugrenzen und dadurch die Vielfalt der Ergebnisse einzuschränken. Eine weitere

Anforderung an die Analyse ist, dass diese in einem adäquaten zeitlichen Rahmen erfolgt und dass deren Ergebnisse für spätere und mehrmalige Verwendung gespeichert werden können.

4.2 Methode

Der Analyseprozess in GBMC gliedert sich im Wesentlichen in drei Teile:

1. Quantisierung der MIDI-Dateien
2. Analyse der Musiksammlung in Form von globalen Kriterien
3. Analyse der einzelnen Lieder innerhalb einer Sammlung in Form von lokalen Kriterien, wobei diese Bezug auf die globalen Kriterien nehmen

Die Analyse der lokalen Kriterien behandelt nur Notenfolgen mit der Länge eines Takts, wobei die aktuelle Taktlänge vom zu analysierenden Lied abhängig ist. Andere Algorithmen verwenden üblicherweise kurze Notenfolgen variabler Länge. Da es nicht notwendig ist alle in einem Lied enthaltenen Motive zu detektieren wurde die Länge eines Takts vom Autor als ausreichend empfunden. Diese trägt zu einer enormen Performancesteigerung bei. Gleichzeitig wird eine ausreichende Anzahl an erhaltenen Motiven gewährleistet.

Weiter ist zu beachten, dass die Ergebnisse der Analyse stark von der Zusammenstellung der gewählten Musiksammlung abhängig sind. Zwei unterschiedliche Musiksammlungen, welche trotzdem dasselbe Genre abbilden, können sehr divergierende Sets an Eigenschaften liefern. Wie Abbildung 4.1 zeigt beginnt der Analyseprozess mit einer Quantisierung der MIDI-Dateien. Diese ist erforderlich, da MIDI-Dateien sowohl am Computer mit exakten Noten- und Taktdauern erzeugt werden können, als auch mittels Eingabegeräten wie Synthesizern. Dabei kommt es automatisch zu ungenauen beziehungsweise schwankenden Notendauern und Pausen, welche bei der Quantisierung exakten Dauern zugeordnet werden.

Zudem erfolgt in diesem Schritt eine Anpassung der Länge der einzelnen Spuren der MIDI-Dateien. Auf Grund des Aufbaus von MIDI kommt es häufig vor, dass die erste Note einer Spur, welche erst im späteren Verlauf des Lieds erklingt, einen anderen Startzeitpunkt hat als die erste Note einer anderen Spur. Außerdem kann es vorkommen, dass eine Spur vorzeitig endet, wenn diese für den restlichen Verlauf des Lieds nicht mehr benötigt wird. Um etwaigen Fehlern und Komplikationen vorzubeugen und die spätere Unterteilung in Takte zu erleichtern werden deshalb alle Spuren auf dieselbe Länge gebracht, indem bis zur ersten Note und nach der letzten Note eine Pause eingefügt wird und somit alle Spuren die gleiche Länge besitzen. Nach der Quantisierung erfolgt die Analyse der globalen, gefolgt von der Analyse der lokalen Kriterien, welche nachfolgend im Einzelnen beschrieben werden. Es wurde darauf geachtet eine ausreichende Anzahl an Attributen zu verwenden, ohne dabei zu viele negative Auswirkungen auf die Vielfalt der Kompositionen zu erhalten. Die nachfolgend vorgestellten Eigenschaften wurden auf Grundlage musikalischer Überlegungen vom Autor gewählt und im weiteren Verlauf der Arbeit erweitert.

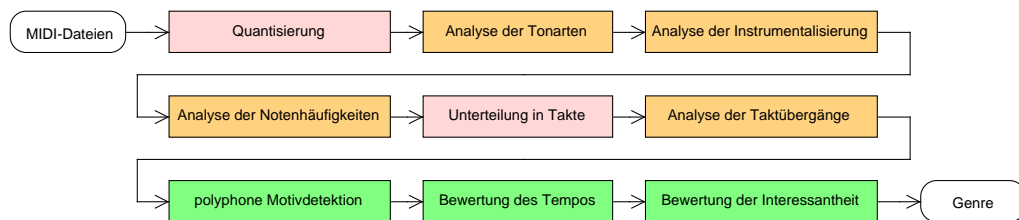


Abbildung 4.1: Analyse eines Genres

4.2.1 Globale Kriterien

Die globalen Kriterien umfassen die Eigenschaften der Musiksammlung, welche den Charakter des Genres widerspiegeln, einige Rahmenbedingungen für die entstehenden Kompositionen liefern und Bewertungsfaktoren für die lokalen Kriterien bereitstellen.

Notenhäufigkeiten

Die Notenhäufigkeit bezeichnet das prozentuale Auftreten einer bestimmten Note innerhalb der analysierten Musiksammlung. Mit Hilfe dieser Eigenschaft wird festgestellt, welche Noten charakteristisch für das Genre sind. Wie man in Abbildung 4.2, welche die Analyse der Notenhäufigkeiten beispielhafter Musiksammlungen von Kinderliedern und von Hip Hop Liedern zeigt, erkennen kann, bestehen mehr als ein Drittel aller Noten des Genres Kinderlieder aus einem G oder aus einem A, wobei der Großteil auf der 3. Oktave gespielt wird. Die Noten G# und A# werden kein einziges Mal in diesem Genre verwendet. Im Gegensatz dazu wird im Genre Hip Hop die Note G# von der 1. bis zu 4. Oktave sehr häufig verwendet. Man kann an diesem Beispiel sehr gut erkennen, dass die beiden Genres grundlegende Unterschiede der Notenhäufigkeiten aufweisen. Die Noten mit einem Anteil von über 5% werden vom System als charakteristisch für das jeweilige Genre bewertet und dienen in der weiteren Verarbeitung als Entscheidungskriterium für die Interessantheit eines genre-spezifischen Motivs.

Octave	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
-1	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
0	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
1	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
2	0%	0%	0%	1%	0%	0%	0%	1%	0%	1%	0%	1%
3	6%	2%	9%	0%	8%	5%	6%	13%	0%	11%	0%	3%
4	4%	0%	6%	0%	3%	2%	3%	3%	0%	6%	0%	1%
5	0%	3%	4%	0%	3%	0%	3%	0%	0%	1%	0%	0%
6	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
7	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
8	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
9	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

a)

Octave	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
-1	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
0	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
1	0%	0%	1%	1%	3%	2%	0%	0%	7%	0%	1%	2%
2	0%	2%	2%	1%	3%	2%	3%	0%	7%	1%	1%	5%
3	3%	2%	4%	5%	2%	1%	3%	6%	7%	2%	3%	3%
4	3%	5%	1%	1%	1%	4%	3%	4%	6%	3%	2%	0%
5	5%	1%	5%	1%	0%	5%	1%	2%	0%	2%	2%	1%
6	0%	0%	0%	0%	1%	0%	0%	0%	0%	0%	0%	1%
7	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
8	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
9	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

b)

Abbildung 4.2: Notenhäufigkeiten von a) Kinderliedern und b) Hip Hop

Instrumentalisierung

Eine weitere wichtige Eigenschaft eines Genres ist die charakteristische Instrumentalisierung eines Stücks. Im Bereich der automatischen Genredetektion ist dies eines der grundlegenden Attribute [13, 11]. Denkt man zum Beispiel an Jazz, empfindet man oft das Saxophon und die Trompete als typisch, bei Kirchenmusik die Orgel, bei Rock die E-Gitarre und bei Klassik das Klavier. Die richtige Auswahl an Instrumenten trägt daher maßgeblich zur Wiedererkennung eines Genres bei. Zu diesem Zweck werden im System die Instrumente jeder Spur ausgewertet und in vom MIDI-Standard vorgegebene Instrumentengruppen eingeteilt. In Abbildung 4.3 sieht man eine Gegenüberstellung der zwei Beispielgenres Kinderlieder und Hip Hop und deren unterschiedlicher Instrumentalisierung. In der Spalte *Count* sieht man die absolute Anzahl an Vorkommen der Instrumente der analysierten Stücke, zusammengefasst in Instrumentengruppen nach dem MIDI-Standard. Kinderlieder verwenden häufig ein Piano, eine Gitarre und ein Ensemble und verzichten auf Percussions, synthetischen Klänge und Soundeffekte. Im Gegensatz dazu werden im Genre Hip Hop alle Instrumentengruppen eingesetzt und die Verwendung von Percussions ist beinahe unumgänglich.

a)		b)	
Instrument Group	Count	Instrument Group	Count
Piano	20	Piano	81
Chromatic Percussion	5	Chromatic Percussion	21
Organ	9	Organ	17
Guitar	14	Guitar	71
Bass	1	Bass	84
Strings	6	Strings	43
Ensemble	12	Ensemble	49
Brass	0	Brass	32
Reed	1	Reed	35
Pipe	5	Pipe	31
Synth Lead	0	Synth Lead	27
Synth Pad	1	Synth Pad	10
Synth Effects	0	Synth Effects	9
Ethnic	0	Ethnic	7
Percussive	6	Percussive	12
Sound Effects	0	Sound Effects	9
Percussions	1	Percussions	1048

Abbildung 4.3: Instrumentalisierung von a) Kinderliedern und b) Hip Hop

Tonarten

Als weitere Eigenschaft hat auch die Tonart eines Lieds einen wesentlichen Einfluss auf den Klang und die Stimmung eines Lieds. Die Entwickler von CBS, einem konzeptbasierten Sequenzer zur Soundtrackkomposition verwenden in Verbindung mit einer passenden Instrumentenwahl das Tongeschlecht Dur für fröhliche Passagen und das Tongeschlecht Moll für traurige Passagen [20]. Dies wird durch die Analyse in Abbildung 4.4 bestätigt. Im Genre der Kinderlieder, welche mehrheitlich einen fröhlichen Charakter besitzen, werden die Lieder fast ausschließlich im Tongeschlecht Dur gespielt, wobei die Grundtonart C-Dur den größten Anteil besitzt. Das Genre Hip Hop verwendet im Gegensatz dazu alle Tonarten, wobei der Großteil der analysierten Lieder im Tongeschlecht Moll angesiedelt ist. Die Ton-

Key	Count	Key	Count
Major C	11	Minor C	0
Major C#	0	Minor C#	0
Major D	7	Minor D	0
Major D#	0	Minor D#	0
Major E	2	Minor E	0
Major F	8	Minor F	0
Major F#	1	Minor F#	0
Major G	8	Minor G	0
Major G#	0	Minor G#	0
Major A	4	Minor A	1
Major A#	0	Minor A#	0
Major B	0	Minor B	0

a)

Key	Count	Key	Count
Major C	4	Minor C	8
Major C#	2	Minor C#	2
Major D	4	Minor D	9
Major D#	3	Minor D#	5
Major E	4	Minor E	11
Major F	1	Minor F	4
Major F#	2	Minor F#	4
Major G	4	Minor G	9
Major G#	2	Minor G#	3
Major A	1	Minor A	10
Major A#	2	Minor A#	4
Major B	2	Minor B	6

b)

Abbildung 4.4: Tonarten von a) Kinderliedern und b) Hip Hop (Major = Dur, Minor = Moll)

art eines Lieds ist ohne konkrete Angabe aber nicht immer zuverlässig zu bestimmen. Oft kann sie nur geschätzt werden. Zu diesem Zweck wurde von Carol Krumhansl und Mark Schmuckler der nach ihnen benannte Krumhansl-Schmuckler-Algorithmus entwickelt [36]. Es handelt sich hierbei um einen Algorithmus, welcher 24 Tonartprofile verwendet, 12 für jede Dur-Tonart und 12 für jede Moll-Tonart. Jedes Tonartprofil besteht aus einem Vektor

mit 12 Werten, welche den 12 verschiedenen Noten einer Tonart entsprechen. Diese Profile entstanden im Zuge von Experimenten von Krumhansl und Kessler, in welchen Probanden einzelne Töne bestimmten Tonarten zuordnen mussten. Ein hoher Wert in diesem Profil bedeutet daher, dass der Ton sehr gut zur angegebenen Tonart passt, ein geringer Wert das Gegenteil. Das Profil der C-Dur-Tonart ist in Abbildung 4.5 dargestellt. Zur Schätzung der Tonart eines Lieds benötigt man nur mehr einen Vektor mit den jeweils aufsummierten Dauern der zwölf verschiedenen Noten, wobei dieser auf Grund des Aufbaus einer MIDI-Datei sehr leicht zu erstellen ist. Mit Hilfe des in 4.1 dargestellten Korrelationskoeffizienten berechnet man anschließend pro Tonart die Übereinstimmung zu den Features. Der Wert x entspricht den Werten des Vektors mit den aufsummierten Dauern, \bar{x} steht für den Durchschnitt dieser Werte, y bezeichnet die Werte des aktuellen Tonartprofils und \bar{y} beinhaltet den Durchschnitt der y-Werte [36].

$$r = \frac{\sum(x - \bar{x})(y - \bar{y})}{\sqrt{\sum(x - \bar{x})^2 \sum(y - \bar{y})^2}} \quad (4.1)$$

Schlussendlich wird die Tonart des Profils mit dem höchsten Korrelationskoeffizienten als korrekte Tonart angesehen [36].

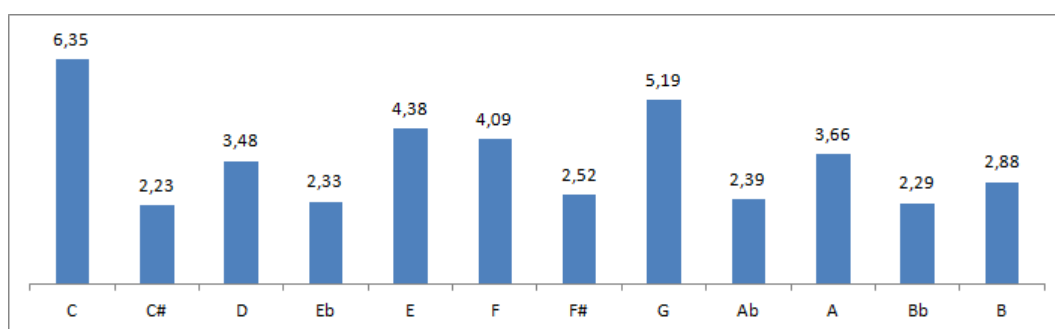


Abbildung 4.5: Profil der C-Dur-Tonart [36]

Taktübergänge

Das letzte globale Kriterium behandelt die Taktübergänge innerhalb eines Lieds. Mit Hilfe dieser Eigenschaft ist es später im Kompositionsprozess möglich, auf Basis der letzten Note des aktuellen Takts zu entscheiden, mit welcher Note ein neuer Takt beginnen soll. Dies schließt zwar generell keine Takte aus, macht es aber unwahrscheinlich, dass gewisse Takte als Nachfolger herangezogen werden. Zu diesem Zweck wurde die übliche Vorgangsweise der Speicherung der Wahrscheinlichkeiten in Matrizen, welche vom Problem der Auswahl der nächsten Note bekannt ist, für den taktbasierten Ansatz übernommen. Abbildung 4.6 zeigt zwei Beispiele dieser Matrizen, welche sich deutlich von einander unterscheiden. In der Tabelle der Übergangswahrscheinlichkeiten der Kinderlieder gibt es keine klare Struktur der Taktübergänge. Die Wahrscheinlichkeiten sind über die gesamte Matrix verteilt, wobei es mehrere Kombinationen gibt, welche niemals verwendet werden. Die Tabelle der Übergangswahrscheinlichkeiten für das Genre Hip Hop weist hingegen eine klare Struktur auf. Die meisten Übergänge in diesem Genre passieren bei gleichbleibender Note. Bis auf drei Ausnahmen kamen aber alle möglichen Taktübergänge bei der Analyse vor.

4.2.2 Lokale Kriterien

Die lokalen Kriterien umfassen die Eigenschaften der einzelnen Lieder, wobei der wichtigste Bestandteil das Motiv ist. Die beiden weiteren Kriterien berechnen sich aus dem Motiv und dienen dessen Kategorisierung.

Motive

Wie schon in den Grundlagen erwähnt ist das musikalische Motiv die kleinste sinntragende Einheit eines Lieds und für den späteren Kompositionsprozess von höchster Bedeutung. Auf Grund der geringen Anzahl (beziehungsweise zu spezieller) Algorithmen im Bereich

a)

nach von	C	C#	D	Eb	E	F	F#	G	Ab	A	Bb	B
C	23	0	7	0	5	26	1	13	0	15	5	5
C#	0	19	4	1	16	2	28	1	1	13	7	7
D	7	5	36	0	8	1	5	14	0	12	2	11
Eb	0	12	8	23	0	4	8	15	12	0	0	19
E	16	7	15	0	20	5	4	8	2	20	0	3
F	13	2	15	4	11	17	2	17	0	8	10	2
F#	3	16	6	0	11	1	16	15	9	15	1	6
G	16	2	16	0	8	8	4	26	0	14	1	5
Ab	0	5	0	0	27	2	5	7	16	23	7	9
A	7	6	11	0	9	4	7	12	4	28	6	6
Bb	24	2	0	4	11	16	11	6	2	11	8	5
B	6	4	9	1	11	1	3	23	9	13	3	16

b)

nach von	C	C#	D	Eb	E	F	F#	G	Ab	A	Bb	B
C	43	3	5	9	7	6	3	8	2	5	4	7
C#	3	40	8	9	1	1	9	6	7	3	10	4
D	14	3	47	3	5	5	2	8	0	7	3	3
Eb	6	4	10	48	1	6	5	3	6	1	8	2
E	5	2	4	3	46	7	8	7	1	8	1	7
F	9	2	4	11	6	43	2	10	4	2	7	2
F#	4	6	2	2	9	6	38	4	13	6	4	5
G	8	0	9	5	6	6	5	35	4	14	6	2
Ab	2	3	0	6	2	6	8	2	56	3	5	6
A	9	1	11	1	6	4	4	12	2	43	3	5
Bb	10	4	3	7	1	5	4	5	8	3	45	5
B	7	9	3	1	8	1	6	3	3	4	2	53

Abbildung 4.6: Taktübergänge von a) Kinderliedern und b) Hip Hop

der polyphonen Motivdetektion, wurde ein eigener Algorithmus entwickelt, welcher in der Lage ist, die Detektion und Extraktion innerhalb eines MIDI-Parts durchzuführen. Ziel dieses Algorithmus ist es, Motive in der Form von Wiederholungen, Sequenzen, Krebse und Spiegel zu detektieren, welche die Länge von genau einem Takt besitzen. Die einzelnen Schritte des Algorithmus werden nachfolgend grob erläutert und sind auch in Abbildung 4.7

dargestellt. Diese gelten sowohl für monophone als auch für polyphone Motive, wobei bei polyphonen Motiven ein zusätzlicher Zwischenschritt notwendig ist.

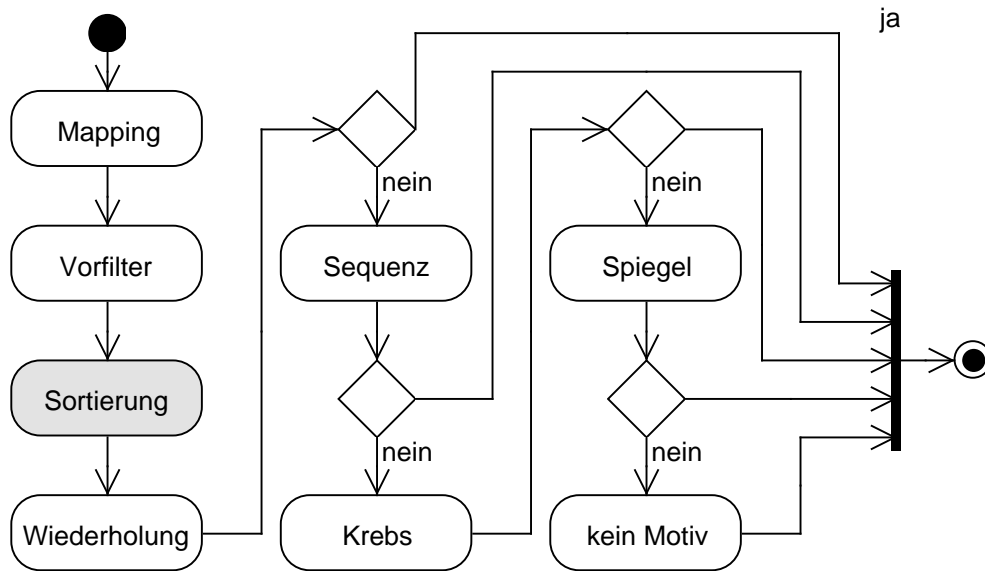


Abbildung 4.7: grober Ablauf der Motividetektion

1. Zuerst erfolgt pro Takt ein Mapping der Notendauern auf Buchstaben. Zum Beispiel hätte ein 4/4-Takt, welcher abwechselnd eine Viertel und zwei Achtel beinhaltet, danach die Form VAAVAA und ein Takt bestehend nur aus Vierteln die Form VVVV.

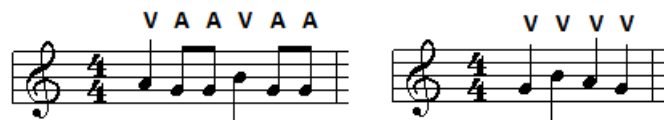


Abbildung 4.8: Mapping von Notendauern auf Zeichen

2. Danach erfolgt eine Vorfilterung der Takte. Hierbei wird mit Hilfe eines Stringvergleichs untersucht, ob es möglich wäre, von einem Takt, welcher als Vorlage dient, eine der gesuchten Motivarten zu detektieren. Zum Beispiel ist es bei einer Taktvorlage, welche aus lauter Viertelnoten (VVVV) besteht möglich, bei einem Vergleich mit einem Takt, der ebenfalls dieses Mapping besitzt, jede der vier gesuchten Motivarten zu detektieren. Bei einer Vorlage der Form VVA AV wäre bei einem Vergleichstakt derselben Form eine Wiederholung, eine Sequenz oder ein Spiegel möglich, aber kein Krebs. Bei einer Form des Vergleichstakts von VAAVV ist ein Krebs eine potenzielle Motivdetektion. Auf Grund dieser Vorfilterung wird eine Vielzahl von Takten bereits im 1. Schritt für eine positive Detektion ausgeschlossen und muss im 2. Schritt nicht mehr untersucht werden. Dies führt zu einer deutlichen Performancesteigerung.

3. Der Zwischenschritt der Sortierung, welcher in Abbildung 4.7 grau hinterlegt ist, ist nur bei polyphonen Motiven erforderlich. Polyphone Takte bestehen innerhalb einer MIDI-Datei aus mehreren monophonen Takten, welche von demselben Instrument gespielt werden und sich innerhalb des gleichen Parts befinden. Auf Grund der Speicherung der Noten in den verschiedenen Phrasen kann es vorkommen, dass zwei gleiche polyphone Takte unterschiedlich gespeichert werden. Um dieses Problem besser zu verdeutlichen, wird es in Abbildung 4.9 mit einem Beispiel veranschaulicht. Der polyphone Takt auf der linken Seite der Abbildung kann sich aus den drei monophonen Takten rechts oben oder aus den drei monophonen Takten rechts unten zusammensetzen, wobei der überlagerte polyphone Ton immer der gleiche ist. Um zu verhindern, dass beim Vergleich polyphoner Takte eine Abweichung detektiert wird, obwohl die Takte eigentlich gleich wären, werden polyphone Takte pro Notenposition innerhalb des Takts absteigend sortiert. Dies entspricht einer Aufteilung der monophonen Takte wie in Abbildung 4.9 rechts oben und gewährleistet somit eine korrekte Ver-

gleichbarkeit.

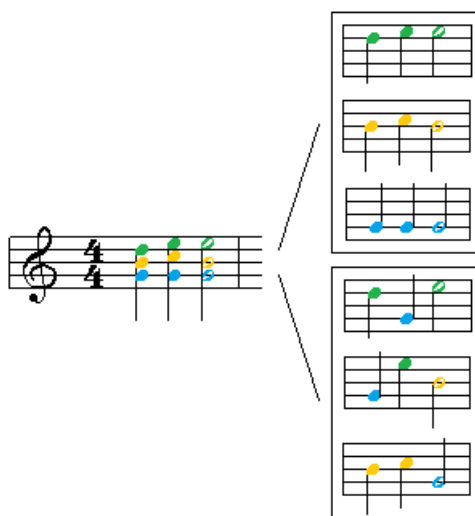


Abbildung 4.9: Speicherung polyphoner Takte

4. Anschließend werden die möglichen Motivvorkommen genauer untersucht, indem die Tonhöhen der einzelnen Noten miteinander verglichen werden. Hierfür wird der erste Takt als Vorlage gewählt und mit allen vorgefilterten Takten nacheinander verglichen. Wird der aktuell untersuchte Takt als Wiederholung des Vergleichstakts erkannt, so wird dieser markiert und bei späteren Vergleichen ignoriert. Handelt es sich nicht um eine Wiederholung, besteht die Möglichkeit einer Detektion einer Sequenz. Falls der aktuell untersuchte Takt danach noch keinem Motivtyp zugeordnet wurde, wird er noch auf die Möglichkeit eines Krebs und eines Spiegels hin untersucht. Fällt er in keine der vier Kategorien, so handelt es sich bei dem Takt um keine Motivvariation des Vergleichstakts. Auf Grund der Markierung der Takte verringert sich die Anzahl der zu vergleichenden Takte pro Durchlauf um die Anzahl der markierten Takte und den Vergleichstakt selbst, wodurch die Dauer eines Durchlaufs pro Runde verringert wird. In Abbildung 4.10 ist ein Beispiel einer durchgeführten Motividetek-

tion zu sehen. In der ersten Notenzeile sieht man rot markiert einen Krebs, gefolgt von einer blau markierten Sequenz, danach einer grün markierten Wiederholung und dazu zwei gelb markierte Sequenzen des ersten grünen Takts. Weiter folgt eine braun markierte Wiederholung, ein matt blau markierter Spiegel und ein weiterer Spiegel am Ende der Zeile violett eingerahmt.

Krebs	Wiederholung	Sequenz	Spiegel
Sequenz	Wiederholung	Sequenz	Spiegel

Abbildung 4.10: Beispiel einer Motividetektion

5. Wurden alle Takte analysiert, stehen diese nun in Form von acht Listen zur Weiterverarbeitung bereit; eine Liste pro Motivtyp jeweils für monophone und polyphone Motive.

Um die Ergebnisse des taktbasierten polyphonen Detektionsalgorithmus der Programmiererweiterung mit dem Algorithmus der ursprünglichen Version des PBMC zu vergleichen, wurden beide Algorithmen am selben System unter den gleichen Voraussetzungen getestet. Die monophone Motividetektion der ursprünglichen Version des PBMC verfolgt einen Brute-Force-Ansatz und ist in der Lage Motive variabler Länge zu detektieren. Es wurden beide Algorithmen auf Basis einer lokalen Datenbank und einer Musiksammlung bestehend aus 114 Countryliedern getestet. Die Ergebnisse des Tests werden in Tabelle 4.1 gezeigt. Wie man sieht beträgt die Anzahl der erkannten Motive beim taktbasierten Algorithmus nur einen Bruchteil des ursprünglichen Algorithmus, jedoch wird eine für den nachfolgenden

Algorithmus	Dauer	detektierte Motive
Brute-Force	12 h 17 min 5 sec	12.300.878
Taktbasiert	9 min 12 sec	5.406

Tabelle 4.1: Ergebnisse der Motivdetektion

Kompositionsprozess ausreichende Anzahl an Motiven detektiert. Die benötigte Rechenzeit des taktbasierten Algorithmus ist außerdem um ein Vielfaches geringer als die der Brute-Force Methode. Das eingangs erwähnte Ziel der Motivdetektion, genügend Motive als Grundlage für Kompositionen in einer adäquaten Zeit zu liefern, wird klar erreicht.

Tempo

Im Anschluss an die Motivextraktion der Musiksammlung erfolgt die Bewertung des Tempos der einzelnen Motive und die Kategorisierung in langsame, normale und schnelle Takte. Da die diversen Genres ein unterschiedliches Basistempe besitzen, ist die Bewertung eines Takts nur innerhalb des eigenen Genres gültig. Ein Takt eines Genres, welcher aus lauter Achtelnoten besteht wird in einem langsamen Genre als schnell bewertet. In einem Genre, in dem eine Achtelnote der durchschnittlichen Notendauer entspricht und schnelle Passagen aus Sechzehntelnoten bestehen, wird derselbe Takt hingegen als langsam bis normal bewertet. In Abbildung 4.11 werden Beispiele für jeweils einen langsamen, einen normalen und einen schnellen Takt der Genre Kinderlieder und Hip Hop gezeigt. Man kann sehr leicht erkennen, dass das Genre Kinderlieder einen langsameren Charakter als das Genre Hip Hop besitzt und die Takte daher unterschiedlich bewertet werden.

Interessantheitsmaß

Abschließend wird die Interessantheit jedes Motivs bewertet. Hierfür werden die anfangs erwähnten Notenhäufigkeiten benötigt. Beinhaltet ein Motiv eine oder mehrere Noten, wel-

	a) Kinderlieder	b) Hip Hop
langsam		
normal		
schnell		

Abbildung 4.11: Takte von a) Kinderliedern und b) Hip Hop Liedern

che charakteristisch für das Genre sind, so wird dieses für interessant befunden. Die Häufigkeiten der auftretenden Noten bestimmen dabei das Interessantheit des Motivs. Besitzt ein Motiv kein Vorkommen einer charakteristischen Note, führt dies zu einer schlechten Bewertung. Mit Hilfe des Attributs der Interessantheit kann man während des Kompositionsprozesses die am besten geeigneten Motive selektieren.

4.3 Ergebnisse

1. Die durch die Analyse erhaltenen Daten bilden ein Set bestehend aus mehreren Eigenschaften, welches versucht das analysierte Genre zu beschreiben. Mit Hilfe der globalen Kriterien wurden Rahmenbedingungen gesetzt, welche den Charakter des Genres widerspiegeln.
2. Durch die Extraktion der Motive wird die Grundlage für den folgenden Kompositionsprozess geschaffen. Die Motive sind der Ausgangspunkt bei der Entstehung einer neuen Komposition und ermöglichen es neue Takte zu generieren.
3. Es ist durch die Analyse später möglich, diverse Eigenschaften, wie zum Genre passende Tonarten und Instrumente beim Erstellen einer Komposition vorzugeben beziehungsweise vorzuschlagen.

- Die während der Analyse erhaltenen Daten werden in einer Datenbank abgelegt um mehreren Benutzern gleichzeitig Zugriff darauf zu geben. Außerdem muss dadurch eine Musiksammlung nur einmal analysiert werden und die Parameter stehen danach allen anderen Benutzern immer wieder für neue Kompositionen bereit.

Eine schematische Darstellung eines solchen Sets wird in Abbildung 4.12 gezeigt. Diese Sets stehen beim Kompositionsprozess zur Auswahl und geben die Rahmenbedingungen vor, um genre-spezifische Musikstücke zu erzeugen.

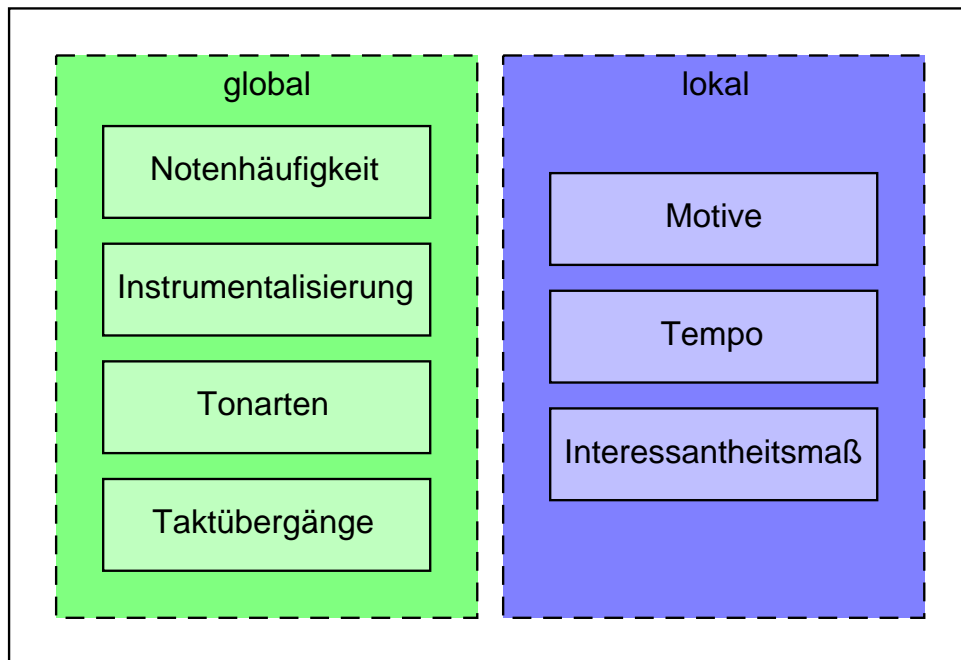
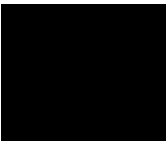


Abbildung 4.12: Set an Eigenschaften eines Genres



Genre-Basierte Komposition

Durch die Analyse der MIDI-Dateien wurden die benötigten Sets an Eigenschaften generiert, um genrespezifische Kompositionen zu erzeugen. Dieses Kapitel widmet sich der Erzeugung dieser Kompositionen. Es werden zunächst die Ziele erläutert, gefolgt vom grundlegenden Ablauf des Algorithmus zur Erzeugung einer neuen Komposition.

5.1 Ziel

Das angestrebte Ziel ist es, automatisch neue Kompositionen zu erzeugen, welche aus einer Hauptmelodie und diversen optionalen Begleitspuren bestehen. Die neuen Lieder sollen mit Hilfe der in der Analyse erhaltenen Eigenschaften erzeugt werden und dadurch den Charakter des Genres widerspiegeln. Wurde zu einem früheren Zeitpunkt die Musiksammlung, deren Charakter das neue Lied besitzen soll, analysiert, so ist es möglich auf die Daten der Analyse zurückzugreifen und direkt mit dem Kompositionsprozess zu beginnen. Außerdem soll Benutzern die Möglichkeit gegeben werden, die Ergebnisse der Kompositionen beeinflussen zu können.

5.2 Methode

Vor dem Start einer neuen Komposition sind zwar alle dafür benötigten Eigenschaften vorgegeben, es ist aber möglich, diese nach eigenen Vorstellungen abzuändern. Wie man in Abbildung 5.1 sehen kann, ist der Benutzer in der Lage, die Tonart des Stücks, die Anzahl, die Lautstärke und die Zuordnung der zu verwendenden Instrumente, sowie die Dauer und das Tempo der Komposition nach Belieben zu ändern. Außerdem ist es möglich, das Ergebnis des genetischen Algorithmus zu beeinflussen, indem man die Größe der Startpopulation, den Schwellwert der Tauglichkeitsfunktion und die Art der Taktgenerierung verändert. Die Starteinstellungen leiten sich aus der Häufigkeit des Vorkommens des jeweiligen Attributs innerhalb der analysierten Musiksammlung ab. Nachfolgend werden die einzelnen Schritte

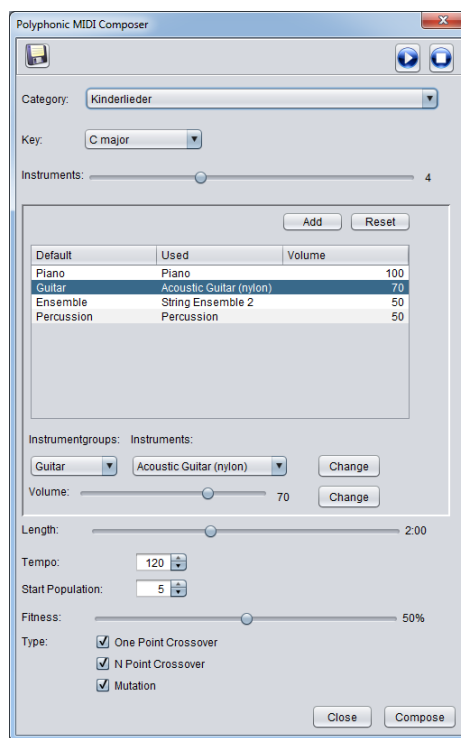


Abbildung 5.1: Kompositionsdialog des GBMC

zur Erzeugung einer neuen Komposition beschrieben. Ein grober Ablauf des Kompositionsprozesses ist in Abbildung 5.2 zu sehen.

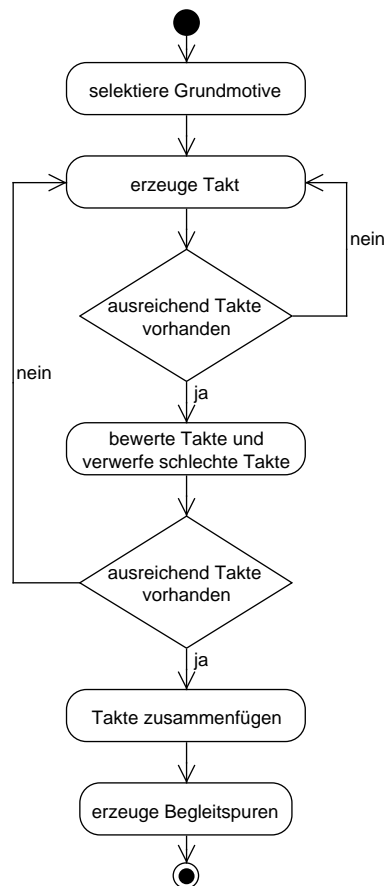


Abbildung 5.2: Schematischer Ablauf des Kompositionsprozesses

1. Zu Beginn wird das für die Hauptmelodie zu verwendende Instrument definiert. Da sich nicht jedes Instrument dafür eignet, eine Melodie zu spielen, wurde mit der Unterstützung eines Komponisten eine Zuordnung erstellt, welche MIDI-Instrumentengruppen dafür geeignet sind. Zum Beispiel übernimmt ein Ensemble selten das Spielen der Hauptmelodie, ebenfalls eignen sich Sound Effekte nicht für diese Aufgabe.

Die Auswahl erfolgt dementsprechend auf Basis einer vordefinierten Zuordnungstabelle.

2. Danach wird per Zufall entschieden, in welcher Taktart ($\frac{2}{4}, \frac{3}{4}, \frac{4}{4}$ etc.) die neue Komposition erzeugt wird. Hierfür wird ein Motiv der Zielkategorie zufällig ausgewählt und dessen Taktart für die neue Komposition übernommen. Auf Grund der Auswahl eines Motivs aus der Kategorie, welches die Parameter für die neue Komposition liefert, ist gewährleistet, dass eine zur Kategorie passende Taktart gewählt wird.
3. Anschließend erfolgt die Auswahl zufälliger Motive aus der gewählten Kategorie, welche als Grundpopulation für den genetischen Algorithmus dienen. Hierbei wird darauf geachtet, dass es sich, je nach Einstellung der gewünschten Tonart der Komposition, nur um Motive in Dur- beziehungsweise Moll-Tonalität handelt. Weiter werden Motive des Instruments, welches das Spielen der Hauptmelodie übernimmt in der Datenbank gesucht. Sind keine vorhanden, wird die Suche auf die Instrumentengruppe ausgeweitet.
4. In weiterer Folge werden die Motive auf ihre Grundtonart (C-Dur beziehungsweise A-Moll) und auf die gleiche Oktave transponiert, um sie zu normalisieren.
5. Nun erfolgt die Generierung der neuen Takte. Die Anzahl der benötigten Takte ergibt sich aus der gewählten Dauer und dem Tempo der neuen Komposition. Der Prozess der Erzeugung und Bewertung der Takte dauert so lange an, bis die erforderliche Anzahl erreicht wird. Zuerst wird per Zufall entschieden, welche der drei verschiedenen Takterzeugungsarten ausgewählt wird, sofern diese nicht zuvor vom Benutzer ausgewählt wurde. Mögliche Arten der Takterzeugung sind der One-Point-Crossover, der N-Point-Crossover und die Mutation. Bei der Mutation handelt es sich entweder um eine Transposition um sechs Halbtonschritte nach oben oder unten, einem zufäl-

ligen Vertauschen der Tonhöhen, einer aufsteigenden beziehungsweise absteigenden Sortierung der Noten oder der Erzeugung eines Krebses oder eines Spiegels. Die Verfahren des One-Point-Crossovers und des N-Point-Crossovers funktionieren wie im Grundlagenkapitel 2.4 beschrieben.

Die Auswahl der Takte erfolgt abermals per Zufall. Zur Auswahl stehen die Motive der Startpopulation und die bisher neu generierten Takte. Dadurch ist es möglich, dass bereits mutierte Takte einer weiteren Mutation unterzogen werden und neu entstandene Takte als Elternteil bei einem Crossover auftreten. In Abbildung 5.3 sieht man im oberen Teil des Bildes fünf Grundmotive des Genres Kinderlieder. Darunter werden beispielhaft einige generierte Takte dargestellt, welche mit ihren potenziellen Ursprungstakten markiert wurden.

Das Diagramm zeigt die Entstehung neuer Takte aus Grundmotiven. Die obere Reihe zeigt fünf Motive (a-e) in der G-Dur-Schlüsselart. Die untere Reihe zeigt generierte Takte, die als Kombinationen dieser Motive (z.B. b+a, b, d+a, d+a, d+b, a+e, e) dargestellt sind.

Abbildung 5.3: Entstehung neuer Takte

6. Im Anschluss an die Erzeugung werden die entstandenen Takte bewertet. Die Bewertungskriterien wurden großteils aus der ursprünglichen Version des PBMC übernommen und für die Bewertung polyphoner Takte angepasst [19]. Der Gütewert berechnet sich aus dem arithmetischen Mittel von sieben zwischen 0 und 1 normierten Bewertungskriterien.

- a) Das erste Kriterium gibt Auskunft über die Beziehungen der Noten innerhalb eines Takts, wobei die Tonhöhenänderungen von jeder Note und ihrem Nachfolger berechnet wird. Liegt die Änderung in einem Bereich von ein bis drei

Halbtönen, so wirkt sich dies positiv auf den Gütewert aus. Je höher die Abweichung von diesem Bereich ist, desto schlechter fällt das Ergebnis dieses Kriteriums aus.

- b) Das zweite Kriterium bewertet die Kontur des Takts, wobei eine auf- beziehungsweise absteigende Tonfolge besser bewertet wird, als ein gleichbleibender beziehungsweise schwingender Verlauf.
- c) Kriterium Nummer drei erzielt einen besseren Wert, je näher die Tonhöhen der ersten und letzten Note beieinander liegen.
- d) Da drastische Tempoänderungen meistens als störend empfunden werden, wird jeder Takt auf diese hin untersucht. Finden solche Tempoänderungen statt, erhält der Takt eine negative Bewertung.
- e) Endet ein Takt mit einer Notendauer schneller als eine Viertelnote, wird dieser als schlecht bewertet.
- f) Durch die bei der Mutation von Takten auftretende Tonhöhenänderung kann es vorkommen, dass sich Noten außerhalb der Grundtonarten C-Dur beziehungsweise A-Moll befinden. Je mehr Noten sich innerhalb der jeweiligen Grundtonart befinden, desto positiver fällt das Ergebnis in diesem Punkt aus.
- g) Das letzte Kriterium verwendet das in der Analyse beschriebene Interessanzmaß. Je mehr charakteristische Noten des Zielgenres im aktuellen Takt vorhanden sind, desto besser fällt die Bewertung aus.

Liegt der Durchschnitt aller sieben Faktoren des zu bewertenden Takts unterhalb des eingestellten Schwellwerts, so wird der Takt verworfen. Andernfalls verbleibt er in der Menge der für die Komposition zu verwendenden potenziellen Takte. Da in den ersten Durchläufen mit einer vom Schwellwert abhängigen Wahrscheinlichkeit viele

Takte verworfen werden, müssen die Schritte der Taktgenerierung und der Bewertung so lange wiederholt werden, bis eine ausreichende Anzahl an positiv bewerteten Takten generiert wurde.

7. Anschließend werden alle Takte, welche sich noch in der jeweiligen Grundtonart befinden, auf die gewünschte Tonart transponiert.
8. Nun werden die vorhandenen Takte zu einer Melodie zusammengefügt. Die Auswahl des ersten Takts der Komposition erfolgt wieder zufällig. Alle folgenden Takte werden vom aktuellen Takt ausgehend mit Hilfe der Taktübergangstabelle 4.6 gewählt. Basierend auf der letzten Note des aktuellen Takts wird per Wahrscheinlichkeitstabelle entschieden, mit welcher Note der neue Takt beginnen soll. Auf Grund der Wahrscheinlichkeiten innerhalb der Übergangsmatrix, werden passendere Anfangsnoten eher gewählt als weniger passende. Gänzlich unpassende Noten am Beginn eines Taktes können dadurch niemals auftreten, da sie eine Auftrittswahrscheinlichkeit von Null besitzen.
9. Nachdem die Erstellung der Hauptmelodie beendet ist, werden Begleitspuren für die gewählten Instrumente erzeugt. Wie einleitend erwähnt, wurde mit der Unterstützung eines Komponisten eine Zuordnung zu Instrumentengruppen getroffen, die festlegt, ob diese melodische und harmonische Begleitungen oder zufällige Töne spielen sollen. Darf ein Instrument sowohl die Aufgabe einer melodischen als auch harmonischen Begleitung übernehmen, so wird per Zufall über die Zuteilung entschieden. Eine melodische Begleitspur besteht aus der Hauptmelodie, welche um eine Oktave nach unten transponiert wird. Eine harmonische Begleitspur beinhaltet zur Melodie passende Akkorde, wobei auf jede ungerade Note der passende Grundakkord erklingt. Instrumente, welche zufällige Noten spielen dürfen, sind zum Beispiel Soundeffekte, Syntheffekte oder auch Schlaginstrumente. Weiter ertönt in der Perkussionspur

eine große Trommel auf jede Viertelnote des Takts. Im Bereich der automatischen Genre-Erkennung nimmt der Rhythmus neben der Instrumentalisierung zwar eine große Rolle ein. Auf Grund der Komplexität einer geeigneten Rhythmusgenerierung wird im Rahmen dieser Arbeit aber nicht im Detail darauf eingegangen [37]. Wie sich im folgenden Kapitel noch zeigen wird, wirkt sich dies aber negativ auf die Zuordbarkeit der Kompositionen zu Genres aus.

Zum Schluss werden alle Spuren in einer MIDI-Datei zusammengefasst. Der Prozess der Kompositionsgenerierung ist abgeschlossen.

5.3 Ergebnisse

Das Endprodukt des Kompositionsprozesses ist eine MIDI-Datei, welche das neu generierte Stück enthält. Der Kompositionsprozess ist meist nach wenigen Sekunden abgeschlossen und das Stück kann sofort wiedergegeben und für eine Weiterverarbeitung in anderen Programmen gespeichert werden. Ist man mit dem Ergebnis nicht zufrieden, verändert man einige der zuvor genannten Parameter und lässt ein neues Lied komponieren. Innerhalb weniger Versuche wurden vom Autor meist zufriedenstellende Ergebnisse erreicht.

Abbildung 5.4 zeigt die Hauptspur einer neuen Komposition, welche die Melodie beinhaltet und die fünf zu Grunde liegenden Motive, aus denen sie mit Hilfe des eben beschriebenen Algorithmus generiert wurde. Um bessere Lesbarkeit zu erhalten, wurde eine Komposition gewählt, welche rein aus monophonen Motiven erzeugt wurde. Betrachtet man die einzelnen Takte, kann man sehr gut erkennen aus welchen Basistakten diese abgeleitet wurden. Trotz der erhaltenen Ähnlichkeit zu den ursprünglichen Takten, wird in der Abbildung deutlich, welche Vielfalt an zufällig generierten neuen Takten aus der geringen Grundmenge an Motiven entstehen kann.

Das primäre Ziel dieser Arbeit ist, neue Kompositionen zu erzeugen, die Gemeinsamkeiten

mit bekannten Genres aufweisen und deren Charakter widerspiegeln. Um zu überprüfen ob dieses Ziel erreicht wurde, bedarf es einer Evaluierung.

Grundmotive:



Komposition (Melodie):



Abbildung 5.4: Grundmotive und Melodiespur einer neuen Komposition

Evaluierung

Ein Problem der automatischen Komposition, welches sich aus der Möglichkeit des schnellen Erzeugens einer großen Zahl neuer Stücke ergibt, ist, Musik als *gut* oder *schlecht* zu erkennen [17]. Hanna Järveläinen [22] meint, die einzige Möglichkeit eine Komposition zu bewerten, ist diese anzuhören. Um die Qualität und die Zuordbarkeit der erzeugten Stücke zu prüfen, wurden diese daher für die vorliegende Arbeit von einer Kontrollgruppe, bestehend aus musikalisch versierten Probanden, angehört und bewertet.

6.1 Ziel

Die geschmackliche Bewertung von Musikstücken ist ohne Zweifel eine subjektive Angelegenheit. Dennoch ist sie wichtig, um Aussagen über die Qualität des Composers treffen zu können. Je größer und vielfältiger das musikalische Interesse einer Person ist, desto eher ist diese in der Lage, die verschiedenen Kompositionen zu bewerten. Da aber nicht nur die Qualität der Stücke, sondern auch die Genre-Zuordbarkeit von Interesse ist, bedarf es an Fachleuten als Testpersonen, welche auch darüber eine korrekte Aussage tätigen können.

6.2 Methode

Für die Befragung wurden Personen im privaten Umfeld des Autors um Mithilfe gebeten, die über die genannten Eigenschaften verfügen. Jede Testperson erhielt die gleiche Sammlung an Kompositionen, welche mit Hilfe des GBMC erstellt wurden. Außerdem wurden die Kompositionen durchgehend mit den Standardeinstellungen des Composers erzeugt. Die Testsammlung bestand aus elf Liedern der sieben Ursprungsgenres *Alternative*, *Drum & Bass*, *Kinderlieder*, *Klassik*, *Pop*, *Punk* und *Rock*. Hierbei wurde versucht, eine breite Auswahl an Genres abzudecken, wobei aber zumindest manche Genres Ähnlichkeiten zu einander aufweisen sollten. Auf Grund der Dauer jeder Komposition von zwei Minuten und die daraus resultierende Befragungsdauer von mindestens 22 Minuten wurde davon abgesehen weitere Kompositionen für die Befragung heranzuziehen. Die verwendeten Lieder können unter https://www.dropbox.com/sh/x8qdqrlb9sthw0o/QueN_N-hSi heruntergeladen werden (zuletzt besucht am 16.04.2014). Jeder Teilnehmer erhielt eine Übersicht der genannten Genres und die Information, dass die zur Auswahl stehenden Genres mehrmals zutreffen können, wobei pro Lied genau ein Genre auszuwählen war. Danach wurden die Lieder abgespielt und die Probanden gebeten, eine Zuordnung zu einem Genre zu treffen. Weiter wurden die Testpersonen darum gebeten, auf einer fünfteiligen Likert-Skala anzugeben, wie gut ihnen die Komposition gefällt, wobei darauf aufmerksam gemacht wurde, den Aspekt der computergenerierten Musik außer acht zu lassen und nur die Qualität der Komposition an sich kritisch zu beurteilen, diese also mit von Menschen erzeugten Kompositionen vergleichen.

6.3 Ergebnisse

6.3.1 Teilnehmer der Befragung

Die Gruppe der 22 Befragten setzte sich aus 15 männlichen und 7 weiblichen Teilnehmern im Alter von 16 bis 35 Jahren (Durchschnitt: 26,2) zusammen. Wie man in Abbildung 6.1 erkennen kann, handelt es sich bei den Testpersonen um Berufsmusiker, Tontechniker, Mitgliedern von Bands, Komponisten, Musikschüler oder Musikliebhaber, welche ein breites Spektrum an musikalischen Genres abdecken. Von den 11 möglichen richtigen Zuordnungen wurden von jeder Testperson mindestens 2 (18%) und maximal 6 (56%) erkannt. Der Durchschnitt lag dabei bei 3,6 richtigen Antworten pro Testperson, was einer Erkennungsrate von 33% entspricht. Eine Zuordnung wurde als korrekt befunden, wenn das gehörte Stück dem zu Grunde liegenden Eigenschaftsset eines Genres, welches beim Kompositionsprozess verwendet wurde, zugeordnet wurde. Dies ist vor dem Hintergrund, dass die Instrumente von einem einfachen MIDI-Klangerzeuger und nicht durch spezielle Plug-Ins wiedergegeben wurden und daher mehr nach synthetisierten als nach realen Instrumenten klangen und die Kompositionen unabhängig vom Genre immer denselben gleichbleibenden Rhythmus hatten, aus Sicht des Autors eine sehr zufriedenstellende Quote. Laut Aussage mehrerer Testpersonen war die größte Hürde bei der Zuordnung der sich nicht ändernde Rhythmus. Auf diese Problematik wird bereits in Kapitel 5 aufmerksam gemacht. Wenn man bedenkt, dass gerade im elektronischen Musikbereich typische Rhythmen existieren, wodurch man sehr schnell eine Aussage zum Genre treffen kann und bei den zu bewertenden Kompositionen dieses Merkmal fehlte, wurde dadurch die korrekte Zuordnung erheblich erschwert.

Person #	Geschlecht	Alter	richtige	Musikbezug
1	m	29	3	Musikliebhaber
2	w	16	2	Musikliebhaberin
3	w	21	3	Musikschülerin
4	m	27	3	Musikschüler
5	m	22	5	Sänger und Gitarrist (Band)
6	m	30	3	Musikliebhaber
7	w	24	3	nebenberufliche Sängerin
8	m	30	2	Radioangestellter und Tontechniker
9	m	29	4	Musikschülerin
10	m	23	6	Komponist, Sänger und Gitarrist (Band und Solo)
11	w	23	4	nebenberufliche Sängerin
12	m	29	6	Musikliebhaber
13	m	29	2	Musikschüler
14	w	28	3	Musikliebhaberin
15	m	35	3	Berufssänger (Tenor)
16	m	31	4	Bassist (Band)
17	w	20	5	Musikschülerin und Chorleiterin
18	m	28	2	nebenberuflicher DJ
19	m	28	5	nebenberuflicher DJ und Komponist
20	w	22	4	Musikliebhaberin
21	m	27	3	Hobby DJ und Komponist
22	m	25	4	Tontechniker

	von 11
	>= 5
	3 - 4
	<= 2

Abbildung 6.1: Übersicht über die Testpersonen

6.3.2 Zuordnung zu einem Genre

Beim Betrachten der Abbildungen 6.2 und 6.3, welche die Zuordnung der Lieder durch die TeilnehmerInnen zu den Genres zeigt, fällt auf, dass bei sechs Liedern (56%) die Mehrheit der TeilnehmerInnen das Genre korrekt erkannt hat. Einige auffallende Zuordnungen von Liedern, beziehungsweise Genres, werden nachfolgend genauer erörtert.

- Lied #5 (Klassik) nimmt eine besondere Rolle ein, da es von fast jeder Testperson richtig zugeordnet werden konnte. Laut den TeilnehmerInnen war es hier am einfachsten, eine Zuordnung zu treffen, da die Instrumentenvielfalt nicht sehr hoch war und dadurch die Melodie besser zu Geltung kam.

- Betrachtet man die Lieder #6 und #8, welche beide auf dem Genre Punk basieren, so sieht man, dass beide Lieder häufig dem Genre Rock zugeordnet wurden. Dies beruht nach Meinung des Autors auf der Tatsache, dass sich beide Genres in ihrer Instrumentalisierung sehr ähnlich sind. Teilnehmer #18 behalf sich bei der Zuordnung, indem er auf Grund der schnellen kurzen Riffs eher zum Genre Punk tendierte.
- Da die Instrumentalisierung der Lieder #1 und #7 (Rock) einen elektronischen Charakter hatte, wurde bei beiden Liedern fälschlich Drum & Bass als korrektes Genre bestimmt. Ein Teilnehmer meinte, dass er diese Lieder als 8-Bit-Musik einstufen und sie deshalb dem Genre *Chiptune* zuordnen würde. Da dieses Genre aber nicht zur Auswahl stand, entschied er sich beide Male für das Genre Drum & Bass. Wäre für jedes Genre ein typischer Rhythmus vorhanden, hätte man diese Option mit hoher Wahrscheinlichkeit ausschließen können.

Lied / Genre	Alternative	Klassik	D&B	Kinderlieder	Pop	Punk	Rock	Summe
1	14		41	5	27	9	5	100
2	9			55	23		14	100
3	23	5	9	14	27	5	18	100
4	18		18		45	9	9	100
5		91			9			100
6	14					50	36	100
7	41		41		14	5	0	100
8	9					45	45	100
9	27	5	23	9	14	5	18	100
10	14		18		50		18	100
11		41	5	41	14			100

	korrekt zugeordnet
	korrekt
	max

Abbildung 6.2: Zuordnung der Kompositionen zu Genres in Prozent

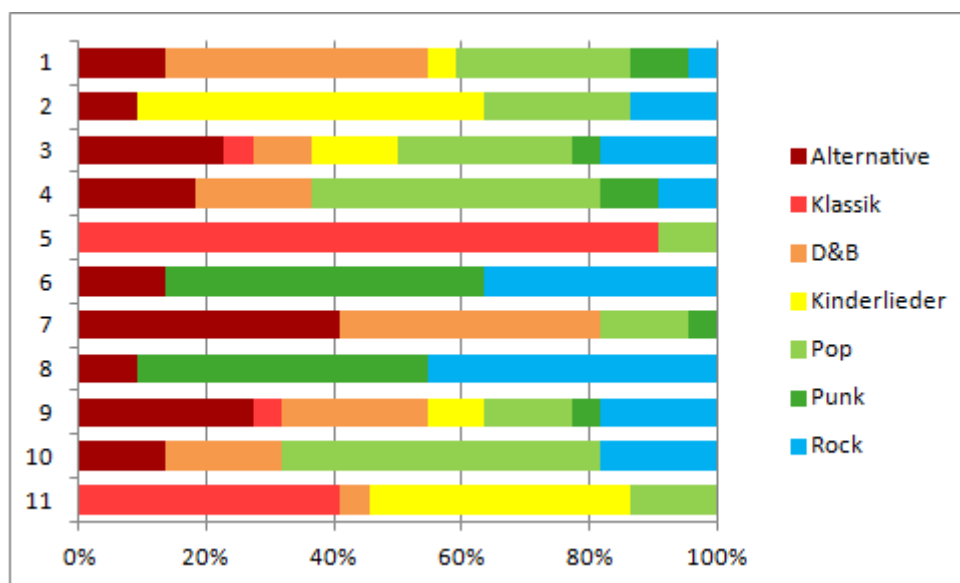


Abbildung 6.3: Zuordnung der Kompositionen zu Genres

6.3.3 Qualität der Kompositionen

Zum Schluss wurde noch ausgewertet, wie gut den Testpersonen die einzelnen Kompositionen gefallen haben. Wie man aus den Abbildungen 6.4 und 6.5 erkennt, wurden zwei Lieder tendenziell eher für gut befunden und drei Lieder für eher schlecht. Die Bewertungen der übrigen Kompositionen sind im Mittelfeld angesiedelt, wobei bei den einzelnen Stücken jeweils eine Tendenz zu einer Seite erkennbar ist. Es fällt auf, dass die Lieder der Genres Klassik (#5) und Kinderlieder (#2 und #11) besser bewertet wurden als andere Testlieder. Dies kann vermutlich auf den fröhlichen Charakter der Genres beziehungsweise der Kompositionen zurückgeführt werden. Ein Teilnehmer erwähnte mehrmals den kindlichen Charakter der Lieder des Genres Kinderlieder. Ein anderer meinte, dass ihn die Komposition an Super Mario (Anmerkung: bunte Computerspielfigur von Nintendo) erinnere. Im Gegensatz dazu wurden beide Lieder des Genres Rock (#1 und #7) negativ bewertet. Der Grund dürfte hier bei der Instrumentalisierung liegen. Ein Teilnehmer bezeichnete beide Lieder als

schnelles kurzes Gepiepe, welches sich für ihn nur sehr schwer einordnen ließe. Das dritte negativ beurteilte Lied gehört dem Genre Drum & Bass an, welches unter der Abwesenheit des für das Genre charakteristischen, aber in GBMC nicht vorhandenen, Rhythmus leidet. Insgesamt waren die TeilnehmerInnen begeistert von den Kompositionen, da die wenigsten überhaupt von der Möglichkeit wussten, Musik von einem Computer auf diese Weise generieren zu lassen. Auch der Autor ist mit den Kompositionen zufrieden. Er vertritt dabei die Meinung, dass es in jeder der erstellten Kompositionen wohlklingende und weniger wohlklingende Passagen gibt.

Lied / Bewertung	sehr schlecht	schlecht	neutral	gut	sehr gut	Summe
1	27	32	27	9	5	100
2			41	41	18	100
3	5	41	27	23	5	100
4		23	41	27	9	100
5	5	9	18	36	32	100
6	14	36	23	18	9	100
7	23	32	41	5		100
8	14	18	32	23	14	100
9	9	32	36	18	5	100
10		50	36	9	5	100
11	5	18	32	32	14	100

	>= 18%
	gut
	mittel
	schlecht

Abbildung 6.4: Bewertung der Kompositionen in Prozent

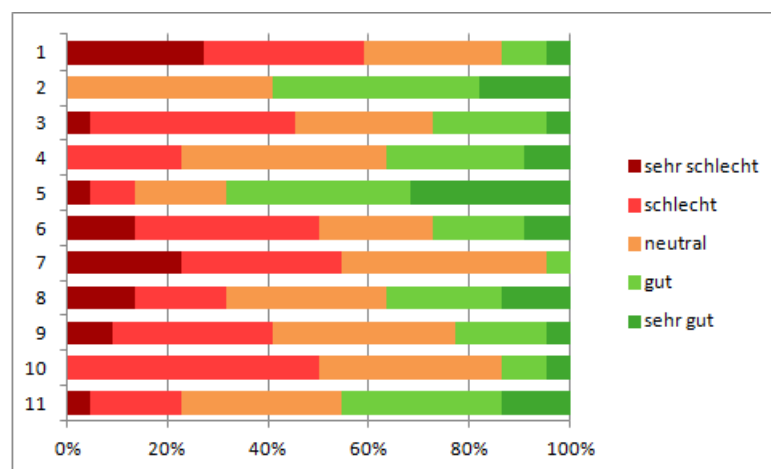


Abbildung 6.5: Bewertung der Kompositionen

Schlussfolgerungen und Ausblick

Die Evaluierung hat gezeigt, dass das hoch gesteckte Ziel der Wiedererkennbarkeit des Genres einer computergenerierten Komposition zum Teil erreicht wurde und der Algorithmus in der Lage ist, wohlklingende Melodien zu erzeugen. Nicht jede Komposition klingt gleich gut und es benötigt manchmal mehrere Versuche, bis ein zufriedenstellendes Ergebnis erzielt wird. Nach Abschluss der praktischen Arbeiten ergaben sich zudem noch einige Ideen für Verbesserungen, welche nachfolgend kurz erläutert werden. Des Weiteren werden einige der verwendeten Methoden reflektiert.

- Die Verwendung eines genetischen Algorithmus war für den Erfolg des Kompositionsprozess von wesentlicher Bedeutung. Durch die nicht deterministischen Ergebnisse, welche aber mit Hilfe von Tauglichkeitsfunktionen in eine gewünschte Richtung gelenkt werden können, erhält man eine sehr große Anzahl an verschiedenen Kompositionen. Weiter erlaubt es der Algorithmus, mit Hilfe verschiedener Parameter ebenfalls Einfluss auf das Ergebnis zu nehmen.
- Die Motivgröße *fix* mit der Länge eines Takts zu wählen, brachte erhebliche Perfor-

mance-Vorteile. Die Analyse der Lieder konnte um ein Vielfaches schneller als mit dem üblichen Brute-Force-Ansatz durchgeführt werden. Jedoch verliert man dadurch Motive mit der Länge kleiner oder größer als einem Takt. Bei der späteren Handhabung im Kompositionsprozess macht es keinen Unterschied, ob man einen Nachfolgetakt auf Grund einer Taktübergangsfunktion oder auf Notenebene mit einer Tabelle der Wahrscheinlichkeiten der nächsten Note auswählt. Die Menge an extrahierten Motiven war trotzdem ausreichend, um genügend Basismotive zur Erstellung neuer Kompositionen zu haben. Ein Nachteil der fixen Länge ist, dass man verschiedenen Taktlängen ($\frac{2}{4}, \frac{3}{4}, \frac{4}{4}$, etc) nicht miteinander mischen kann. Dieses Problem tritt bei Motiven variabler Länge nicht auf, da diese einfach aneinander gekettet werden können.

- Die in dieser Arbeit implementierte Motiverkennung umfasst vier verschiedene Arten von Motiven. Da aber noch weitere Motivarten, wie die Augmentation und Diminution existieren, wäre es sinnvoll, auch diese Motivarten zu detektieren zu versuchen. Da es sich hierbei um eine Streckung oder Stauchung der Motive handelt, muss überlegt werden, ob und wie man diese mit dem taktbasierten Ansatz detektieren kann.
- Da Musikstücke normalerweise aus einer Einleitung, abwechselnden Strophen und einem Refrains bestehen und mit einem Schlussteil enden, wäre es sinnvoll, mehrere Composer für jeweils diese Teile zu erstellen. Man könnte dann in weiterer Folge den Benutzer entscheiden lassen, wie der Verlauf der Intensität des Liedes sein soll. Dies könnte man durch vordefinierte Spannungsbögen, beziehungsweise selbst erstellte Kurven definieren. In Abbildung 7.1 werden drei solche Spannungsbögen beispielhaft dargestellt. Die in GBMC implementierte Bewertung des Tempos der Takte liefert bereits die ersten erforderlichen Voraussetzungen für dieses Konzept. Zusätzlich könnte man die Takte noch mit dem Attribut *aufsteigend*, *absteigend* oder *alternierend* bewerten, um ein weiteres Entscheidungskriterium zur Auswahl von Takten zu

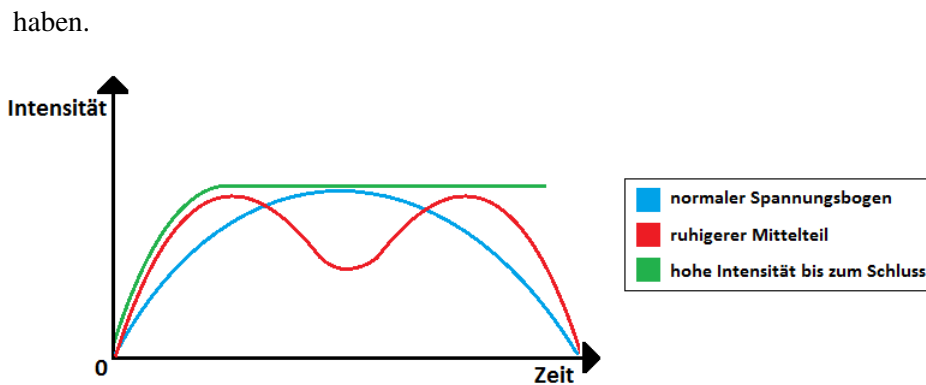


Abbildung 7.1: mögliche Spannungsbögen eines Liedes

- Eine weitere Verbesserungsmöglichkeit liegt in der Optimierung der Begleitspurgenerierung. Da Begleitungen in der aktuellen Version des GBMC entweder aus der Melodie um eine Oktave nach unten versetzt, aus Grundakkorden oder zufälligen Geräuschen bestehen, wäre es denkbar in Abhängigkeit zur Hauptmelodie unabhängige passende Begleitungen zu generieren. Diese könnten aus Motiven generiert werden, welche bei der Analyse von Begleitspuren erkannt wurden.
- Die Erzeugung von passenden Perkussionsspuren wäre, wie sich auch im Rahmen dieser Arbeit gezeigt hat, ein charakteristischer Bestandteil eines Liedes und würde die Zuordnung zu einem Genre enorm erleichtern. Es wäre daher ratsam Methoden zu entwickeln, um für jedes Genre typische Rhythmen zu detektieren und anschließend für Kompositionen zu verwenden.

Die von GBMC erzeugten Kompositionen eignen sich nach Meinung des Autors und einiger Testpersonen aber sehr gut dazu, Komponisten und Liedermacher mit neuen kreativen Melodien zu versorgen, welche als Ausgangspunkt für von Menschen erstellte Kompositionen dienen können. Da die Kompositionen des GBMC der zu Grunde liegenden Musiksammlung ähneln und diese auf Basis von Motiven dieser Sammlung erstellt werden, ist es dem

Benutzer möglich, die zu erzeugenden Melodien in eine gewünschte Richtung zu lenken und somit schneller inspirierende Melodien zu erhalten. Im Assisted Composing liegt daher nach Meinung des Autors der Haupteinsatzbereich von GBMC. Die Möglichkeit für einen Komponisten seine eigenen Werke zu analysieren und sich auf deren Basis ähnlich klingende Melodien generieren zu lassen kann dessen Arbeit erheblich erleichtern.

Literaturverzeichnis

- [1] Midi tech specs and info. <http://www.midi.org/techspecs>. [Online; aufgerufen am 23.03.2014].
- [2] Peter Ahrendt. *Music Genre Classification Systems - A Computational Approach*. PhD thesis, Technical University of Denmark, Kongens Lyngby, Denmark, 2006.
- [3] Daniel Aschauer. *Algorithmic Composition*. Jänner 2008.
- [4] Gérard Assayag, Shlomo Dubnov, and Olivier Delerue. Guessing the composer's mind: Applying universal prediction to musical style. 1999.
- [5] Georg Boenn, Martin Brain, Marina Vos, and John Fitch. Automatic composition of melodic and harmonic music by answer set programming. In Maria Garcia de la Banda and Enrico Pontelli, editors, *Logic Programming*, volume 5366 of *Lecture Notes in Computer Science*, pages 160–174. Springer Berlin Heidelberg, 2008.
- [6] Georg Boenn, Martin Brain, Marina De Vos, and John Fitch. Automatic music composition using answer set programming. *CoRR*, abs/1006.4948, 2010.
- [7] Pedro P. Cruz-Alcázar and Enrique Vidal-Ruiz. Learning regular grammars to model musical style: Comparing different coding schemes. In Vasant Honavar and Giora Slutzki, editors, *Grammatical Inference*, volume 1433 of *Lecture Notes in Computer Science*, pages 211–222. Springer Berlin Heidelberg, 1998.
- [8] Charles Darwin. *Über die Entstehung der Arten im Tier- und Pflanzen-Reich durch natürliche Züchtung*. 1860.
- [9] Alfonso Ortega de la Puente, Rafael Sánchez Alfonso, and Manuel Alfonseca Moreno. Automatic composition of music by means of grammatical evolution. *SIGAPL APL Quote Quad*, 32(4):148–155, June 2002.
- [10] Michael Edwards. Algorithmic composition: Computational thinking in music. *Commun. ACM*, 54(7):58–67, July 2011.

- [11] Slim Essid and Gaël Richard. Instrument recognition in polyphonic music based on automatic taxonomies. In *IEEE Transactions on Speech and Audio Processing*, pages 68–80, 2006.
- [12] L. Macedo A. Cardoso F. Pereira, C. Grilo. Composing music with case-based reasoning. 1997.
- [13] Ferdinand Fuhrmann. *Automatic musical instrument recognition from polyphonic music audio signals*. PhD thesis, Universitat Pompeu Fabra, 2012.
- [14] Liqiang Geng and Howard J. Hamilton. Interestingness measures for data mining: A survey. *ACM Comput. Surv.*, 38(3), September 2006.
- [15] Georg Holzmann. Genetische Algorithmen in Komposition und Computermusik. Dezember 2003.
- [16] Costas S. Iliopoulos, Kjell Lemström, Mohammed Niyad, and Yoan J. Pinzón. Evolution of musical motifs in polyphonic passages. In *Symposium on AI and Creativity in Arts and Science, Proceedings of AISB'02*, pages 67–76, 2002.
- [17] Bruce L Jacob. Algorithmic composition as a model of creativity. *Organised Sound*, vol. 1(no. 3):pp. 157–165, December 1996.
- [18] Bruce L. Jacob. Algorithmic composition as a model of creativity. *Org. Sound*, 1(3):157–165, December 1996.
- [19] Fellner Jakob. Pattern-based MIDI composing. Master's thesis, Technische Universität Wien, 2013.
- [20] M.O. Jewell, M.S. Nixon, and A. Prugel-Bennett. Cbs: a concept-based sequencer for soundtrack composition. In *Web Delivering of Music, 2003. 2003 WEDELMUSIC. Proceedings. Third International Conference on*, pages 105–108, 2003.
- [21] Aída Jiménez, Miguel Molina-Solana, Fernando Berzal, and Waldo Fajardo. Mining transposed motifs in music. *Journal of Intelligent Information Systems*, 36(1):99–115, 2011.
- [22] Hanna Järveläinen. Algorithmic musical composition. April 2000.
- [23] Sunjung Kim and Elisabeth André. E.: Composing affective music with a generate and sense approach. In *In: Proceedings of Flairs 2004 - Special Track on AI and Music, AAAI Press*, 2004.
- [24] Olivier Lartillot, Mondher Ayari, and Ircam Centre Pompidou. Motivic pattern extraction in music and application to the study of tunisian modal music. 2005.

- [25] Anna Lubiw. Pattern matching in polyphonic music as a weighted geometric translation problem. In *In Proc. 5th International Conference on Music Information Retrieval*, pages 289–296, 2004.
- [26] Benoit Meudic. A causal algorithm for beat-tracking. 2002.
- [27] Benoit Meudic and Emmanuel St-James. Automatic extraction of approximate repetitions in polyphonic midi files based on perceptive criteria. In UffeKock Wiil, editor, *Computer Music Modeling and Retrieval*, volume 2771 of *Lecture Notes in Computer Science*, pages 124–142. Springer Berlin Heidelberg, 2004.
- [28] Michael Müller, Carsten Hausdorf, and Josef Schneeberger. Zur interessantheit bei der entdeckung von wissen in datenbanken. In Gholamreza Nakhaeizadeh, editor, *Data Mining*, volume 27 of *Beiträge zur Wirtschaftsinformatik*, pages 248–264. Physica-Verlag HD, 1998.
- [29] Oriol Nieto. Unsupervised music motifs extraction. Dezember 2010.
- [30] Masayuki Numao, Shoichi Takagi, and Keisuke Nakamura. Constructive adaptive user interfaces: Composing music based on human feelings. In *Eighteenth National Conference on Artificial Intelligence*, pages 193–198, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence.
- [31] Tomasz Michal Oliwa and Markus Wagner 0007. Composing music with neural networks and probabilistic finite-state machines. In *EvoWorkshops*, volume 4974, pages 503–508. Springer, 2008.
- [32] G. Ozcan, C. Isikhan, and A. Alpkocak. Melody extraction on midi music files. In *Multimedia, Seventh IEEE International Symposium on*, pages 8 pp.–, 2005.
- [33] Phillip B. Kirlin Paul E. Utgoff. Detecting motives and recurring patterns in polyphonicmusic. In *Proceedings of the International Computer Music Conference*, pages 487–494, 2006.
- [34] Antonio Pertusa and José M. Iñesta. Pattern recognition algorithms for polyphonic music. October 2004.
- [35] N. Scaringella, G. Zoia, and D. Mlynek. Automatic genre classification of music content: a survey. *Signal Processing Magazine, IEEE*, 23(2):133–141, 2006.
- [36] David Temperley. What’s key for key? The Krumhansl-Schmuckler Key-Finding Algorithm Reconsidered. In *Music Perception*, volume 17, pages 65–100, Fall 1999.
- [37] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, Jul 2002.

- [38] Ryosuke Yamanishi, Keisuke Akita, and Shohei Kato. Automated composing system for sub-melody using hmm: A support system for composing music. In Hyun Seung Yang, Rainer Malaka, Junichi Hoshino, and Jung Hyun Han, editors, *Entertainment Computing - ICEC 2010*, volume 6243 of *Lecture Notes in Computer Science*, pages 425–427. Springer Berlin Heidelberg, 2010.