Effiziente Methoden zur hochauflösenden Musiksynchronisation

Diplomarbeit

Sebastian Ewert

Rheinische Friedrich-Wilhelms-Universität Bonn
Institut für Informatik III

19.12.2007

Inhaltsverzeichnis

Kapitel 1

Einleitung

Ausgangspunkt der vorliegenden Arbeit sind große, digitale Musikbibliotheken, die Musikdaten in verschiedenen Ausprägungen und Formaten enthalten. So kann Musik als Audio-CD-Aufnahme vorliegen, aber auch in Form von digitalisierten Notenblättern, rechnerlesbaren symbolischen Formaten wie MusicXML oder Dateien nach dem MIDI-Standard. Das Gebiet des Music Information Retrieval (MIR) hat sich das Ziel gesetzt, Werkzeuge zu entwickeln, die einem Benutzer einer solchen Bibliothek erlauben, auf neuartige und effiziente Art in diesen Daten zu suchen, zu navigieren, zu browsen oder zu editieren. Mit einem Werkzeug wie der inhaltsbasierten Suche kann der Benutzer eine Suchanfrage nach einer bestimmten Notenfolge stellen und erhält als Ergebnis Suchtreffer zu Daten im Audio-CD-Format. Andere Techniken vereinfachen die Orientierung und Navigation innerhalb eines Musikstücks, indem beim Abspielen zusätzliche Informationen begleitend präsentiert werden, zum Beispiel in Form von Partitur- oder Liedtextdaten. Die musikwissenschaftliche Analyse verschiedener Interpretationen eines Stücks kann unterstützt werden, indem automatisch sich entsprechende Abschnitte identifiziert werden. Damit solche Anwendungen möglich werden, müssen Audio-CD-Aufnahmen sinnvoll mit Metainformationen wie Partiturdaten verbunden werden. Bisherige Verfahren zur automatischen Annotation eines Musikstücks liefern jedoch nur eingeschränkt befriedigende Ergebnisse. Liegen jedoch bereits Noteninformationen zu einem Stück vor, so können diese mittels Synchronisationstechniken zur automatischen Annotation von Audio-CD-Aufnahmen herangezogen werden. Hierbei wird unter *Synchronisation* ein Verfahren verstanden, das zu einer bestimmten Position innerhalb einer Variante eines Musikstücks die entsprechende Stelle innerhalb einer anderen Variante bestimmen kann.[1]

Für eine automatische Synchronisationsmethode stellen sich viele Herausforderungen. So können sich die Varianten aufgrund künstlerischer Freiheiten der Interpreten unterscheiden, aber auch aufgrund von Abweichungen von der zugrunde liegenden Partitur oder anhand ihrer Darstellungsformen (zum Beispiel Audio-CD-Aufnahmen oder MIDI-Daten). In [MMK06] wurde ein effizienter und robuster Ansatz zur Synchronisation harmoniebasierter Musik vorgestellt, der sich in vielen Szenarien als zuverlässig erwiesen hat. Für einige Anwendungsfälle reicht die zeitliche Genauigkeit dieser Methode jedoch nicht aus, weshalb in dieser Arbeit einige Strategien und Methoden zur Erhöhung der zeitlichen Genauigkeit vorgestellt werden.

[1] Die Definition einer Synchronisation wurde in angepasster Form aus [Ari02] entnommen.

Beitrag und Gliederung dieser Arbeit

Ansatzpunkt der vorliegenden Diplomarbeit ist das in [MMK06] beschriebene Verfahren zur Musiksynchronisation. Grundlage dieses Verfahrens bildet Dynamic Time Warping (DTW), eine bewährte Technik, um zwei endliche Zeitreihen aneinander auszurichten. Diese Technik wurde in [MMK06] für das Musiksynchronisationsszenario adaptiert, wobei so genannte Chroma-Merkmale eingesetzt wurden, welche den Harmonieverlauf eines Musikstücks grob kodieren. Durch einen Multiskalenansatz konnte dabei eine hohe Laufzeit- und Speichereffizienz des Verfahrens erreicht werden.

Die Musiksynchronisation stellt jedoch ein aktuelles Forschungsgebiet mit noch zahlreichen offenen Fragestellungen dar. So reicht in manchen Anwendungsfällen die mit dem oben erwähnten Verfahren erzielbare Zeitgenauigkeit nicht aus. Da das Verfahren jedoch als zuverlässige Lösung zur Synchronisation harmoniebasierter Musik gilt, soll es in der vorliegenden Arbeit als Grundlage für Erweiterungen dienen. Dazu wird das Verfahren zunächst selbst erweitert und im Anschluss um komplementäre Strategien ergänzt, die durch Nachverarbeitung die zeitliche Auflösung der Synchronisation erhöhen. Durch Adaption des Multiskalenansatzes erreichen auch die erweiterten Verfahren eine hohe Laufzeit- und Speichereffizienz.

In **Kapitel 2** wird beschrieben, was Merkmale sind und wozu sie eingesetzt werden. Vor der Beschreibung der einzelnen Merkmale werden einige grundlegende Konzepte zur Darstellung von Musik und der Signalverarbeitung umrissen.

Kapitel 3 beginnt mit einer kurzen Beschreibung von Dynamic Time Warping. Darauf aufbauend wird in Form von Multiskalen-DTW ein allgemeiner Ansatz zur effizienten Umsetzung von DTW eingeführt. Das Kapitel endet mit einer Beschreibung des MsDTW-Verfahrens, bei dem Merkmale in Verbindung mit DTW zur effizienten Synchronisation von Musikstücken verwendet werden.

Kapitel 4 beschäftigt sich mit einer Erweiterung der Methoden aus Kapitel 3. Über eine Kombination von Merkmalen werden dabei Informationen über Noteneinsatzzeiten in bestehende Methoden integriert. Im Anschluss werden Methoden vorgestellt, die eine effiziente Berechnung der zuvor entwickelten Erweiterungen ermöglichen.

Kapitel 5 stellt Strategien zur Erhöhung der zeitlichen Auflösung eines Synchronisationsergebnisses vor, das mit Hilfe der erweiterten Methoden aus Kapitel 4 berechnet wurde. Begonnen wird mit der Beschreibung einer Methode zur zeitlichen Interpolation eines Synchronisationsergebnisses. Im Anschluss wird beschrieben, wie ein Verfahren mit hoher zeitlicher Genauigkeit zur Nachverarbeitung von Synchronisationsergebnissen eingesetzt werden kann. Abgeschlossen wird das Kapitel mit einer Abhandlung über weitere Möglichkeiten zur Qualitätserhöhung von Synchronisationsergebnissen.

In **Kapitel 6** werden die beschriebenen Methoden auf größeren Testdatenbeständen evaluiert. An einige statistisch bzw. quantitativ ausgewertete Testreihen schließen sich qualitative Einzeluntersuchungen ausgesuchter Stücke an.

Mit **Kapitel 7** wird die Arbeit mit einer Zusammenfassung und einem Ausblick abgeschlossen.

Danksagung

An dieser Stelle bedanke ich mich herzlich bei Allen, die mir bei der Entstehung der vorliegenden Arbeit hilfreich zur Seite gestanden haben. Zunächst gilt mein Dank der gesamten Arbeitsgruppe von Prof. Dr. M. Clausen, in der diese Diplomarbeit entstanden ist. Ein ganz besonderer Dank gebührt dabei meinem Betreuer Dr. Meinard Müller für ein ungewöhnlich hohes Maß an Engagement. Er stand stets für Diskussionen zur Verfügung und hat mit hilfreichen Anregungen und konstruktiver Kritik zur Entstehung dieser Arbeit beigetragen. Außerdem möchte ich mich bei meinen Freunden und insbesondere Daniel Wolff bedanken, die mir eine große Stütze waren. Ein ganz besonderer Dank gilt Anna Bezgubenko, die mich in einem schwierigen Jahr in vieler Hinsicht menschlich bereichert hat.

Kapitel 2

Merkmalsextraktion

Der Begriffs des Merkmals ist im Bereich Music Information Retrieval (MIR) essenziell. Um die Verwendung von Merkmalen zu motivieren, wird für den Moment postuliert, dass man Musik in Form einer Funktion darstellen kann. Später wird dies in Form der Wellenformdarstellung von Musik genauer beschrieben. Weiterhin stelle man sich vor, dass die Funktionen aus Abbildung 2.1(a) und (b) Musik repräsentieren und die Aufgabe gestellt wird, die beiden auf Ähnlichkeit zu prüfen. Vergleicht man die Funktionen nur punktweise anhand ihrer Werte, so wären sie aufgrund ihrer teils starken punktuellen Abweichung wohl nicht ähnlich zu nennen. Die punktweise Differenz der beiden Funktionen wird in Abbildung 2.1(c) dargestellt. Der durchaus ähnliche Kurvenverlauf der beiden Funktionen kann durch einen punktweisen Vergleich der Funktionen nicht erkannt werden.

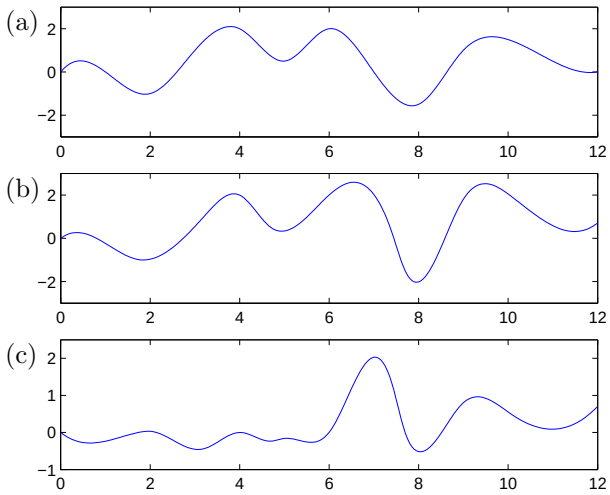


Abbildung 2.1: Zwei Signale ((a) und (b)) und ihre punktweise Differenz (c).

Wird Musik anhand solcher Funktionen dargestellt, kann Ähnlichkeit auf viele verschiedene Arten definiert werden, wobei die meisten dieser Ähnlichkeitsbegriffe nicht sinnvoll über einen punktweisen Vergleich der Funktionen ausgedrückt werden können, wie am Beispiel oben anhand des Ähnlichkeitsbegriffs „ähnlicher Kurvenverlauf“ verdeutlicht wurde. Dies wird erst möglich, wenn man den Begriff der Ähnlichkeit bezüglich anderer Eigenschaften einer solchen Musikfunktion definiert. Solche besonderen oder beschreibenden Eigenschaften eines Signals bezeichnet man auch als *Merkmal*. Erst auf Grundlage passender Merkmale können Audio-

CD-Aufnahmen miteinander verglichen werden oder bezüglich ihres semantischen Gehalts eingestuft werden. Für eine genauere Beschreibung, was Merkmale sind und wie sie berechnet werden, sind jedoch einige Grundlagen nötig, die im Folgenden eingeführt werden.

2.1 Grundlegende Darstellungsformen von Musik

Musik lässt sich auf viele verschiedene Weisen darstellen. So kann Musik mit einer Audio-CD anders präsentiert werden als mit einem Notenblatt. Die Formate unterscheiden sich in semantischem Gehalt, aber auch in der Komplexität, bestimmte Eigenschaften der dargestellten Musik aus diesen Formaten zu erkennen. So kann von einem Notenblatt abgelesen werden, in welchen relativen Zeitabständen welche Note gespielt wird. Aus einer Audio-CD-Aufnahme ist dies nicht direkt erkennbar. In diesem Abschnitt werden die benötigten Darstellungsformen von Musik bzw. deren Formate kurz vorgestellt.

2.1.1 Wellenformdarstellung

Die Wellenformdarstellung ist eng verknüpft mit den physikalischen Ursachen von Schall und Grundlage jeder Audio-CD. Schall wird von vibrierenden Objekten erzeugt, die ihre Schwingung an ein kompressibles Trägermedium weitergeben, welches im Rahmen dieser Arbeit stets Luft ist. Dabei wird das Trägermedium durch die Objektschwingung gestaucht und gedehnt, was lokale Druckveränderungen zur Folge hat. Trägt man den Luftdruck in der Nähe des schwingenden Objekts in einem Zeit/Luftdruck-Graph auf, erhält man die *Wellenformdarstellung*. Der normale Luftdruck (oder Umgebungsdruck) tritt dabei als Referenzdruck in Form der Nulllinie auf, zu dem alle Werte relativ sind. Abbildung 2.2 zeigt ein einfaches Beispiel einer Wellenform, die von einem gleichmäßig schwingenden Objekt erzeugt wurde.

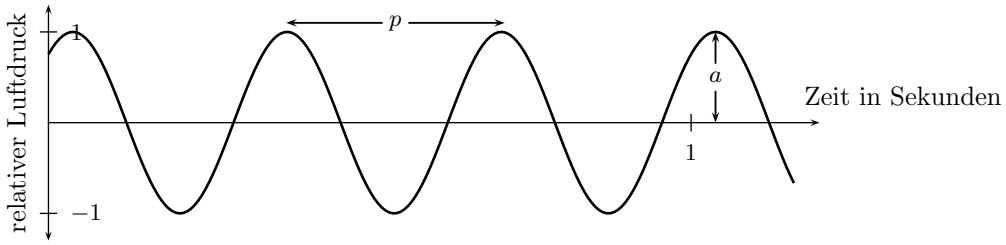


Abbildung 2.2: Wellenformdarstellung von Schall, erzeugt von einer gleichmäßigen Schwingung (Frequenz 3 Hz). Zusätzlich eingezeichnet ist die Amplitude a und die Periodendauer p (angepasst aus [Mül07]).

Wenn ein Objekt derart schwingt, dass sich die Werte der zugehörigen Wellenformdarstellung in regelmäßigen Abständen wiederholen, nennt man diese Schwingung *periodisch*. In diesem Fall kann man wie in Abbildung 2.2 die *Periode* p über die Länge dieser Abstände, sowie die *Frequenz* f mit $f = 1/p$ als den reziproken Wert der Periode definieren. Ist die Dauer einer Periode in Sekunden angegeben, so ist die Einheit der Frequenz 1 / Sekunde oder 1 *Hertz* ($1Hz = 1/s$). Eine weitere wichtige Kenngröße ist die *Amplitude*, welche die maximale Auslenkung innerhalb einer Periode bezeichnet. Für eine genauere Beschreibung dieser

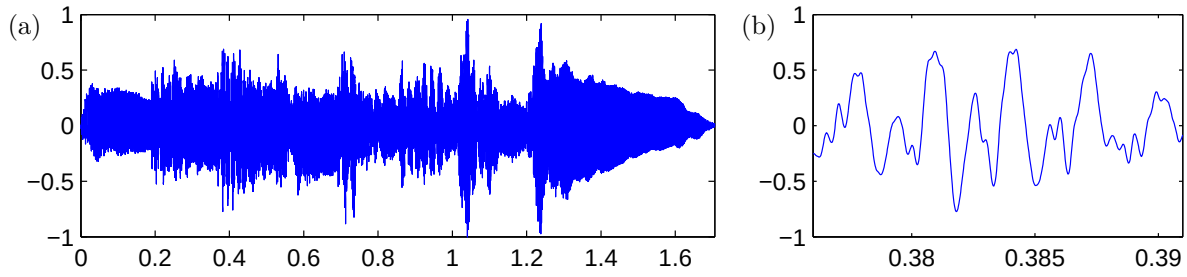


Abbildung 2.3: (a) Wellenformdarstellung einer auf einem Klavier gespielten C-Dur Tonleiter. (b) Vergrößerung eines Ausschnitts.

Begriffe siehe auch [CM01]. Abbildung 2.3 zeigt als Beispiel die Wellenformdarstellung einer Tonleiter.

Die Wellenform ist mathematisch eine kontinuierliche Funktion. In der Praxis ist es jedoch im Allgemeinen nicht möglich, kontinuierliche Funktionen im Rechner darzustellen. Aus diesem Grund diskretisiert man eine Wellenform. Dazu wird zunächst der Wert der Wellenform an äquidistant verteilten Stellen bestimmt bzw. *abgetastet*. Im Anschluss wird der Wert über eine endliche Menge von Werten approximiert bzw. *quantisiert*. Diese diskrete Darstellungsform nennt man auch *Pulse Code Modulation (PCM)*. Abbildung 2.4 zeigt ein Beispiel.

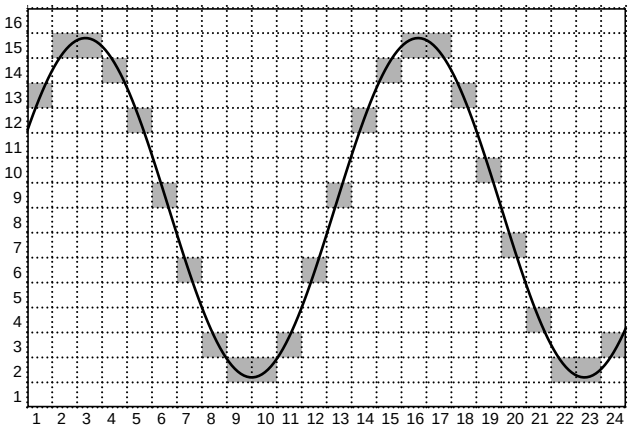


Abbildung 2.4: Kontinuierliche Wellenform (schwarze Kurve) wird diskretisiert (graue Flächen). Es werden 24 Abtastpunkte und $2^4 = 16$ Quantisierungswerte verwendet (aus [Mül07]).

Im Audio-CD-Standard wird eine Abtastung mit 44100 Abtastwerten pro Sekunde verwendet (*Abtastfrequenz* 44100 Hz), wobei für die Quantisierung ein Wertebereich bestehend aus 2^{16} verschiedenen Werten verwendet wird.

In dieser Arbeit wird der Begriff *Audio* synonym zur Wellenformdarstellung verwendet. Eine „Audioaufnahme“ entspricht somit einer Aufnahme von Musik in Wellenformdarstellung.

2.1.2 Symbolische Darstellung und das MIDI-Format

Das Konzept der Wellenformdarstellung ist stark von den physikalischen Ursachen von Schall geprägt. Im Unterschied dazu wird Musik in der Musiktheorie in Form von Noten dargestellt. Die dabei verwendete Notenschrift spezifiziert bestimmte Anweisungen, wie ein Instrument zu spielen ist. Zentrales Element dieser Notenschrift ist die Angabe bestimmter Tonhöhen, die zu relativ angegebenen Zeitpunkten wiedergegeben werden sollen. Für Details siehe zum Beispiel [Hem97].

Bei der Definition dieser Tonhöhen wurde auf bestimmte Aspekte des menschlichen Hörapparats Rücksicht genommen. So hat man festgestellt, dass der Mensch Frequenzen nicht linear wahrnimmt, sondern in etwa logarithmisch. Es zeigt sich, dass eine Frequenz von 880 Hz nicht vier, sondern nur drei mal so hoch wie 220 Hz empfunden wird. Das Intervall zwischen zwei als doppelt so hoch empfundenen Frequenzen wird dabei als *Oktave* bezeichnet und ist Grundlage des Tonhöhensystems, das sich in der gesamten westlichen Musikwelt durchgesetzt hat.

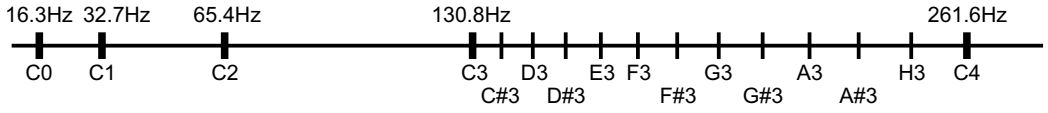


Abbildung 2.5: Oktav-basiertes Tonmodell.

Dabei wird ausgehend von einer Grundfrequenz von etwa 16 Hz das Hörspektrum in Oktaven unterteilt. Jede Oktave wird nochmals in zwölf so genannte *Halbtöne* unterteilt. Die Bezeichner der Halbtöne werden zusammengesetzt. Der erste Teil kennzeichnet, welcher Halbton innerhalb einer Oktave gemeint ist: C, C# oder Cis, D, D# oder Dis, E, F, F# oder Fis, G, G# oder Gis, A, A# oder Ais, H. Zusammen mit einer natürlichen Zahl, die die Oktave angibt, kann so aus den Bezeichnern direkt abgelesen werden, welche Tonhöhenunterschiede vorliegen[1]. So wird ein $A3$ dreimal so hoch wahrgenommen wie ein $A1$. Die Beschränkung auf zwölf Halbtöne ist zunächst willkürlich, findet jedoch ihre Berechtigung in Beziehungen der auftretenden *Obertöne* untereinander. Obertöne werden für gewöhnlich von jedem Instrument erzeugt. Beim Anspielen schwingt ein Instrument nicht gleichmäßig mit nur einer einzigen Frequenz. Stattdessen wird eine Grundfrequenz erzeugt, und ausgehend von dieser weitere Frequenzen, die ganzen Vielfachen dieser Grundfrequenz entsprechen (*Harmonische* oder *Naturtonreihe*). Diese weiteren Frequenzen heißen Obertöne. Der menschliche Hörsinn ist in der Lage, aus diesem Frequenzgemisch die Grundfrequenz zu erkennen und nimmt sie als Tonhöhe wahr. Die Unterteilung in zwölf Halbtonschritte kann nun darüber erklärt werden, dass gleichzeitig gespielte Tonhöhen in einem ästhetischen Sinn in vielen Kombination gut zusammenklingen. Weitere Informationen zum Zusammenspiel von Obertönen und die Auswirkung auf Instrumente und Musiktheorie finden sich z.B. in [Set05] oder [Bla98].

Auf der Suche nach einem Standard, mit dem die Parameter dieses Tonsystems in einem rechnerlesbaren Format gespeichert werden können, stößt man schnell auf das *MIDI-Format*

[1] Hier wird die ISO-Bezeichnung der Halbtöne beschrieben. In der Musiktheorie gab es historisch weitere Bezeichner und auch Zwischentöne. In der heute vorherrschenden, so genannten wohltemperierten Stimmung kann aufgrund der enharmonischen Verwechslung jedoch auf diese zusätzlichen Bezeichner verzichtet werden. Siehe dazu [Hem97].

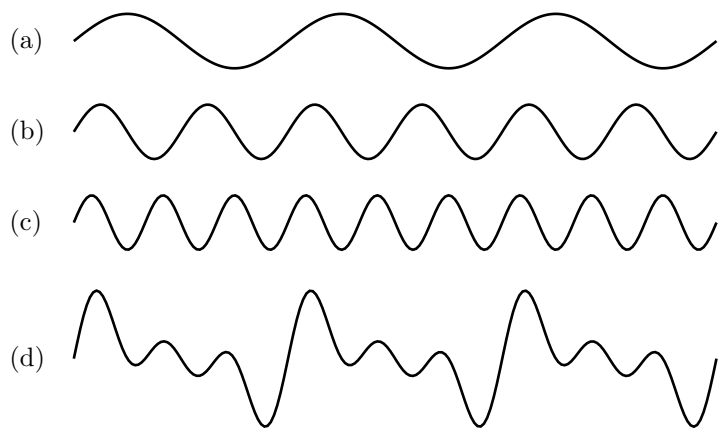


Abbildung 2.6: Oberton Beispiel: Ausgehend von einer Grundfrequenz von 3 Hz (a) werden die ersten beiden Obertöne 6 Hz (b) und 9 Hz (c) gezeigt. (d) zeigt die Überlagerung von (a) - (c).

(*Musical Instrument Digital Interface*). Im MIDI-Format wird Musik in Form so genannter *MIDI-Ereignisse* repräsentiert. Die wichtigsten sind die *NOTE ON* und *NOTE OFF* Ereignisse, welche kennzeichnen, dass ein bestimmter Ton zu einem gewissen Zeitpunkt angespielt und bis zu einem zweiten Zeitpunkt gehalten werden soll. Andere Ereignisse regeln die Intensität (*Velocity*), mit der ein Ton gespielt werden soll, oder steuern technische Details des Standards. MIDI-Ereignisse werden zur Klangerzeugung an MIDI-Instrumente, wie Keyboards, Synthesizer oder Sampler übertragen und dort in hörbare Klänge umgesetzt.

Im MIDI-Format werden Tonhöhen in Form natürlicher Zahlen angegeben. Dazu werden alle Halbtöne mit aufsteigender Frequenz durchnummeriert. Mit folgender Formel kann aus einer Tonhöhe in MIDI-Form auf die Grundfrequenz geschlossen werden, die einer Tonhöhe p zugeordnet ist (eine genauere Tabelle ist im Anhang B zu finden):

$$f(p) = 2^{\frac{p-69}{12}} \cdot 440$$

Besonders wichtig sind die MIDI-Tonhöhen 21 bis 108, da diese genau der Belegung einer modernen Klaviatur entsprechen. Die später besprochenen Merkmale werden sich deshalb auf diesen Tonhöhenbereich beschränken.

Verglichen mit einer Darstellung über ein Notenblatt kodiert das MIDI-Format viele Parameter genauer und lässt deutlich weniger Raum für Interpretationen. Einige semantische Inhalte, die ein Notenblatt darstellen kann, werden im MIDI-Format jedoch nur indirekt gespeichert und sind deshalb oftmals verloren. So würde ein „ritardando“ im Notenblatt lediglich durch eine langsamer werdende Abfolge der Noten im MIDI-Format realisiert oder ein „forte“ würde zu einer Erhöhung der Anspielstärke jeder Note führen. Die Anweisungen selbst können aber nicht gespeichert werden.

MIDI-Daten werden häufig mit Hilfe einer *Piano-Roll-Darstellung* visualisiert. Balken kodieren dabei in vertikaler Richtung die Tonhöhe und in horizontaler die Zeit, in der eine Note angespielt und gehalten wird. Anspielstärke und weitere MIDI-Informationen werden dabei

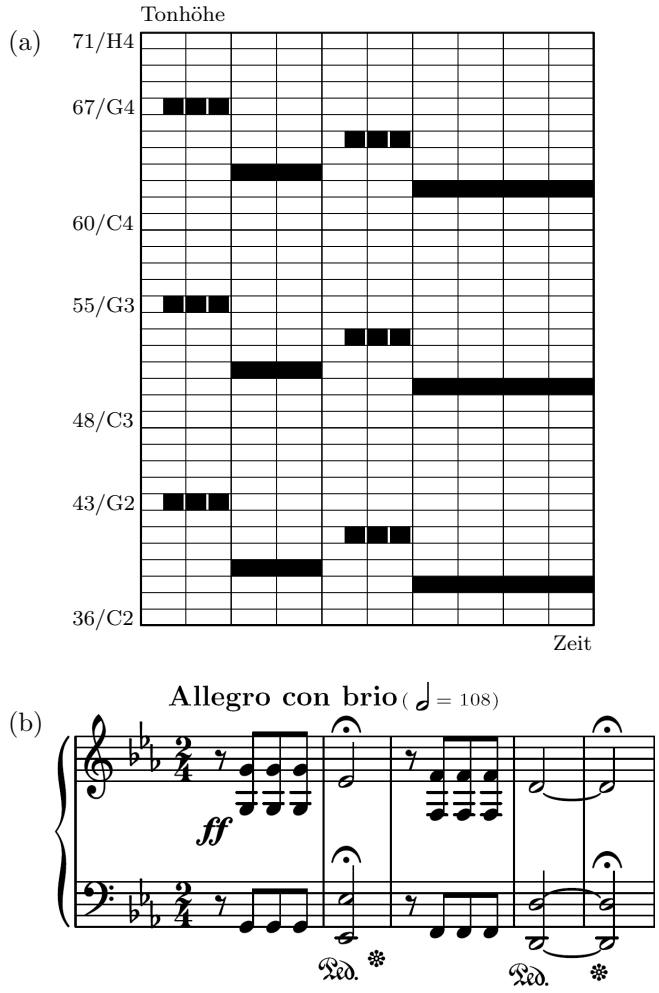


Abbildung 2.7: Beginn von Beethovens fünfter Symphonie (Opus 67) in c-Moll, dargestellt in (a) Piano-Roll-Darstellung (b) Notenschrift (aus [Mül07]).

nicht dargestellt. Der Name soll an gelochte Papierbänder erinnern, mit denen in der Vergangenheit selbstspielende Klaviere gesteuert wurden. Abbildung 2.7 zeigt ein Beispiel einer Piano-Roll-Darstellung und stellt diese einer klassischen Notenschrift gegenüber.

2.2 Grundlagen der Signalverarbeitung

In Abschnitt 2.3 werden einige grundlegende Begriffe und Methoden der Signalverarbeitung benötigt. Auch wenn an dieser Stelle auf eine vollständige Einführung verzichtet werden muss, sollen zumindest Notation und exakte Definition dieser Begriffe festgelegt werden. Für eine detaillierte Einführung wird auf [CM01] verwiesen.

Definition 2.1 (Signal). *Ein Signal ist ein Element des Lebesgueraums $\ell^2(\mathbb{Z})$, wobei:*

$$\ell^2(\mathbb{Z}) = \left\{ x : \mathbb{Z} \to \mathbb{C} \;\middle|\; \|x\|_2 := \left(\sum_{n \in \mathbb{Z}} |x(n)|^2 \right)^{\frac{1}{2}} < \infty \right\}$$

Man bezeichnet $\ell^2(\mathbb{Z})$ auch als *Signalraum*. Für eine genaue Betrachtung der Eigenschaften von $\ell^2(\mathbb{Z})$ siehe z.B. [Fol84].

Hinweis: Musik in Wellenformdarstellung (Abschnitt 2.1) kann als ein Signal mit endlichem Träger aufgefasst und in den $\ell^2(\mathbb{Z})$ eingebettet werden. Man erkennt leicht, dass die so entstehende Funktion aufgrund ihres endlichen Trägers die Definition einer $\ell^2(\mathbb{Z})$ Funktion erfüllt.

Auf Signalen kann eine Ableitung definiert werden.

Definition 2.2 (Differentiation von Signalen). *Sei $x \in \ell^2(\mathbb{Z})$. Dann ist die Ableitung x' von x definiert über:*

$$x'(n) := x(n) - x(n-1)$$

Eine der wichtigsten Abbildungen auf dem Signalraum $\ell^2(\mathbb{Z})$ ist die Fouriertransformation.

Definition 2.3 (Fouriertransformation). *Die Fouriertransformierte $\hat{x}$ eines Signals x ist definiert über:*

$$\begin{aligned} \hat{x} &: [0,1] \to \mathbb{C} \\ \hat{x}(\omega) &:= \sum_{n\in\mathbb{Z}} x(n) e^{-2\pi i \omega n} \end{aligned}$$

Auch wenn die Fouriertransformation sehr vielfältig verstanden und eingesetzt werden kann, wird sie in dieser Arbeit ausschließlich als *Frequenzdarstellung eines Signals* interpretiert. Wichtig ist vor allem der Betrag der Fouriertransformierten $|\hat{x}(\omega)|$, welcher als Intensität angesehen werden kann, mit der die Frequenz ω im Signal x auftritt.

Obwohl das Intervall $[0,1]$ kontinuierlich ist, kann die Fouriertransformation in der Praxis nur für ausgewählte Frequenzen ω berechnet werden. Zudem treten praktisch nur Signale mit endlichem Träger auf. Legt man sich auf eine äquidistant in $[0,1]$ verteilte Menge von zu berechnenden Frequenzen fest, kann man diese Vereinfachungen zur Definition einer praktischen Berechnungsmöglichkeit mit einbeziehen und erhält die DFT.

Definition 2.4 (Diskrete Fouriertransformation). *Die Diskrete Fouriertransformation oder* DFT *der Länge N ist eine unitäre Abbildung auf $\mathbb{C}^N$, welche mit $\Omega_N := e^{-\frac{2\pi i}{N}}$ durch folgende Matrix gegeben ist:*

$$DFT_N := \left(\Omega_N^{kj}\right)_{0\le k,j<N} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & \Omega_N^1 & \dots & \Omega_N^{(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \Omega_N^{(N-1)} & \dots & \Omega_N^{(N-1)(N-1)} \end{pmatrix}$$

Bezeichnet $x \in \mathbb{C}^N$ ein Signal mit einem endlichem Träger, das in den $\mathbb{C}^N$ eingebettet wurde, so erhält man die DFT-Approximation der Fouriertransformierten von x über eine Matrix-Vektor-Multiplikation: $DFT_N x$.

Häufig interessiert aber nicht nur die reine Frequenzdarstellung eines Signals, sondern eine Mischform von Zeit- und Frequenzinformationen. Eine Möglichkeit dazu bietet die gefensterte Fouriertransformation.

Definition 2.5 (gefensterte Fouriertransformation). *Sei $g \in \ell^2(\mathbb{Z})$ mit $\|g\|_2 \neq 0$. Dann heißt g Fensterfunktion und für ein $x \in \ell^2(\mathbb{Z})$ heißt*

$$\begin{aligned} \tilde{x} &: [0,1] \times \mathbb{Z} \to \mathbb{C} \\ \tilde{x}(\omega, n) &:= \sum_{\ell \in \mathbb{Z}} \bar{g}(\ell) x(\ell + n) e^{-2\pi i \omega \ell} \end{aligned}$$

die (g-)gefensterte Fouriertransformierte von x.

Zu den wichtigsten Fensterfunktionen gehören das Rechteck- und das Hann-Fenster.

Definition 2.6 (Rechteck- und Hann Fenster). *Das Rechteckfenster der Länge N ist definiert über:*

$$g(n) := \begin{cases} 1 & \text{falls } n \in \{1, \ldots, N\} \\ 0 & \text{sonst} \end{cases}$$

Das Hann-Fenster der Länge N ist definiert über:

$$g(n) := \begin{cases} 0.5 \cdot (1 - \cos(2\pi \frac{n}{N+1})) & \text{falls } n \in \{1, \ldots, N\} \\ 0 & \text{sonst} \end{cases}$$

Indem man sich auf eine endliche Anzahl von zu berechnenden Frequenzen beschränkt, kann analog zur DFT eine diskrete Form der gefensterten Fouriertransformation definiert werden.

Definition 2.7 (gefensterte diskrete Fouriertransformation). *Sei $g \in \mathbb{C}^N$ mit $g \neq 0$, $S \in \mathbb{N}$ und $x \in \ell^2(\mathbb{Z})$. Dann heißt*

$$\begin{aligned} \tilde{x} &: [0, \ldots, N-1] \times \mathbb{Z} \to \mathbb{C} \\ \tilde{x}(k, m) &:= \sum_{\ell=0}^{N-1} \bar{g}(\ell) x(\ell + m \cdot S) \Omega_N^{\ell k} \end{aligned}$$

(g-) gefensterte diskrete Fouriertransformierte von x mit Schrittweite S.

Definition 2.8 (Spektrogramm). *Es sei $x \in \ell^2(\mathbb{Z})$. Ferner bezeichne $\tilde{x}$ die (g-) gefensterte diskrete Fouriertransformierte von x mit Schrittweite S für ein $g \in \mathbb{C}^N$, $g \neq 0$ und $S \in \mathbb{N}$. Außerdem sei:*

$$|\tilde{X}_m| := \begin{pmatrix} |\tilde{x}(0, m)| \\ |\tilde{x}(1, m)| \\ \vdots \\ |\tilde{x}(N-1, m)| \end{pmatrix}$$

Dann heißt die Folge $\left(|\tilde{X}_m|\right)_{m \in \mathbb{Z}}$ Spektrogramm.

Ein *System* bezeichnet eine beliebige Abbildung zwischen Signalräumen. Wichtige Systeme sind der M-Dezimierer und der M-Expandierer.

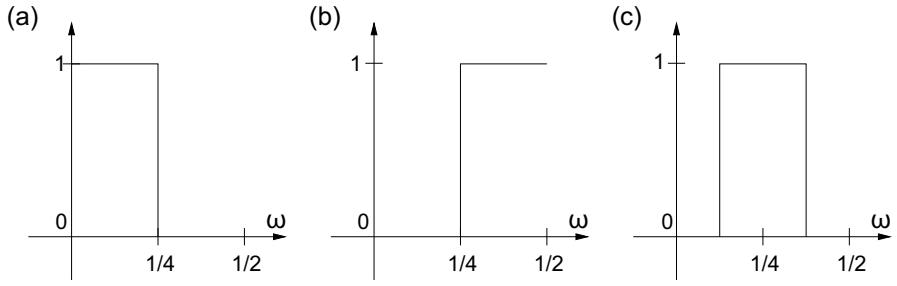


Abbildung 2.8: Stilisierte Filter. Horizontal ist die Frequenz aufgetragen, vertikal der zugehörige Dämpfungsfaktor: (a) Tiefpass-, (b) Hochpass- und (c) Bandpassfilter (adaptiert von [Mül07]).

Definition 2.9 (M-Dezimierer und M-Expandierer). *Sei $x \in \ell^2(\mathbb{Z})$. Dann ist der M-Dezimierer (Downsampler) definiert durch:*

$$(\downarrow M)[x](n) := x(M \cdot n)$$

Der M-Expandierer (Upsampler) ist definiert durch:

$$(\uparrow M)[x](n) := \begin{cases} x(n/M) & \text{falls } M \mid n \\ 0 & \text{sonst} \end{cases}$$

Interpretiert man ein Signal analog zur Fouriertransformation als Überlagerung von sinusartigen Schwingungen, so können Systeme zur kontrollierten Verstärkung oder Abschwächung bestimmter im Signal auftretender Frequenzbereiche eingesetzt werden. Zu dieser Art von Systemen gehören *Hoch-*, *Tief-* und *Bandpassfilter*. Mit solchen Systemen (auch *Filter* genannt) wird ein Frequenzbereich assoziiert, der *Durchlassbereich*. Frequenzen aus diesem Bereich erfahren bei Anwendung des Systems auf ein Signal nahezu keine Veränderung. Frequenzen außerhalb des Durchlassbereichs sollen jedoch möglichst stark gedämpft werden. Abbildung 2.8 stellt den Durchlassbereich idealer Hoch-, Tief- und Bandpassfilter stilisiert dar, wobei Frequenzen nicht aus dem Bereich $[0, 1]$ angegeben werden, wie aus der Definition der Fouriertransformation zu vermuten wäre, sondern aus dem Bereich $[0, 1/2]$. Der Grund dafür liegt in einer Symmetrieeigenschaft der Fouriertransformation, auf die hier aber nicht weiter eingegangen werden soll (siehe auch [CM01]).

Oftmals ist es notwendig ein Signal in mehrere Frequenzbereiche aufzuteilen. Eine Menge von Bandpassfiltern, deren Durchlassbereich jeweils einem dieser Frequenzbereiche entspricht, bezeichnet man auch als *Filterbank*.

Mathematisch lässt sich zeigen, dass unter bestimmten zusätzlichen Forderungen nach Linearität, zeitlicher Invarianz und Stetigkeit sich jedes System als eine *Faltung* darstellen lässt.

Definition 2.10 (Faltung). *Die Faltung eines Signals x mit einem Signal y ist definiert über:*

$$(x * y)(n) = \sum_{k \in \mathbb{Z}} x(k)y(n-k)$$

Damit wurden alle grundlegenden Konzepte der Signalverarbeitung eingeführt, die zur Beschreibung der in dieser Arbeit verwendeten Merkmale benötigt werden.

2.3 Verwendete Merkmale

Wie in der Kapiteleinleitung bereits motiviert wurde, dienen Merkmale der Reduktion auf semantisch interessante Eigenschaften von Signalen. Über solche Eigenschaften können verschiedene Arten von Ähnlichkeit zwischen Signalen definiert und untersucht werden. An folgendem Beispiel soll motiviert werden, dass nicht alle Merkmalstypen gleich gut zur Lösung einer Aufgabenstellung geeignet sind. Lokale Extrema eines Signals sind ein einfaches Beispiel für Merkmale (Abbildung 2.9).

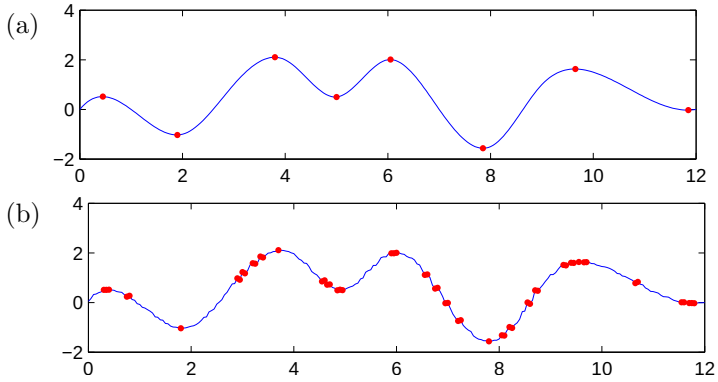


Abbildung 2.9: lokale Minima / Maxima zweier Signale

Es sei die Beispielaufgabe gestellt, anhand lokaler Extrema zu prüfen, ob die beiden Signale aus Abbildung 2.9 ähnlich sind. Unter Ähnlichkeit soll dabei verstanden werden: „identisch bis auf einen Rauschanteil“, womit die beiden Signale ähnlich zu nennen wären, da das zweite Signal durch Hinzufügen eines leichten Rauschanteils aus dem ersten erzeugt worden ist.

Bestimmt man nun die Folge lokaler Extrema der Signale, so stellt man fest, dass durch das leichte Rauschen eine große Anzahl zusätzlicher lokaler Extrema entstehen. Lokale Extrema sind somit nicht invariant bezüglich Rauschen und eignen sich schlecht, um Ähnlichkeit in obiger Form zu untersuchen. Würden rauschinvariante Merkmalstypen verwendet, so erhielte man für beide Signale identische Merkmale und ein Vergleich wäre einfach.

Verallgemeinert lässt sich sagen, dass die Wahl der Merkmale stets anwendungsabhängig zu treffen ist. Die Qualität und Aussagekraft einer Methode im Bereich MIR hängt häufig eng mit der Wahl der verwendeten Merkmale zusammen. Die im Folgenden vorgestellten Merkmale eignen sich im Speziellen für den Anwendungsfall „Musiksynchronisation“ und werden im Rahmen dieser Arbeit benötigt (es werden die englischen Orginalnamen verwendet):

- STMSP-Merkmale
- Onset-Merkmale
- Chroma/CENS-Merkmale
- Novelty-Merkmale

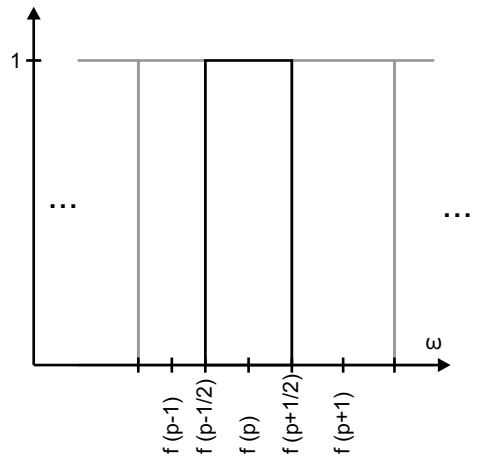


Abbildung 2.10: Stilisierte Darstellung der Bandpassfilter zur Halbton-Subbandzerlegung. Horizontal ist die Frequenz aufgetragen, vertikal der Dämpfungsfaktor der Filter. Jedes Rechteck kennzeichnet den Durchlassbereich eines Filters.

Bis auf Novelty-Merkmale, welche in dieser Reihe eine Sonderrolle einnehmen, versucht man mit all diesen Merkmalen Informationen über die Entwicklung von Tonhöhen aus dem Signal zu extrahieren. Töne werden nach dem Halbtonmodell unterschieden, wie es unter 2.1.2 vorgestellt wurde, wobei die Tonhöhen mit ihrer MIDI-Kennung bezeichnet werden. Dabei unterscheiden sich die Merkmale anhand ihrer Robustheit gegenüber Variationen in Klangfarbe (Timbre), Instrumentation und Dynamik. Für jedes Merkmal werden Standardparameter festgelegt, so dass diese im weiteren Verlauf der Arbeit nicht unnötig wiederholt werden müssen.

2.3.1 Zerlegung in Halbton-Subbandsignale

Mit Ausnahme der Novelty-Merkmale basieren alle verwendeten Merkmale auf einer Zerlegung des Signals in *Halbton-Subbandsignale*. Für jeden Halbton wird dazu ein Bandpassfilter definiert, dessen Durchlassbereich einen geeigneten Frequenzbereich um die Grundfrequenz abdeckt, die mit dem Halbton assoziiert wird. Diese Bandpassfilter werden zu einer Filterbank zusammengefasst. Es sei daran erinnert, wie sich die zentrale Grundfrequenz eines Halbtons p anhand der MIDI-Nummerierung der Halbtöne ergibt:

$$f(p) = 2^{\frac{p-69}{12}} \cdot 440$$

Über diese Formel lassen sich auch Frequenzen angeben, die der Hörsinn zwischen zwei Halbtönen einordnen würde. Diese Zwischenfrequenzen eignen sich damit ideal zur Definition der Grenzen des Durchlassbereichs der Halbton-Bandpassfilter (Abbildung 2.10):

$$f_{tief}(p) := f(p - 1/2) = 2^{\frac{p-69.5}{12}} \cdot 440 \qquad f_{hoch}(p) := f(p + 1/2) = 2^{\frac{p-68.5}{12}} \cdot 440$$

In der Praxis sind Bandpassfilter mit perfektem Dämpfungsverhalten jedoch nicht realisierbar. Bei der Wahl der Grenzen des Durchlassbereichs muss dieser Umstand für praktische Berech-

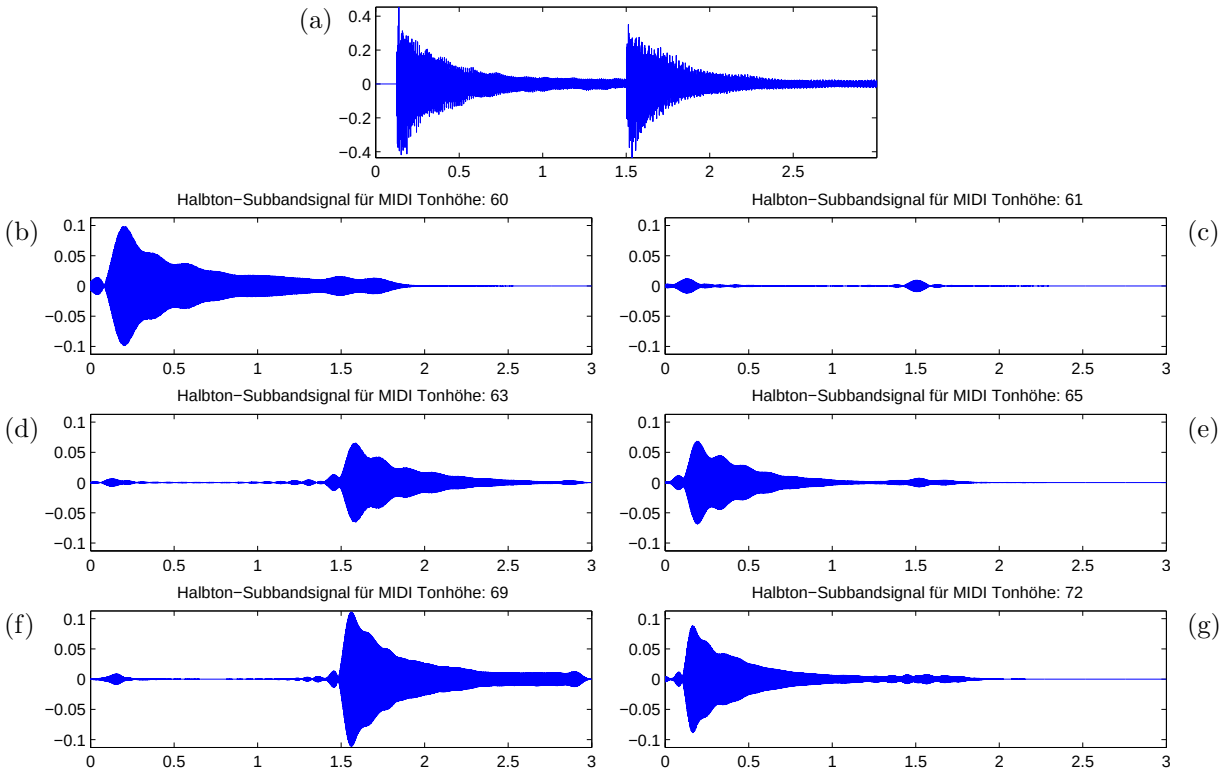


Abbildung 2.11: (a) Wellenform einer Klavieraufnahme zweier Akkorde (C4/F4 und D#4/A4). (b)-(g) Halbton-Subbandsignale zu ausgewählten MIDI-Tonhöhen.

nungen berücksichtigt werden. Für eine idealisierte Vorstellung reicht die oben vorgestellte Wahl der Grenzen jedoch aus. [2]

Abbildung 2.11(a) zeigt die Wellenform einer Klavieraufnahme zweier Akkorde (C4/F4 und D#4/A4). C4 entspricht nach MIDI-Nummerierung der Tonhöhe 60, F4 der 65, D#4 der 63 und A4 der 69. Die Halbton-Subbandsignale zu diesen vier Tonhöhen werden in (b), (e), (d) und (f) gezeigt. Zusätzlich dargestellt sind zwei Halbton-Subbandsignale zu Tonhöhen, die nicht mit diesen vier übereinstimmen. In (c) erkennt man die erzielte Dämpfung des Signals für die Tonhöhe 61. Abbildung (g) zeigt jedoch einen deutlichen Ausschlag im Halbton-Subbandsignal, obwohl die zugehörige Tonhöhe 72 nicht angespielt wird. Dieser Ausschlag wird vom ersten Oberton von C4 verursacht, welcher in den Frequenzbereich der Tonhöhe 72 fällt.

Das in Abbildung 2.11(a) gezeigte Signal wird im Folgenden als Beispiel bei der Beschreibung der Merkmale verwendet.

[2] Es seien einige Details über die technische Umsetzung erwähnt, wobei jedoch auf die Einführung aller verwendeten Begriffe verzichtet werden muss. Für eine komplette Beschreibung wird auf [Mül07] verwiesen. Die Filterung wird über eine Multiraten-Filterbank realisiert (drei verschiedene Abtastraten). Sie besteht aus 88 elliptischen Filtern der Ordnung 8 mit einer Dämpfung von 50dB außerhalb des Durchlassbereichs, die in einer Vorwärts-Rückwärts Filterung eingesetzt eine Gruppenlaufzeit nahe 0 aufweisen. Die 88 Filter decken die MIDI-Tonhöhen von 21 bis 108 ab.

2.3.2 STMSP-Merkmale

Die zeitliche Auflösung der Halbton-Subbandsignale ist sehr hoch. Es stellt sich für die weiterführende Signalanalyse als günstig heraus, die Auflösung zu verringern. Formal geschieht diese Reduktion durch Bilden der *STMSP* (*Short-Time Mean-Square Power*) Merkmale. Dazu wird jedes Halbton-Subbandsignal in gleich große Zeitabschnitte unterteilt, die sich auch überlappen können. Für jeden dieser Zeitabschnitte wird eine Energie oder Aktivität berechnet. Genauer ist gegeben $w \in \mathbb{N}$ die STMSP eines Halbton-Subbandsignals x an der Stelle $n \in \mathbb{Z}$ definiert über:

$$\sum_{k \in [n-\lfloor \frac{w}{2} \rfloor : n+\lfloor \frac{w}{2} \rfloor]} |x(k)|^2$$

Dabei gibt $w \in \mathbb{N}$ die Größe des Zeitabschnitts in Abtastwerten an. Zur Datenreduktion wird die STMSP nur alle $d \in \mathbb{N}$ Abtastwerte berechnet. Es sei erwähnt, dass diese Berechnung auch als Faltung des quadrierten Signals mit einem Rechteckfenster und anschließendem Downsampling verstanden werden kann. Unter Einsatz anderer Fensterfunktionen können die Faktoren auch gewichtet werden.

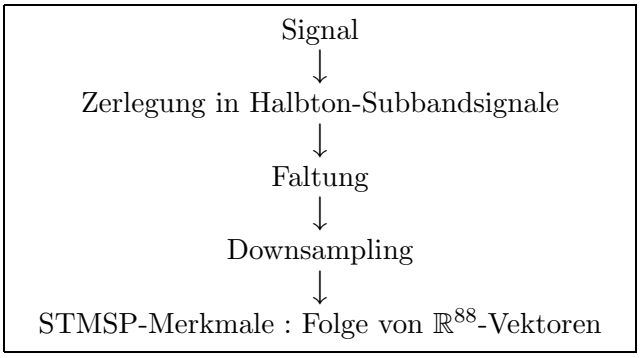


Abbildung 2.12: Berechnung der STMSP-Merkmale

Dies wird für jedes der 88 Halbton-Subbandsignale (MIDI-Tonhöhen 21 bis 108) durchgeführt. Mit festen Werten für w und d entstehen so pro Zeitabschnitt 88 reelle Werte, die zusammen einen Vektor aus $\mathbb{R}^{88}$ bzw. ein STMSP-Merkmal ergeben. Abbildung 2.12 und 2.13 zeigen den Ablauf der Berechnung der STMSP-Merkmale und die STMSP-Merkmale für das Beispielsignal aus Abbildung 2.11(a).

Man legt für die STMSP-Merkmale folgende Standardparameter fest. Die Fensterbreite beträgt standardmäßig 200 ms, was in Abhängigkeit von der Abtastfrequenz in Abtastwerte umzurechnen ist. Die Fensterüberlappung wird auf 100 ms festgelegt. Als Fensterfunktion wird, wie oben vereinfacht dargestellt, das Rechteckfenster verwendet.

2.3.3 STMSP-Merkmale aus MIDI-Daten

Für den Vergleich von MIDI-Daten und Musik in Wellenformdarstellung ist es wichtig, eine Vergleichsbasis für beide Formate durch Verwendung derselben Merkmalstypen zu schaffen. Da alle folgenden Merkmale, mit Ausnahme der Novelty-Merkmale, auf STMSP beru-

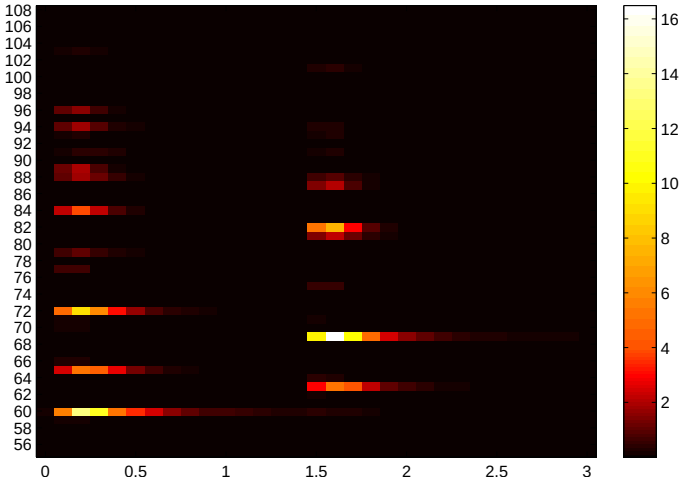


Abbildung 2.13: STMSP-Merkmale unter Standardparametern für das Signal aus Abbildung 2.11(a). Es werden lediglich die Tonhöhen 55 bis 108 gezeigt.

hen, reicht es zu definieren, wie STMSP-Merkmale aus MIDI-Daten erzeugt werden. Da mit STMSP-Merkmalen versucht wird, gewisse Informationen des Tonsystems zu extrahieren, wie es auch im MIDI-Standard zum Einsatz kommt, verläuft die Erzeugung von STMSP-Merkmalen aus MIDI-Daten deutlich direkter.

Mit Kenntnis der Länge des Stücks kann nach Festlegung von Fensterbreite und -überlappung eine Folge von STMSP-Merkmalen erzeugt werden, deren Komponenten auf 0 gesetzt werden. Beim Durchlaufen der durch MIDI-Daten beschriebenen Notenliste erhält man für jede Note die Anspielzeit und Haltedauer sowie die Anspielstärke und Tonhöhe. Über den durch Anspielzeit und Haltedauer definierten Zeitbereich kann eine Menge von STMSP-Merkmalen identifiziert werden, die diesen Zeitbereich beschreiben. Anhand der Anspielstärke und Tonhöhe können dann Werte in die entsprechenden Komponenten der STMSP-Merkmale eingetragen werden. Zusätzlich lassen sich einfache Obertonmodelle umsetzen.

Eine Alternative dazu ist das Erzeugen einer Audioaufnahme aus MIDI-Daten mit Hilfe einer geeigneten MIDI Synthese-Software und anschließender STMSP-Erzeugung wie in Abschnitt 2.3.2. Diese Möglichkeit ist aber deutlich rechenintensiver und hat kaum Auswirkung auf die merkmal-verarbeitenden Methoden gezeigt.

2.3.4 Onset-Merkmale

Aus den Halbton-Subbandsignalen in Abbildung 2.11 kann man erkennen, zu welcher Zeit eine Note angespielt wird. So ist aus Abbildung 2.11(f) (wiederholt in Abbildung 2.14) ersichtlich, dass bei etwa 1.5 Sekunden ein A4 gespielt wird (bzw. eine Note, deren Obertöne in dem Frequenzbereich von A4 liegen). Diese Zeitpunkte sind durch einen Anstieg der Schwingung gekennzeichnet. Mit *Onset-Merkmalen* wird versucht, solche Noteneinsätze automatisch zu erkennen.

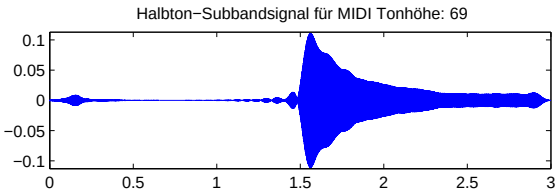


Abbildung 2.14: Wiederholung von Abbildung 2.11(f)

Mathematisch sind solche Veränderungen an der Ableitung einer Funktion erkennbar. Für die Onset-Merkmale wird deshalb jedes Halbton-Subbandsignal differenziert. Da nur Signalanstiege einen Noteneinsatz kennzeichnen, werden negative Werte der Ableitung auf 0 gesetzt, was man analog zu elektrischen Schaltungen auch *Einweggleichrichtung* nennt. Der genaue Ablauf wird ausgehend von STMSP-Merkmalen in Abbildung 2.15 gezeigt.

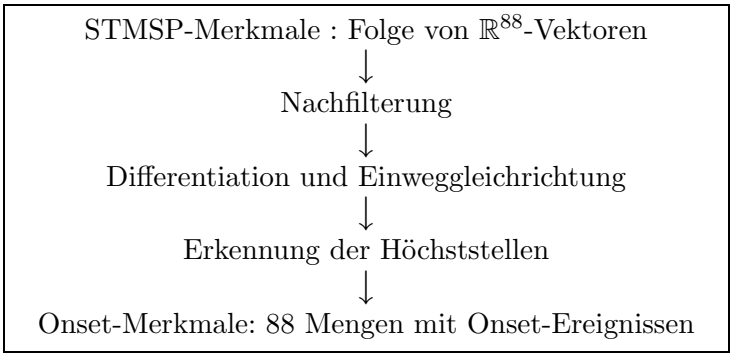


Abbildung 2.15: Berechnung der Onset-Merkmale ausgehend von STMSP-Merkmalen

Einige der Schritte zur Berechnung der Onset-Merkmale dienen maßgeblich der Unterstützung des letzten Schritts. Das Quadrieren der Signalamplitude bei der STMSP-Berechnung stellt sich als günstig heraus, da auf diese Weise starke Ausschläge besser von schwächeren unterschieden werden können. Über die Nachfilterung, die technisch über so genannte Tschebyscheff Typ 2 Filter umgesetzt wird, kann die Anzahl kleiner lokaler Maxima verringert werden kann, welche die Auswahl von Höchststellen erschweren. Im letzten Schritt wird eine Methode zur Auswahl von Höchststellen verwendet, die auf einem lokalen Schwellwertverfahren basiert. Dazu werden absolute und relative Schwellwerte angegeben, die mit dem Ableitungswert verglichen werden. Erkennt die Methode links bzw. rechts von einer potentiellen Höchststelle einen Wertanstieg bzw. Wertabstieg, der mindestens dem relativen Schwellwert entspricht, und liegt die potentielle Höchststelle zudem über dem absoluten Schwellwert, so wird dieser Wert als lokale Höchststelle erfasst. Für Details zu dieser Methode wird auf [MKR04] und die Referenzen darin verwiesen. Abbildung 2.16 illustiert das Ergebnis für das Halbton-Subbandsignal aus Abbildung 2.14.

Die Onset-Merkmale eines Signals werden in Form von 88 Mengen gespeichert (MIDI-Tonhöhen 21 bis 108). Jede Menge beinhaltet die *Onset-Ereignisse*, die in dem Halbton-Subbandsignal erkannt wurden, welches mit der Menge assoziiert wird. Die Onset-Ereignisse sind Elemente aus $\mathbb{R}^2$, deren erste Komponente angibt, zu welcher Zeit ein Ausschlag erkannt

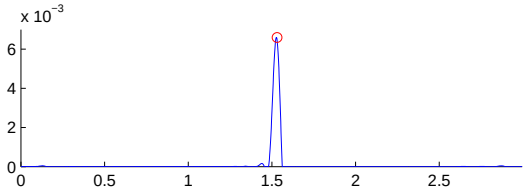


Abbildung 2.16: Ergebnis der Onset-Merkmal Berechnung nach Differentiation und Einweggleichrichtung für das Halbton-Subbandsignal aus Abbildung 2.14. Die erkannte Höchststelle ist umkreist.

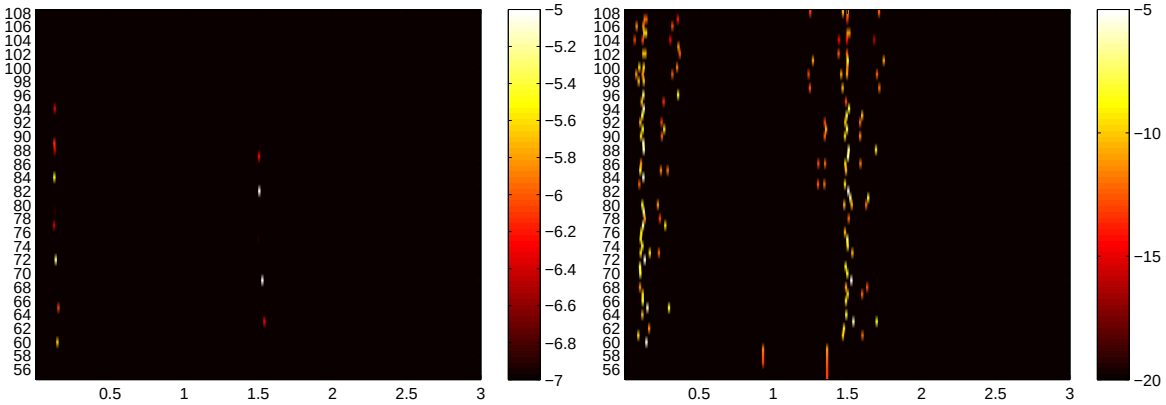


Abbildung 2.17: Onset-Merkmale unter Standardparametern zum Signal aus Abbildung 2.11(a). Dargestellt sind nur die Tonhöhen 55 bis 108. Die Abbildungen unterscheiden sich im dargestellten Wertebereich.

wurde, und deren zweite Komponente die Stärke des Ausschlags festhält. In Abbildung 2.16 wurde nur ein Onset-Ereignis erkannt, so dass die Menge für den Halbton 69 lediglich aus einem Element besteht (Zeit angegeben in Millisekunden): $\{(1532, 6.7 \cdot 10^{-3})\}$.

Abbildung 2.17 zeigt eine vollständige Darstellung der Onset-Merkmale des Beispielsignals aus Abbildung 2.11(a). Dazu wurde für jedes Onset-Ereignis ein kurzer Zeitabschnitt mit dem Wert belegt, der der Ausschlagsstärke des Ereignisses entspricht. Da die Werte relativ klein sind, wurden sie bezüglich einer logarithmischen Skala dargestellt. Die Abbildung enthält zwei Darstellungen der Onset-Merkmale, wobei sich beide lediglich durch eine unterschiedliche Wahl des dargestellten Wertebereichs unterscheiden. Man erkennt in der zweiten Darstellung, dass in diesem Beispiel in nahezu jedem Halbton-Subbandsignal ein Notenanschlag über die Onset-Merkmale erkannt wurde, die meisten davon aber relativ wenig Energie haben. Ursache dafür sind nicht-harmonische, perkussive (d.h. breitbandige, rauschartige) Schwingungen, die beim Notenanschlag ebenfalls vom Klavier erzeugt werden (Tastenmechanik, Hammerbewegung und -aufprall, bestimmte Resonanzeffekte. Siehe dazu auch [FR91] und [Bla98]). Allgemein lässt sich sagen, dass oftmals nicht einfach zu erkennen ist, welche Onset-Ereignisse relevant sind und welche nicht. Einfache Schwellwertverfahren sind hier häufig nicht ausreichend.

Es gestaltet sich schwierig, Standardparameter für Onset-Merkmale anzugeben, ohne tiefgrei-

fende Details der eingesetzten Filterbank aufzugreifen. Deshalb sei nur erwähnt, dass je nach Tonhöhe verschiedene Fensterbreiten zwischen 4.6 ms und 23.8 ms im STMSP-Teilschritt verwendet werden. Die Fensterüberlappung schwankt zwischen 2.3 ms und 11.9 ms. Parameter der Nachfilterung können ohne Erklärung der Tschebyscheff-Filter nicht sinnvoll angegeben werden. Für weitere Details wird auf [MKR04] verwiesen.

2.3.5 Chroma-/CENS-Merkmale

Das Tonsystem, wie es in Abschnitt 2.1.2 eingeführt wurde, basiert auf der Unterteilung in Oktaven und Halbtöne. Ein bestimmter Halbton in einer Oktave wird dabei als doppelt so hoch empfunden wie der entsprechende Halbton in der Oktave darunter. Man hat schon früh festgestellt, dass der menschliche Hörsinn sich um Oktaven unterscheidende Halbtöne anhand einer „Farbigkeit“ in Klassen unterteilt. So werden z.B. die Halbtöne *C0*, *C1*, *C2*,... der Halbtonklasse C zugeordnet. Dieser Effekt wird als so natürlich empfunden, dass es beispielsweise nicht weiter verwundert, wenn eine Tochter den Gesang ihres Vaters ohne Problem in einer anderen Oktave nachsingen kann.

Zur Erhöhung der Robustheit der STMSP-Merkmale hat es sich als vorteilhaft erwiesen, diese Klasseneinteilung nachzumodellieren. Als einfaches Mittel werden in jedem STMSP-Merkmal die Werte der Halbtöne aufaddiert, die der Hörsinn einer Halbtonklasse zuordnet. Diese neuen Merkmale werden *Chroma-Merkmale* genannt, als Anspielung auf die Einteilung in „Farbklassen“. Abbildung 2.18 fasst die Berechnung der Chroma-Merkmale ausgehend von STMSP-Merkmalen in einem Diagramm zusammen.

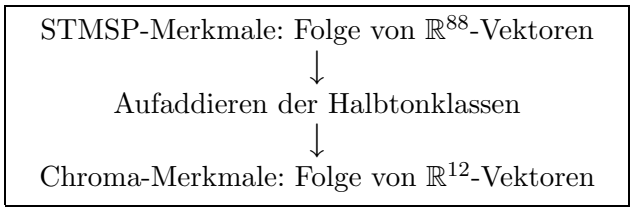


Abbildung 2.18: Berechnung der Chroma-Merkmale, ausgehend von STMSP-Merkmalen.

Bei dieser einfachen Zusammenfassung der Halbtonenergie ergeben sich Vorteile aus der Obertonstruktur. So zeigt sich, dass die Obertöne mehrfach wieder in die Halbtonklasse fallen, in der auch der Grundton liegt. Beispielsweise liegt der erste Oberton immer genau eine Oktave höher als der Grundton und trägt zudem meist ähnlich viel Energie. Die weiteren Obertöne verlieren hingegen oftmals zunehmend an Energie. Durch das Aufsummieren ergibt sich so die Tendenz, dass die Energie in der Halbtonklasse des Grundtons im Vergleich zu der Energie in den Halbtonklassen der Obertöne relativ groß wird. Da ein Chroma-Merkmal somit den Grundton bzw. dessen Halbtonklasse markanter kodiert, ergeben sich Vorteile beim Vergleich von Chroma-Merkmalen.

Ein damit verwandter Vorteil ist die weitgehende Robustheit gegenüber Klangfarbe. Klangfarbe oder Timbre ist ein psychoakkustischer Effekt, durch den der Hörsinn verschiedene Instrumenttypen unterscheiden kann. Ursache ist hauptsächlich eine instrumentabhängige

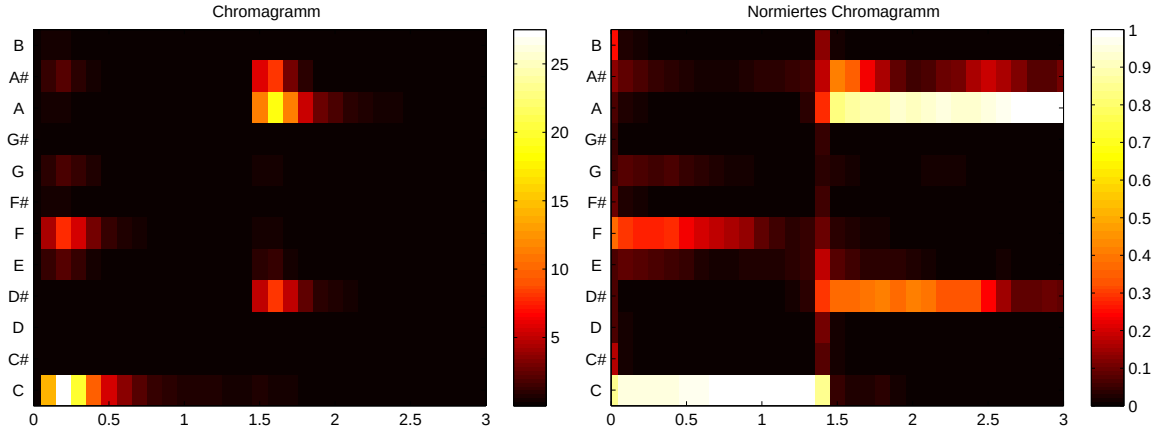


Abbildung 2.19: (Normiertes) Chromagramm zum Signal aus Abbildung 2.11. Für die normierten Chroma-Merkmale wurde die euklidische Norm verwendet.

Verteilung der Energie in den Obertönen. Durch Beschränkung auf eine Oktave und der damit verbundenen Zusammenfassung von Obertönen entfällt dieser Effekt.

Für das kleine Vater/Tochter Beispiel von oben zeigt sich, dass mit Chroma-Merkmalen und deren Oktavinvarianz der Gesang des Vaters sinnvoll mit dem der Tochter verglichen werden kann.

Ein Problem ergibt sich jedoch, sobald Stücke verglichen werden sollen, die sich stark in Lautstärke bzw. Dynamik unterscheiden, da die erzeugten Chroma Merkmale zwar bezüglich ihrer Energieverteilung ähnlich sind, aber nicht bezüglich ihres Energiebetrags. Deshalb kann es sinnvoll sein, Informationen über den Verlauf der Dynamik zu verwerfen. Technisch setzt man diese Anforderung durch Normierung der einzelnen Chroma-Merkmale um, d.h. man ersetzt ein Chroma-Merkmal v durch seine normierte Version $v/\|v\|$. Gängig ist dabei die Betragsnorm $\|v\|_1 := \sum_k |v(k)|$ oder die euklidische Norm $\|v\|_2 := \sqrt{\sum_k |v(k)|^2}$.

Falls $\|v\|$ sehr klein wird (z.B. im Falle einer Pause), wird die Energie nahezu zufällig über die Komponenten in $v/\|v\|$ verteilt. Da somit ein Vergleich mit solchen Merkmalen nicht mehr sinnvoll ist, ersetzt man v nicht mit $v/\|v\|$, sondern mit der normierten gleichverteilten Version. Die so berechneten v bezeichnet man als *normierte Chroma-Merkmale*. Abbildung 2.19 zeigt die Chroma und normierten Chroma-Merkmale des Beispiels aus Abbildung 2.11. Diese Art der Darstellung heißt auch *(normiertes) Chromagramm*.

Je nach Anwendung können unterschiedliche zeitliche Auflösungen der Merkmale sinnvoll sein. Gründe können qualitativer Natur sein, aber auch die Effizienz der anwendungsabhängigen, merkmal-verarbeitenden Methoden betreffen. In dieser Arbeit wird hauptsächlich der Aspekt der Effizienz im Vordergrund stehen, wie später in Abschnitt 3.3 in Form der MsDTW Methode deutlich werden wird. Unabhängig von dem Grund, weshalb unterschiedliche Auflösungen benötigt werden, wäre es möglich, für jede Anwendung eigene Chroma-Merkmale mit passender Fensterbreite und Fensterüberlappung zu berechnen. Die Zerlegung in Halbton-Subbandsignale ist aber ein sehr rechenintensiver Vorgang, weshalb es sich anbietet, die Chroma-Merkmale auf einer höheren zeitlichen Standardauflösung einmalig zu berechnen und diese Auflösung mittels Durchschnittsbildung und Downsampling zu vergröbern. Dies führt zu den

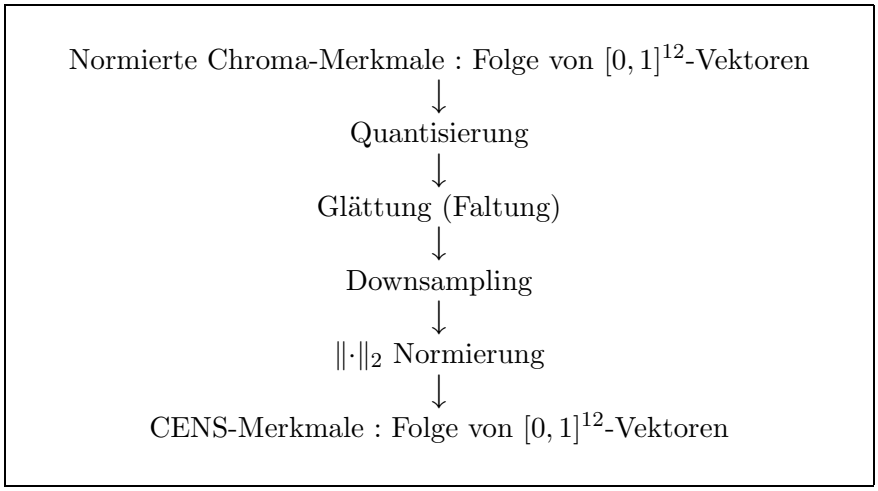


Abbildung 2.20: Berechnung von CENS-Merkmalen aus normierten Chroma-Merkmalen

CENS (*Chroma Energy Normalized Statistics*)-Merkmalen, welche im Folgenden beschrieben werden.

Abbildung 2.20 stellt den Ablauf der Berechnung der CENS-Merkmale schematisch dar. Ausgehend von normierten Chroma-Merkmalen $(v_1, \ldots, v_n)$ wird zunächst eine Quantisierung der Werte durchgeführt, wozu die Funktion $\tau : [0,1] \to \{0,1,2,3,4\}$ definiert wird:

$$\tau(a) := \begin{cases} 0 & \text{falls} \quad 0 \leq a < 0.05 \\ 1 & \text{falls} \quad 0.05 \leq a < 0.1 \\ 2 & \text{falls} \quad 0.1 \leq a < 0.2 \\ 3 & \text{falls} \quad 0.2 \leq a < 0.4 \\ 4 & \text{falls} \quad 0.4 \leq a < 1 \end{cases}$$

Die Quantisierung wird komponentenweise durchgeführt, d.h. aus $v_n = \big(v_n(1), \ldots, v_n(12)\big) \in [0,1]^{12}$ wird $\tau(v_n) := \big(\tau(v_n(1)), \ldots, \tau(v_n(12))\big)$. Anschließend wird $\big(\tau(v_1), \ldots, \tau(v_n)\big)$ komponentenweise mit einem Hann-Fenster der Länge $\mathbf{w} \in \mathbb{N}$ gefaltet, was zu einer gewichteten Durchschnittsbildung der Werte führt. Das anschließende Downsampling um Faktor $\mathbf{d} \in \mathbb{N}$ setzt die erwünschte Verringerung der Auflösung um. Formal sind CENS- und normierte Chroma-Merkmale Elemente der Menge $[0,1]^{12}$, weshalb man auch sagen kann, dass die beiden Merkmalstypen „kompatibel“ sind. Die so entstehenden Merkmale heißen auch $\text{CENS}^{\mathbf{w}}_{\mathbf{d}}$-Merkmale.

Ausgehend von STMSP-Merkmalen mit einer Fensterbreite von 200 ms und -überlappung von 100 ms erhält man eine Zeitauflösung von etwa 100 ms pro Merkmal. Daraus berechnete $\text{CENS}^{\mathbf{41}}_{\mathbf{10}}$ Merkmale haben entsprechend eine Zeitauflösung von etwa 1000 ms, wobei für die Berechnung jedes Merkmals 4100 ms des Ausgangssignals berücksichtigt wurden.

2.3.6 Novelty-Merkmale

Novelty-Merkmale wurden unter Anderem in [Ari02] zur Erkennung von Einsatzzeiten eingesetzt. Prinzipelle Strategie ist das Erkennen von Veränderungen der Amplitude oder des

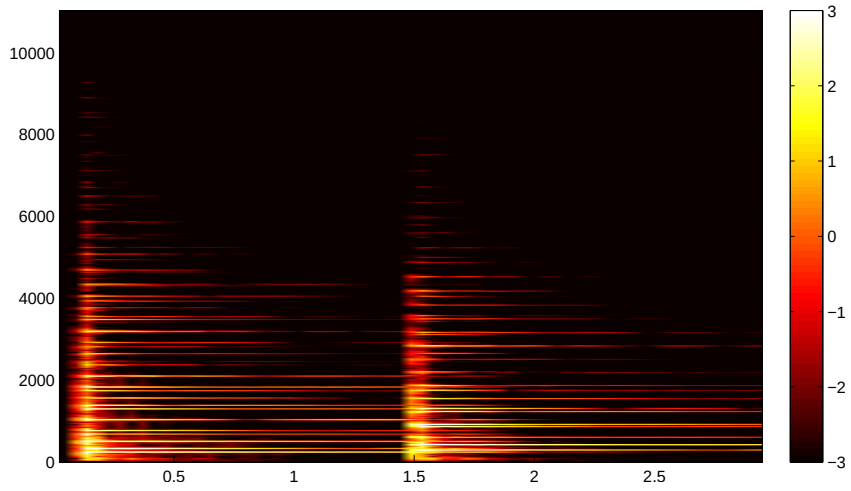


Abbildung 2.21: Spektrogramm des Signals aus Abbildung 2.11(a). Fensterfunktion: Hann. Fensterbreite: 93 ms. Fensterüberlappung: 46.5 ms

Frequenzverlaufs aus dem Spektrogramm. Andere Methoden, die in [Ari02] und [KP95] beschrieben wurden, zeigten sich weniger zur Erkennung von Einsatzzeiten geeignet, da sie sich auf einen Typ von Veränderung konzentrierten und den jeweils anderen ignorierten. Im Unterschied zu den übrigen Merkmalen in diesem Abschnitt, basieren Novelty-Merkmale nicht auf der unter 2.3.1 beschriebenen Halbtonzerlegung. Dadurch ist es nicht direkt möglich, erkannte Einsatzzeiten bestimmten Tonhöhen zuzuordnen, wie es mit Onset-Merkmalen möglich ist. Obwohl Novelty-Merkmale damit nur einen kleinen Teil der Information von Onset-Merkmalen enthalten, ist ihr Einsatz dennoch aus Gründen der Robustheit gerechtfertigt. Zur Definition der Novelty-Merkmale wird zunächst der Begriff der Noveltykurve benötigt.

Definition 2.11 (Noveltykurve). *Sei $x \in \ell^2(\mathbb{Z})$ ein Signal und bezeichne $\left(|\tilde{X}_n|\right)_{n\in\mathbb{Z}}$ das Spektrogramm von x. Dann ist die Noveltykurve $\mathcal{T}$ zu x definiert über:*

$$\begin{aligned} \mathcal{T} &: \mathbb{Z} \to \mathbb{R} \\ \mathcal{T}(n) &:= \| \, |\tilde{X}_{n+1}| - |\tilde{X}_n| \, \|_1 \end{aligned}$$

Ein Algorithmus zur Auswahl von Höchststellen bildet mit der Noveltykurve als Eingabe die eigentlichen Novelty-Merkmale. Dazu wird, wie auch schon in Abschnitt 2.3.4, ein Verfahren basierend auf lokalen Schwellwerten verwendet. Ein Novelty-Merkmal ist dann ein Element aus $\mathbb{R}^2$, dessen erste bzw. zweite Komponente die Zeit bzw. den Wert der erkannten Höchststelle angeben. Abbildung 2.21 zeigt ein Spektrogramm des Beispielsignals aus Abbildung 2.11(a) und Abbildung 2.22 die dazu berechnete Noveltykurve mitsamt der erkannten Höchststellen. Es lässt sich erkennen, dass die Noteneinsatzeiten mit den erkannten Höchststellen übereinstimmen.

Zur Verbesserung der Erkennungsleistung kann neben der Parameterjustierung der Methode zur Auswahl der Höchststellen die Noveltykurve geglättet werden. Dies bietet sich an, wenn eine hohe Zeitauflösung des Spektrogramms gewählt wird, da ansonsten eine große Zahl lokaler Maxima in der Noveltykurve auftritt, die fälschlicherweise als relevante Höchststelle und

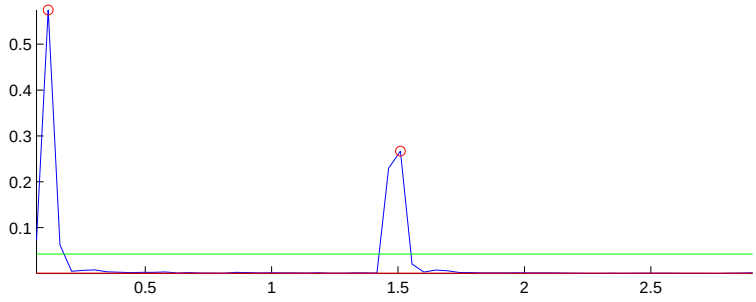


Abbildung 2.22: Noveltykurve (blau) mit erkannten Höchststellen (rote Kreise). Absolute Schwellwertkurve (rot) und relative Schwellwertkurve (grün) der Methode zur Auswahl von Höchststellen.

damit als Noteneinsatz erkannt werden. Zur weiteren Verbesserung kann die Noveltykurve in verschiedenen Zeitauflösungen berechnet werden und durch geeignete Operationen zu einer Kurve zusammengesetzt werden. Oftmals kann so eine Addition der Kurven die Vorteile einer Kurve mit hoher Zeitauflösung mit denen einer niedriger aufgelösten vereinen.

Bei der Berechnung des Spektrogramms wird bei Novelty-Merkmalen standardmäßig ein Hannfenster mit einer Fensterbreite von 93 ms und einer Fensterüberlappung von 46.5 ms verwendet. Ferner wird keine weitere Nachverarbeitung angewendet. Als absoluter Schwellwert für die Höchstellenerkennung wird die Hälfte des globalen arithmetischen Mittels und als relativer Schwellwert die Hälfte der globalen Standardabweichung verwendet.

Kapitel 3

Musiksynchronisation mit MsDTW

Vergleicht man die Interpretationen eines Musikstücks verschiedener Künstler, zeigen sich häufig Unterschiede in Dynamik, Agogik, Instrumentation oder Klangfarbe. Trotz dieser Unterschiede wird jedoch dieselbe Partitur interpretiert. Sollen die Unterschiede konkreter Aufnahmen untersucht werden, mussten sich entsprechende Zeitbereiche in den Varianten bisher manuell identifiziert werden. Mit Hilfe von Synchronisationstechniken kann eine solche Zeitzuordnung automatisch erfolgen. Hierbei wird unter *Synchronisation* ein Verfahren verstanden, das zu einer bestimmten Position innerhalb einer Variante eines Musikstücks die entsprechende Stelle innerhalb einer anderen Variante bestimmen kann. Dabei können sich die Varianten auch anhand ihrer Darstellungsformen unterscheiden. Liegen zum Beispiel Noteninformationen zu einer Audioaufnahme vor, so können diese mittels Synchronisationstechniken zur automatischen Annotation der Aufnahme herangezogen werden. In Abbildung 3.1 werden die in der vorliegenden Arbeit betrachteten Varianten der Musiksynchronisation dargestellt.

Abbildung 3.1: Varianten von Musiksynchronisation.

Im diesem Kapitel werden grundlegende Techniken zur Musiksynchronisation beschrieben. Dazu wird Dynamic Time Warping eingeführt, das in Kombination mit Chroma-Merkmalen aus Kapitel 2 zur Musiksynchronisation eingesetzt werden kann. Durch einen so genannten Multiskalenansatz kann dabei eine hohe Laufzeit- und Speichereffizienz des Verfahrens erreicht werden. Die in diesem Kapitel gebildeten Grundlagen werden in folgenden Kapiteln erweitert und um komplementäre Techniken ergänzt.

3.1 Dynamic Time Warping

Dynamic Time Warping (DTW) ist eine bewährte Technik, um zwei endliche Zeitreihen aneinander auszurichten. Die anschauliche Vorstellung dabei ist, dass die Zeitreihen aus zwei Datenströmen erzeugt wurden, die lokal gestreckte oder gestauchte Versionen des jeweils anderen sind. Man interessiert sich dafür, welche Elemente der beiden Zeitreihen einander entsprechen. Da DTW eine Standardtechnik ist, wird an dieser Stelle nur eine kurze Übersicht gegeben und für eine detaillierte Einführung auf [CM07] verwiesen.

Seien $X = (x_1, x_2, \ldots, x_N)$ und $Y = (y_1, y_2, \ldots, y_M)$ zwei N- bzw. M-Tupel mit Elementen einer Menge $\mathcal{F}$. Eine Funktion, die Ähnlichkeiten zwischen Elementen dieser Tupel beschreibt, wird in der DTW-Terminologie als *Lokales Kostenmaß* bezeichnet und ist formal eine Funktion $c : \mathcal{F} \times \mathcal{F} \to \mathbb{R}_{\geq 0}$. Durch Evaluierung des lokalen Kostenmaßes für alle Paare (x_n, y_m) für $(n, m) \in \{1, \ldots, N\} \times \{1, \ldots, M\}$ erhält man die *Kostenmatrix* $C \in \mathbb{R}^{N \times M}$. Ziel des klassischen DTW wird sein, jedem Element der Tupel mindestens ein Element des jeweils anderen Tupels zuzuordnen, was formal durch einen Warping-Pfad geschieht.

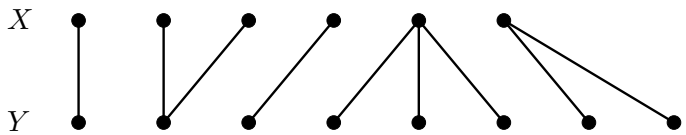


Abbildung 3.2: Zuordnung zwischen Elementen der Tupel X und Y

Definition 3.1 (Warping-Pfad). *Ein Warping-Pfad zu den Folgen X und Y ist ein L-Tupel $p = (p_1, \ldots, p_L)$ mit $p_\ell = (n_\ell, m_\ell) \in \{1, \ldots, N\} \times \{1, \ldots, M\}$ für alle $\ell \in \{1, \ldots, L\}$, welches folgende Bedingungen erfüllt:*

(i) Randbedingung: $p_1 = (1, 1)$ und $p_L = (N, M)$

(ii) Monotoniebedingung: $n_1 \leq n_2 \leq \ldots \leq n_L$ und $m_1 \leq m_2 \leq \ldots \leq m_L$

(iii) Schrittweitenbedingung: $\forall \ell \in \{1, \ldots, L-1\} : p_{\ell+1} - p_\ell \in \{(1, 0), (0, 1), (1, 1)\}$

Ein Warping-Pfad $p = (p_1, \ldots, p_L)$ definiert eine Zuordnung zwischen $X = (x_1, x_2, \ldots, x_N)$ und $Y = (y_1, y_2, \ldots, y_M)$, indem das n_ℓ-te Element von X dem m_ℓ-ten Element von Y zugeordnet wird. Abbildung 3.3 zeigt einen gültigen Warping-Pfad p mit

$$p = \big((1\ 1), (2\ 2), (2\ 3), (2\ 4), (2\ 5), (3\ 5), (4\ 5), (4\ 6), (4\ 7), (5\ 7), (6\ 7), (7\ 7), (8\ 7), (9\ 8)\big)$$

Die *Kosten eines Warping-Pfads* $p = (p_1, \ldots, p_L)$ zwischen $X = (x_1, x_2, \ldots, x_N)$ und $Y = (y_1, y_2, \ldots, y_M)$ bezüglich des lokalen Kostenmaßes c sind definiert über:

$$c_p(X, Y) := \sum_{\ell=1}^{L} c(x_{n_\ell}, y_{m_\ell})$$

Die *DTW-Distanz* zwischen X und Y ist definiert über:

$$DTW(X, Y) := \min\{c_p(X, Y) \mid p \text{ ist Warping-Pfad zwischen } X \text{ und } Y\}$$

Ein *optimaler Warping-Pfad* p^* ist dann ein Warping-Pfad, der minimale Kosten unter allen möglichen Warping-Pfaden hat, d.h.

$$c_{p^*}(X, Y) = DTW(X, Y)$$

Um einen optimalen Warping-Pfad zwischen X und Y zu berechnen, könnte man alle möglichen Warping-Pfade durchlaufen und deren Kosten aufstellen. Die Anzahl möglicher Warping-

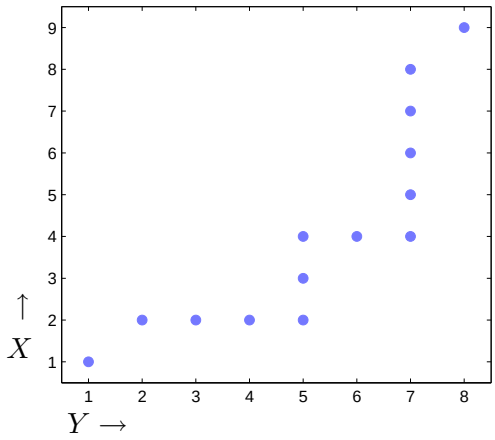


Abbildung 3.3: Gültiger Warping-Pfad zwischen $X = (x_1, \ldots, x_9)$ und $Y = (y_1, \ldots, y_8)$, wobei die erste Komponente vertikal und die zweite horizontal in einem kartesischen Koordinatensystems aufgetragen wurde.

Pfade wächst aber auch unter den Einschränkungen von Definition 3.1 exponentiell in N und M. Mittels eines Algorithmus basierend auf dynamischer Programmierung kann die obere Grenze der algorithmischen Komplexität jedoch auf $O(NM)$ beschränkt werden. Dazu seien $X(1:n) := (x_1, \ldots, x_n)$ mit $n \in \{1, \ldots, N\}$ und $Y(1:m) := (y_1, \ldots, y_m)$ mit $m \in \{1, \ldots, M\}$ Präfixfolgen von X und Y. Über

$$D(n, m) := DTW(X(1:n), Y(1:m))$$

wird die *akkumulierte Kostenmatrix* $D \in \mathbb{R}^{N \times M}$ definiert. Man kann zeigen, dass die akkumulierte Kostenmatrix über folgende Rekursion in $O(NM)$ arithmetischen Operationen bestimmt werden:

$$D(n, m) = \min \begin{cases} D(n-1, m) + c(x_n, y_m), \\ D(n, m-1) + c(x_n, y_m), \\ D(n-1, m-1) + c(x_n, y_m) \end{cases}$$

Die Ränder von D werden dabei über $D(1,1) = c(x_1, y_1)$, $D(n,1) = \sum_{k=1}^{n} c(x_k, y_1)$ für $n \in \{1, \ldots, N\}$ und $D(1,m) = \sum_{k=1}^{m} c(x_1, y_k)$ für $m \in \{1, \ldots, M\}$ initialisiert. Um den Verlauf eines optimalen Warping-Pfads beeinflussen zu können, führt man den Gewichtsvektor $(w_x, w_y, w_{xy}) \in \mathbb{R}^3$ ein. Damit kann die gewichtete akkumulierte Kostenmatrix über folgende Rekursion bestimmt werden:

$$D(n, m) = \min \begin{cases} D(n-1, m) + w_x \cdot c(x_n, y_m), \\ D(n, m-1) + w_y \cdot c(x_n, y_m), \\ D(n-1, m-1) + w_{xy} \cdot c(x_n, y_m) \end{cases}$$

Unter Verwendung von Gewichten können die Ränder von D über $D(1,1) = c(x_1, y_1)$, $D(n,1) = \sum_{k=1}^{n} w_x \cdot c(x_k, y_1)$ für $n \in \{1, \ldots, N\}$ und $D(1,m) = \sum_{k=1}^{m} w_y \cdot c(x_1, y_k)$ für $m \in \{1, \ldots, M\}$ berechnet werden. Ein optimaler Warping-Pfad p kann ausgehend von $D(N, M)$ mittels Backtracking der rekursiven Definition von D berechnet werden, d.h. der Verlauf des Warping-Pfads wird in Abhängigkeit davon bestimmt, welches Argument zur Bildung des Minimums

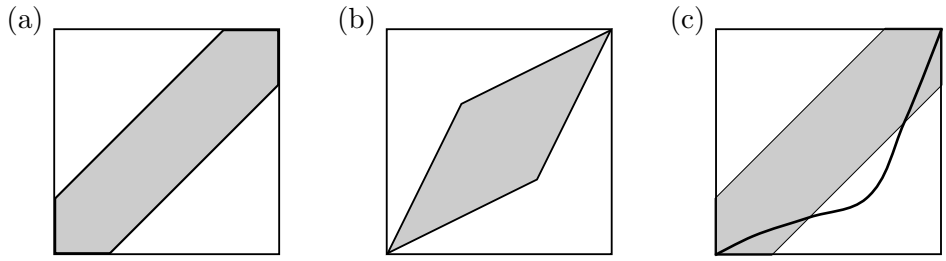


Abbildung 3.4: Einschränkungsbereiche (grau). (a) Sakoe-Chiba Band (b) Itakura Parallelogramm (c) Optimaler Warping-Pfad (schwarze Kurve), der nicht im Einschränkungsbereich verläuft.

verwendet wurde. Dabei wird ersichtlich, dass mit den DTW-Gewichten (w_x, w_y, w_{xy}) eine Richtungspräferenz ausgedrückt wird. Je niedriger ein Gewicht, desto wahrscheinlicher verläuft ein Warping-Pfad in diese Richtung (d.h. in X-Richtung, Y-Richtung oder diagonal). Dazu sei angemerkt, dass der Fall $(w_x, w_y, w_{xy}) = (1, 1, 1)$ eine deutliche Diagonalpräferenz ausdrückt, da für jeden Diagonalschritt (einfache Kosten) jeweils ein Schritt in X- und ein Schritt in Y-Richtung gemacht werden muss (zweifache Kosten). Für Details siehe [RJ93] und [CM07].

3.2 Multiskalen-DTW

Auch wenn die algorithmische Komplexität von DTW durch die Methode über dynamische Programmierung auf $O(NM)$ beschränkt ist, ist die Laufzeit häufig dennoch für große Tupellängen N bzw. M für praktische Anwendungen zu groß. So gilt häufig $N \approx M$, womit sich die Anforderungen von DTW bezüglich Laufzeit und Speicher quadratisch in N bzw. M entwickeln. Vor diesem Hintergrund wurden bereits in der Vergangenheit verschiedene effizienzsteigernde Methoden vorgeschlagen. Grundprinzip dabei ist, nicht die gesamte akkumulierte Kostenmatrix, sondern nur bestimmte Ausschnitte zu berechnen, so genannte *Einschränkungsbereiche*. Der Verlauf eines Warping-Pfads wird auf den Einschränkungsbereich begrenzt, was man formal so ausdrücken kann, dass nicht berechneten Einträgen der Wert ∞ zugewiesen wird. In Abbildung 3.4 werden zwei typische Einschränkungsbereiche dargestellt, das *Sakoe-Chiba-Band* und das *Itakura-Parallelogramm*. Da solch einfache Techniken jedoch kein a-priori-Wissen über den Verlauf eines optimalen Warping-Pfads verwenden, kann nicht ausgeschlossen werden, dass ein optimaler Warping-Pfad, der über eine uneingeschränkte akkumulierte Kostenmatrix berechnet wurde, außerhalb des Einschränkungsbereichs verläuft (Abbildung 3.4(c)).

Aus diesem Grund wurde von Salvador et al. [SC04] ein anderer Ansatz vorgeschlagen. Prinzipielles Vorgehen ist auch hier die Definition eines Einschränkungsbereichs, der aber mit Zusatzwissen so gewählt wird, dass der optimale Warping-Pfad wahrscheinlich in diesem Bereich verläuft. Dazu vergröbert man die Folgen X und Y geeignet und berechnet auf diesen vergröberten Versionen einen optimalen Warping-Pfad. Eine Vergröberungsmethode heißt dabei geeignet, wenn es begründet ist anzunehmen, dass der Warping-Pfad auf der feineren Stufe in etwa so verläuft wie der Warping-Pfad auf der gröberen. Der auf der gröberen Stufe berechnete Warping-Pfad kann dann durch Projektion auf die feinere Stufe zur Definition eines

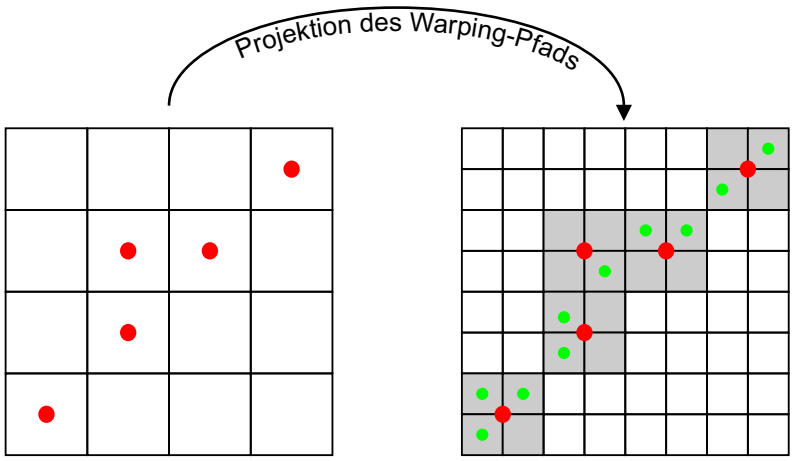


Abbildung 3.5: Definition eines Einschränkungsbereichs bei Multiskalen-DTW. Ein auf grober Auflösung berechneter Warping-Pfad (rot) wird auf die feinere Auflösungsstufe projeziert und definiert einen Einschränkungsbereich (grau). Ein optimaler Warping-Pfad auf der feinen Auflösungsstufe (grün) darf nur innerhalb des Einschränkungsbereichs verlaufen.

Einschränkungsbereichs verwendet werden (siehe Abbildung 3.5). Dieses Vorgehen kann iteriert werden. Man berechnet dazu mehrere Vergröberungsstufen von X und Y. Ausgehend von der gröbsten Stufe wird auf jeder Stufe ein Warping-Pfad berechnet, der auf der nächst feineren Stufe einen Einschränkungsbereich definiert. Daher der Name *Multiskalen-DTW*.

Obwohl beim Multiskalen-Ansatz a-priori-Wissen über den Verlauf eines optimalen Warping-Pfads in die Definition eines Einschränkungsbereichs einfließt, gibt es auch bei dieser Bereichseinschränkung prinzipielle Probleme. So kann auch hier nicht ausgeschlossen werden, dass ein optimaler Warping-Pfad, der über klassisches DTW berechnet wurde, außerhalb der Einschränkung verlaufen würde. Zudem ist für jede Anwendung ein hoher Entwicklungsaufwand notwendig, da geeignete Vergröberungsstrategien gefunden werden müssen.

3.3 Musiksynchronisation mittels DTW

In diesem Abschnitt wird beschrieben, wie Dynamic Time Warping in Verbindung mit Merkmalen aus Abschnitt 2.3 effizient zur Musiksynchronisation eingesetzt werden kann. Bei der Musiksynchronisation sollen ähnliche Abschnitte zweier Varianten eines Musikstücks identifiziert werden. Wie im Abschnitt 2.3 jedoch motiviert wurde, kann die Ähnlichkeit von Audioaufnahmen in den meisten Fällen nicht sinnvoll anhand ihrer Wellenform untersucht werden. Im selben Abschnitt wurden deshalb Merkmale als Vergleichsgrundlage eingeführt, wobei sich normierte Chroma-Merkmale in verschiedenen Arbeiten ([BW05], [NHT03], [MKC05]) besonders zum Vergleich harmoniebasierter Musik geeignet gezeigt haben. Aus diesem Grund wird dieser Merkmalstyp im Folgenden in Verbindung mit DTW zur Musiksynchronisation eingesetzt. Die DTW-Zeitreihen X und Y entsprechen somit Folgen von normierten Chroma-Merkmalen. Die in Abschnitt 2.3.5 definierten Standardparameter der Chroma-Merkmale haben sich dabei in Experimenten bewährt ([MKC05], [MMK06]). Somit wird bei der Berechnung ein Rechteckfenster mit einer Fensterbreite von 200 ms und einer Fensterüberlappung von 100 ms verwendet.

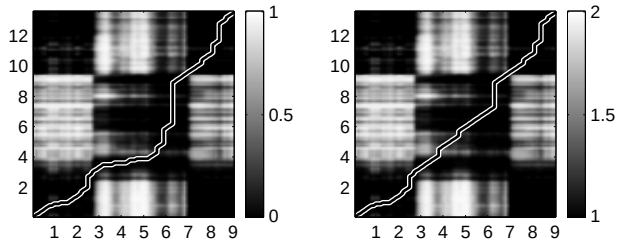


Abbildung 3.6: Optimaler Warping-Pfad unter Verwendung des Kostenmaßes c_α mit $\alpha = 1$ (links) und $\alpha = 2$ (rechts) (aus [MMK06]).

Zum Vergleich zweier normierter Chroma-Merkmale $x, y \in \mathcal{F} = [0,1]^{12}$ wird folgendes lokales Kostenmaß verwendet:

$$c_\alpha : \mathcal{F} \times \mathcal{F} \to \mathbb{R}$$
$$c_\alpha(x,y) := \alpha - \langle x, y \rangle$$

wobei $\alpha \in \mathbb{R}_{\geq 1}$. Dabei bezeichnet $\langle \cdot, \cdot \rangle$ das euklische Skalarprodukt im $\mathbb{R}^{12} \supset \mathcal{F}$. Weiterhin entspricht $\langle x, y \rangle$ wegen $\mathcal{F} = [0,1]^{12}$ dem Kosinus des Winkels zwischen x und y und ist somit ein Orthogonalitätsmaß. Der Parameter α wird ins Kostenmaß eingeführt, um Kontrolle darüber zu erhalten, welchen Verlauf ein Warping-Pfad in Gebieten niedriger Kosten nimmt. Ein Gebiet niedriger Kosten ist ein rechteckiger Ausschnitt einer Kostenmatrix, in dem die Werte des Kostenmaßes relativ klein werden. Verursacht wird dies durch Musikabschnitte mit wenig Varianz in der Spielweise, wie etwa Pausen oder lang gehaltene Akkorde. Beim Vergleich von Chroma-Merkmalen aus solchen Abschnitten ergeben sich rechteckige Gebiete mit niedrigen Kosten innerhalb der Kostenmatrix. Abbildung 3.6 zeigt ein Beispiel eines Gebiets niedriger Kosten.

Man kann zeigen, dass die DTW-Gewichte in Gebieten niedriger Kosten weniger Einfluss auf den Verlauf eines optimalen Warping-Pfads haben, als in anderen Bereichen der Kostenmatrix. Mit größer werdenden Werten für α nimmt dieser Effekt jedoch ab. Experimentell hat sich gezeigt ([MMK06]), dass $\alpha = 2$ unter Verwendung der DTW-Gewichte $(w_x, w_y, w_{xy}) = (1.5, 1.5, 2)$ gute Gesamtergebnisse liefert. Die DTW-Gewichte sind so gewählt, dass der Warping-Pfad in Gebieten gleicher Kosten tendenziell diagonal verläuft, die Diagonalpräferenz jedoch im Allgemeinen nicht zu dominant ist. Abbildung 3.6 zeigt ein Beispiel für den Einfluss des Parameters α. Damit ist die Spezifikation der DTW-Parameter vollständig, womit ein komplettes Verfahren zur Musiksynchronisation beschrieben ist.

Die hohen Anforderungen von DTW an Speicher und Laufzeit erweisen sich jedoch als problematisch für die praktische Einsetzbarkeit dieses Verfahrens. Da einfache bereichseinschränkende Verfahren oftmals entweder wenig effizienzsteigernd wirken oder ein hohes Risiko bergen, dass der optimale Warping-Pfad außerhalb des Einschränkungsbereichs verläuft, stellen sie praktisch häufig keine Alternative zur kompletten Berechnung der akkumulierten Kostenmatrix dar. In [MMK06] konnte nun gezeigt werden, dass der DTW-Multiskalen-Ansatz für das Musiksynchronisations-Szenario adaptiert werden kann. Wie in Abschnitt 3.2 beschrieben, muss dafür eine geeignete Methode angegeben werden, mit der die DTW-Merkmalsfolgen vergröbert werden können. Geeignet heißt dabei, dass ein Warping-Pfad auf einer gröberen Auflösung mit hoher Wahrscheinlichkeit einen ähnlichen Verlauf nimmt wie ein Warping-Pfad, der auf einer feineren Auflösung berechnet wurde. Der Begriff „geeignet“ lässt jedoch schwer

exakt spezifizieren und bedeutet letztlich, dass nur empirisch erprobt werden kann, ob eine Methode wirklich geeignet ist.

In Abschnitt 2.3 wurden $\text{CENS}_{\mathbf{d}}^{\mathbf{w}}$-Merkmale als eine vergröberte Version normierter Chroma-Merkmale eingeführt. Ob diese Merkmale eine „geeignete“ Methode zur Vergröberung im Sinne des Multiskalenansatzes sind, wurde in [MMK06] untersucht. Auf feinster Stufe wurden dazu normierte Chroma-Merkmale mit Standardparametern verwendet, auf den Stufen zwei und drei $\text{CENS}_{\mathbf{10}}^{\mathbf{41}}$- und $\text{CENS}_{\mathbf{30}}^{\mathbf{121}}$-Merkmale. Tests auf großen Datenbeständen haben gezeigt, dass ein dreistufiger Multiskalenansatz basierend auf diesen Merkmalen dieselben Warping-Pfade berechnete wie klassisches DTW. Zur Definition eines Einschränkungsbereichs auf den verschiedenen Stufen wird der berechnete optimale Warping-Pfad auf die nächst feinere Stufe projeziert. Die dabei verwendete Strategie wurde für in späteren Kapiteln entwickelte Methoden verallgemeinert und wird deshalb in Abschnitt 4.2 in ähnlicher Form separat beschrieben. Für eine genaue Analyse der Effizienz und Beschreibung der Strategie zur Definition eines Einschränkungsbereichs wird auf [Mat06] verwiesen.

Mit Abschluss dieses Kapitels steht somit ein erprobtes, effizientes Verfahren zur Synchronisation harmoniebasierter Musik zur Verfügung, das im Folgenden als *MsDTW-Verfahren* bezeichnet wird. In Kapitel 4 wird das MsDTW-Verfahren als Grundlage für Erweiterungen dienen.

Kapitel 4

Erweiterung der MsDTW Synchronisationsmethode

Die in Kapitel 3 beschriebene MsDTW-Methode hat sich als robuste und zuverlässige Lösung zur Synchronisation harmoniebasierter Musik erwiesen. Unter Verwendung eines DTW-Multiskalenansatzes können dabei auch sehr lange Musikstücke miteinander synchronisiert werden. Experimentell konnte die Qualität dieser Methode durch Tests auf großen Datenbeständen belegt werden (siehe auch [MMK06]). Die Zuverlässigkeit der MsDTW-Methode ist dabei insbesondere von folgenden Aspekten abhängig:

1. Die zeitliche Auflösung der eingesetzten Chroma-Merkmale muss geeignet gewählt werden.

2. Die zu synchronisierenden Musikstücke müssen harmoniebasiert sein.

Wird eine zu feine Zeitauflösung gewählt, so erfassen die Chroma-Merkmale zunehmend kurzzeitige, nicht harmonische Elemente im Musikstück, wie z.B. Geräusche der Tastenmechanik beim Klavier. Die harmonische Grundstruktur der Musik wird in Folge dessen nicht mehr robust wiedergegeben. Zu grobe Auflösungen hingegen erlauben lediglich ungenaue Zeitzuordnungen. Verwendet man die unter Abschnitt 2.3 eingeführten Standardparameter für Chroma-Merkmale, erreicht die Synchronisation Zeitauflösungen im dreistelligen Millisekunden-Bereich, was für Retrieval-Anwendungen ausreichend ist, für andere Szenarien hingegen bereits zu grob aufgelöst sein kann.

Des Weiteren kann nicht prinzipiell beantwortet werden, ob eine Methode, die wie MsDTW harmoniebasiert arbeitet, „bessere“ Ergebnisse liefert als eine Methode, die Rhythmus, Klangfarbe oder Dynamik als Grundlage einer Synchronisationstechnik verwendet. Um Musikstücke unabhängig vom Genre mit hoher Qualität synchronisieren zu können, müssen verschiedene solcher Musikaspekte zur Synchronisation eingesetzt werden.

In diesem Abschnitt wird dazu ein erster Schritt beschrieben, wobei die rein harmoniebasierte MsDTW-Methode um dynamikbezogene Zusatzinformationen in Form von Noteneinsatzzeiten erweitert wird. Im Anschluss werden Methoden vorgestellt, die eine effiziente Berechnung der zuvor entwickelten Erweiterungen ermöglichen. Obwohl die Zeitauflösung der Chroma-Merkmale dabei aus Robustheitsgründen nicht verändert wird, kann die Synchronisationsgenauigkeit dennoch unter Einbeziehung von Noteneinsatzzeiten oftmals erhöht werden.

4.1 Erweiterung mittels Merkmalen zur Erkennung von Einsatzzeiten

4.1.1 Unerwünschte Ergebnisse unter Verwendung der MsDTW-Methode

Bevor im weiteren Verlauf Erweiterungen der MsDTW-Methode beschrieben werden, wird zunächst dargestellt, welche Art von Problemen bei dieser Methode auftreten und durch Erweiterungen beseitigt werden sollen. Dabei stehen vor allem das Kostenmaß und die verwendeten Merkmale im Vordergrund. Deshalb wird vorab betrachtet, welche Informationen bei der Extraktion von normierten Chroma-Merkmalen erhalten bleiben und welche durch Invarianzen verloren werden. Dies wird anhand eines einfachen Beispiels ausgeführt. Die zugehörigen Beispieldaten sind in Abbildung 4.1 visualisiert. Dargestellt sind die Wellenformen und (normierten) Chromagramme einer Klavieraufnahme eines mehrfach angespielten C4, sowie einer zeitlich modifizierten Variante.

Damit Stücke unterschiedlicher globaler bzw. lokaler Dynamik sinnvoll über Chroma-Merkmale verglichen werden können, ist meist eine Normierung notwendig. Andererseits stellt die Normierung einen Informationsverlust dar, da ein Großteil der Information über zeitliche Energieunterschiede aus den Chroma-Merkmalen entfernt wird. In Abbildung 4.1 wird deutlich, dass durch den fehlenden Dynamikverlauf in der normierten Chromadarstellung nicht mehr gut erkennbar ist, zu welchen Zeitpunkten Noten angespielt werden. Im Falle des Klaviers oder ähnlicher Saiteninstrumente wird dieser Effekt teilweise dadurch gemindert, dass Notenanschläge von Nebeneffekten perkussiver, nicht-harmonischer Natur begleitet werden. Im Spektrogramm zeigt sich dies als gleichförmige Verteilung der Energie über alle Spektralbänder hinweg. Wie in Abbildung 4.1 zu sehen, ist dieser Effekt aber nicht sehr stark ausgeprägt: Im Chromagramm erkennt man diesen Effekt kaum. Auftretende Energie in den Tonklassen E, G und $A\#$ stammt aus den Obertönen des angespielten $C4$ und nicht aus nicht-harmonischen Anteilen (Ausgehend vom $C4$ entspricht der erste Oberton einem $C5$, der zweite einem $G5$, der dritte einem $C6$, der vierte Oberton in etwa einem $E6$, der fünfte in etwa einem $G6$ und der sechste in etwa einem $A\#6$). Im normierten Chromagramm erkennt man bei genauer Betrachtung einen leichten Energieanstieg in allen Tonklassen zu den Anspielzeiten. Die Energie der perkussiven Anteile ist aber relativ klein. In der Abbildung wird sie nur deshalb sichtbar, weil eine spezielle Farbskala verwendet wird, die auch relativ betragsschwachen Werten noch eine von schwarz unterscheidbare Farbe zuweist.

Bei DTW können jedoch selbst geringe Unterschiede in den Merkmalen noch zu großen globalen Änderungen eines optimalen Warping-Pfads führen. Aus diesem Grund wird nun das Ergebnis der MsDTW-Methode aus Kapitel 3.3 auf der Folge der normierten Chroma-Merkmale aus Abbildung 4.1 unter dem Aspekt untersucht, ob die Noteneinsatzzeiten korrekt durch den berechneten Warping-Pfad aufeinander abgebildet werden. Abbildung 4.2 zeigt den berechneten optimalen Warping-Pfad und die zugehörige Kostenmatrix. Unter Beachtung der Farbskala ist erkennbar, dass kaum Schwankungen der Werte in der Kostenmatrix auftreten. Der dargestellte Wertebereich musste extrem verkleinert werden, damit überhaupt Strukturen sichtbar werden. Es zeigen sich drei vertikale bzw. horizontale Streifen, mit gegenüber der Umgebung leicht erhöhten Kosten. Dabei treten an den Überschneidungen geringere Kosten auf, da hier Notenanschläge aufeinander treffen und damit durch ihren leicht perkussiven Charakter bezüglich der Kostenfunktion geringere Kosten verursachen. Die ebenfalls erkennbaren Randeffekte sind zunächst nicht weiter interessant.

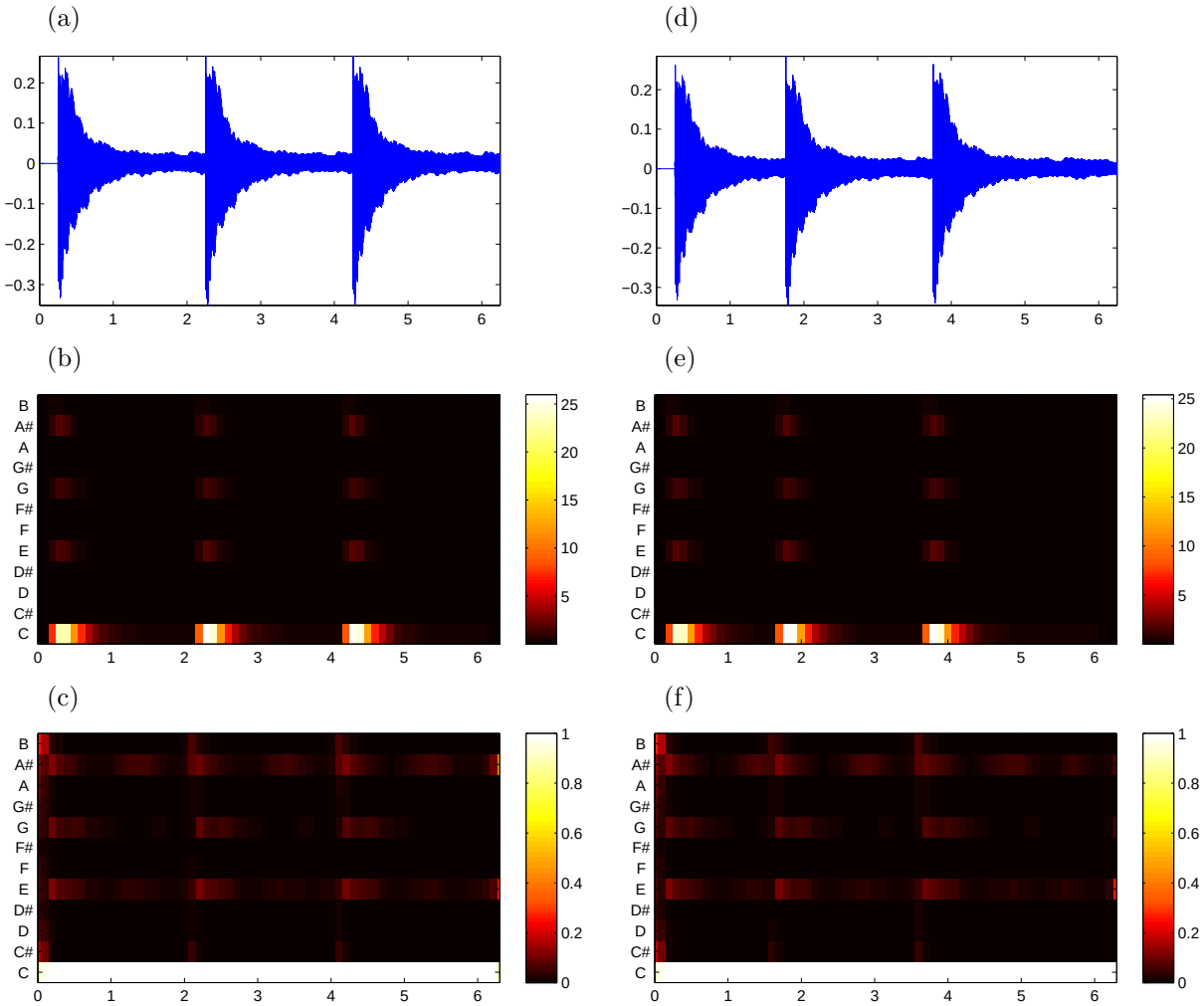


Abbildung 4.1: (a) Wellenform einer Klavieraufnahme eines dreifach wiederholten C4. (b) Chromagramm der Aufnahme. (c) normiertes Chromagramm der Aufnahme. (d)-(f) analog, wobei die Anspielzeit der Noten variiert wurde. Die Zeit ist horizontal in Sekunden aufgetragen.

Durch die in Abschnitt 3.3 eingeführten DTW-Gewichte $(w_x, w_y, w_{xy}) = (1.5, 1.5, 2)$ wird jedoch ein diagonal verlaufender Warping-Pfad begünstigt. Erst nachdem die Diagonalpräferenz mit $(w_x, w_y, w_{xy}) = (1, 1, 2)$ aufgehoben wurde, verlief der optimale Warping-Pfad durch diese „Schneisen“ niedriger Kosten. Zur Erinnerung: In Kapitel 3.3 wurde beschrieben, dass eine leichte Diagonalpräferenz wünschenswert ist, damit ein optimaler Warping-Pfad in Gebieten gleicher Kosten keinen willkürlichen Verlauf nimmt. So kommt es mit den bestehenden Methoden zu einem unbefriedigenden Synchronisationsergebnis, da diese perkussiven Anteile in den Chroma-Merkmalen nicht ausreichend Einfluss in DTW haben, um Noteneinsatzzeiten exakt aufeinander abzubilden. In Abbildung 4.2 ist dies durch Pfeile markiert: Die durchgezogen gezeichneten Pfeile symbolisieren die berechnete Paarung, wünschenswert wäre jedoch die Paarung mit dem gestrichelt gezeichneten Pfeil. Damit weicht das Synchronisationsergebnis an dieser Stelle vom gewünschten Ergebnis um 0.5 Sekunden ab.

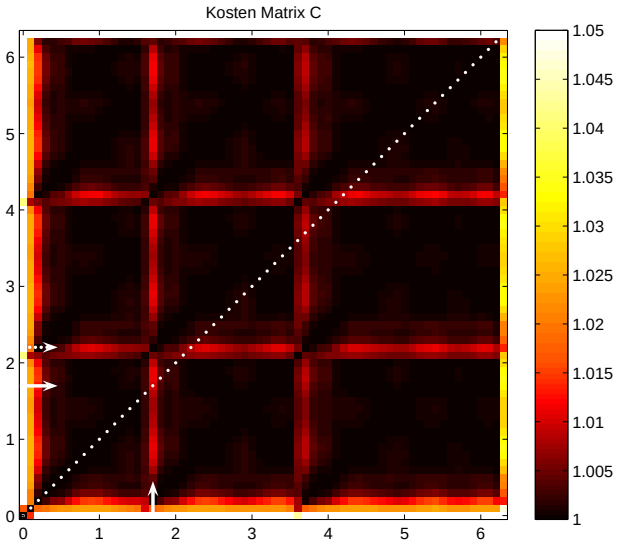


Abbildung 4.2: Kostenmatrix mit optimalem Warping-Pfad unter Verwendung der MsDTW-Methode mit Standardparametern. Vertikal aufgetragen ist die Folge normierter Chroma-Merkmale aus Abbildung 4.1(c), horizontal aus Abbildung 4.1(f). Die Achsenbeschriftung gibt die Zeit in Sekunden an.

4.1.2 CN-Merkmale und das lokale Kostenmaß $c^{CN}_{\alpha,\beta}$

Um dem gerade beschriebenen Effekt entgegen zu wirken, wird der rein harmoniebasierte Ansatz der MsDTW-Methode im Folgenden um dynamikbezogene Informationen in Form von Noteneinsatzzeiten ergänzt. Liegt ein Musikstück im MIDI-Format vor, so lassen sich Notenliste und Einsatzzeiten auf direkte Weise auslesen. Ausgehend von einer Audioaufnahme ist die Extraktion der benötigten Einsatzzeiten jedoch ungleich schwieriger. Im Abschnitt 2.3 wurden zwei Merkmalstypen vorgestellt, die sich prinzipiell eignen würden: Novelty- und Onset-Merkmale. Abbildungen 4.3 und 4.4 zeigen, welches Ergebnis diese Merkmale auf den Beispieldaten aus Abbildung 4.1 liefern. Die Parameter der Novelty- und Onset-Merkmale entsprechen den Standardparametern, wie unter 2.3.6 und 2.3.4 festgelegt.

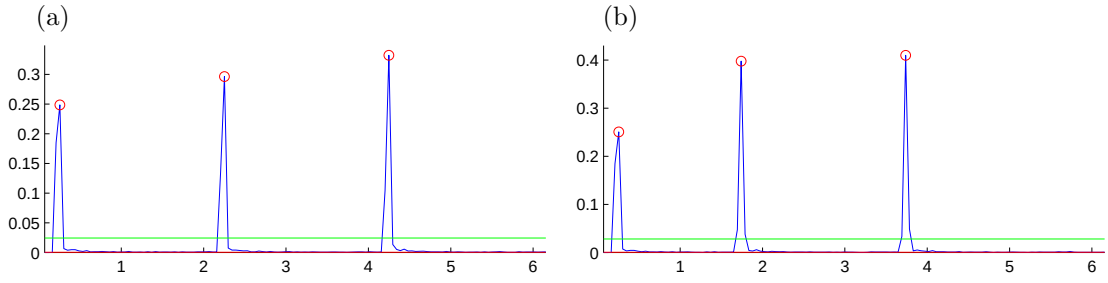


Abbildung 4.3: Novelty-Merkmale bzw. Novelty-Kurven unter Standardparametern. Eingabedaten wie in Abbildung 4.1(a) und 4.1(d) dargestellt.

Onset-Merkmale wurden zur Erkennung von Noteneinsatzzeiten entworfen, weshalb sich diese zunächst gut für diese Aufgabe zu eignen scheinen. In der Praxis stellt sich aber heraus,

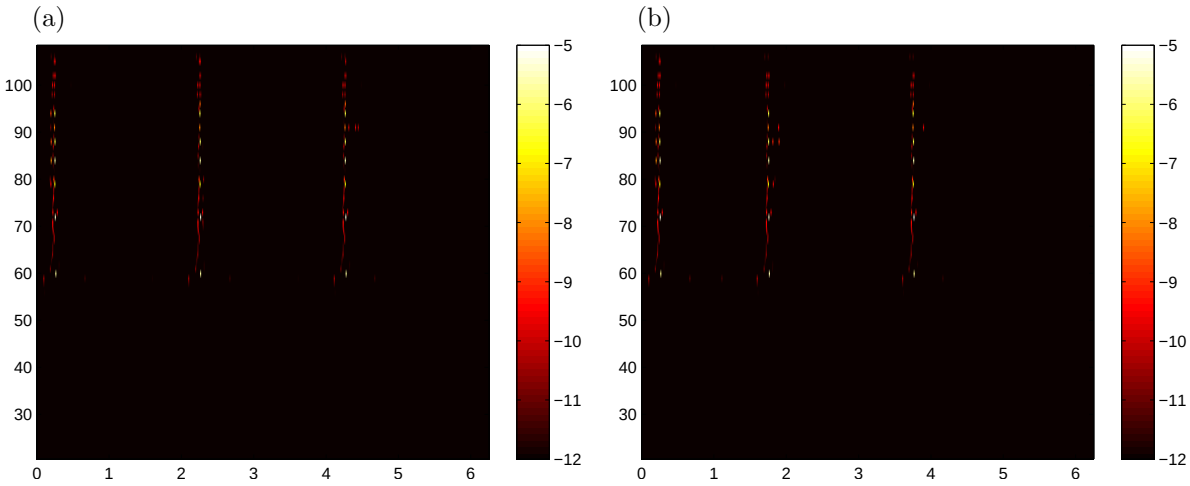


Abbildung 4.4: Onset-Merkmale unter Standardparametern (dargestellt bezüglich einer logarithmischen Skala). Eingabedaten wie in Abbildung 4.1(a) und 4.1(d) dargestellt.

dass Onset-Merkmale relativ unrobust bezüglich Klangfarbe oder perkussiven Anteilen im Klavierspiel sind. Sollen zuverlässig Noteneinsatzzeiten erkannt werden, gestaltet sich die Parametrisierung aus diesem Grund oftmals schwierig.[1] Da lediglich eine robuste Erkennung von Noteneinsatzzeiten erreicht werden soll, werden an dieser Stelle Novelty-Merkmale verwendet.

Um das Zusatzwissen über Noteneinsatzzeiten formal in die bestehenden Methoden einzugliedern, wird die Merkmalsmenge $\mathcal{F} = [0,1]^{12}$ der normierten Chroma-Merkmale um eine Dimension erweitert.

Definition 4.1 (Chroma-Novelty-Merkmale). *Die normierte Chroma-Novelty Merkmalsmenge ist definiert über:*

$$\mathcal{F}_{CN} := [0,1]^{12} \times \{0,1\}$$

Ihre Elemente heißen normierte Chroma-Novelty (CN)-Merkmale.

Dabei modelliert eine 1 in der letzten Dimension einen erkannten Noteneinsatz, eine 0 entsprechend keinen. Abbildung 4.5 stellt dies anhand des Beispiels aus Abbildung 4.1(a) exemplarisch dar. Auf den CN-Merkmalen wird nun formal ein lokales Kostenmaß definiert.

Definition 4.2 (lokales Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$). *Seien:*

- $(c_1, n_1), (c_2, n_2) \in \mathcal{F}_{CN}$ *CN-Merkmale*
- $\alpha \in \mathbb{R}_{\geq 1}$
- $\beta \in \mathbb{R}_{\geq 0}$

[1] Eine Methode, die diese Probleme behandelt und Onset-Merkmale direkt zur Synchronisation einsetzt, wird in [MKR04] vorgestellt.

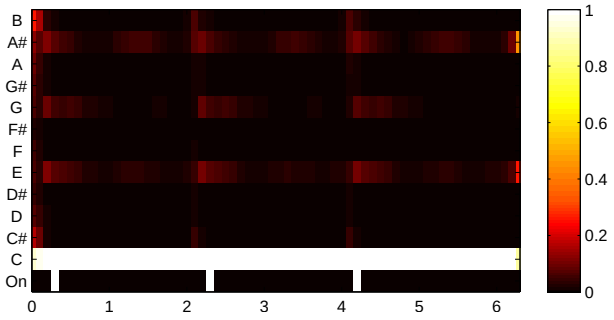


Abbildung 4.5: CN-Merkmale zur Wellenform aus Abbildung 4.1(a). „On“ zeigt die erkannten Noteneinsatzszeiten.

Dann ist das lokale Kostenmaß $\boldsymbol{c}^{CN}_{\alpha,\beta}$ *definiert über:*

$$\boldsymbol{c}^{CN}_{\alpha,\beta}\big((c_1,n_1),(c_2,n_2)\big) := \alpha - (1 + n_1 \cdot n_2 \cdot \beta)\langle c_1, c_2\rangle$$

Anschaulich wird mit dieser Definition das Kostenmaß c_α aus Abschnitt 3.3 um einen zusätzlichen kostensenkenden Faktor β erweitert, der genau dann zum Tragen kommt, wenn die beiden zu vergleichenden Merkmale jeweils einen Noteneinsatz enthalten (dann gilt $n_1 \cdot n_2 = 1$).

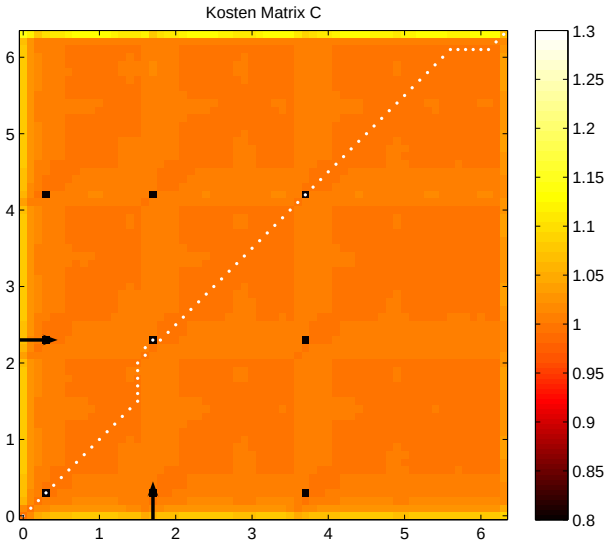


Abbildung 4.6: Kostenmatrix mit optimalem Warping-Pfad unter Verwendung von CN-Merkmalen und dem Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$. Vertikal aufgetragen ist die Folge von CN-Merkmalen zu der Wellenform wie in Abbildung 4.1(a), horizontal zu der Wellenform wie in Abbildung 4.1(d). Die Achsenbeschriftung gibt die Zeit in Sekunden an.

Abbildung 4.6 zeigt die Kostenmatrix und den berechneten optimalen Warping-Pfad unter Verwendung von CN-Merkmalen und dem Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$, wobei als Eingabe die Wellenformen aus Abbildung 4.1 dienen. Die Parameter (α, β) wurden mit $(2, 1)$, alle sonstigen Parameter mit Standardwerten belegt. Man erkennt, wie nun ein optimaler Warping-Pfad die Noteneinsatzzeiten korrekt aufeinander abbildet (markiert durch Pfeile).

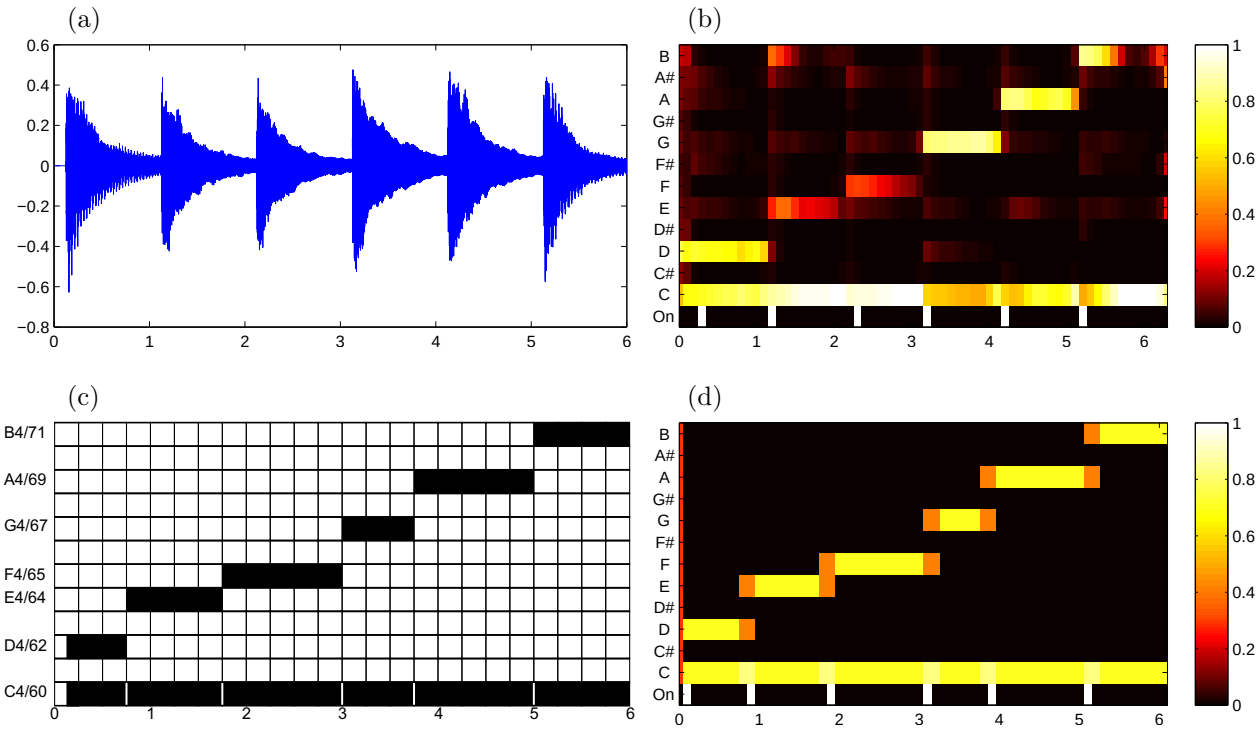


Abbildung 4.7: (a) Wellenform einer Klavieraufnahme der Akkordfolge C4/D4, C4/E4, C4/F4, C4/G4, C4/A4, C4/B4 (b) normierte Chroma-Novelty-Merkmale der Aufnahme (c) Piano Roll Darstellung von MIDI-Daten. Es werden dieselben Akkorde gespielt, nur die Anspielzeiten sind variiert (d) normierte Chroma-Novelty-Merkmale der MIDI-Daten.

4.1.3 CNO-Merkmale und das lokale Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CNO}$

Das bisher verwendete Beispiel der drei C4 Noten wird nun variiert, um im Folgenden ein Problem zu beschreiben, das durch die Verwendung des lokalen Kostenmaßes $\mathbf{c}_{\alpha,\beta}^{CN}$ entsteht. Dazu werden die drei C4 Noten durch die Akkordfolge C4/D4, C4/E4, C4/F4, C4/G4, C4/A4, C4/B4 ersetzt. Zudem wird im Folgenden nicht mehr eine Synchronisation zwischen zwei Wellenformen betrachtet, sondern zwischen einer Wellenform und MIDI-Daten. Abbildung 4.7 stellt die Wellenform, die Piano-Roll-Darstellung der MIDI-Daten und die zugehörigen CN-Merkmale dar.

Abbildung 4.8 zeigt die zugehörige Kostenmatrix unter Verwendung des lokalen Kostenmaßes $\mathbf{c}_{\alpha,\beta}^{CN}$ und CN-Merkmalen. Zusätzlich ist stilisiert ein Warping-Pfad eingezeichnet (blau), anhand dessen im Folgenden ein typisches Problem mit dem Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CN}$ beschrieben wird. Dieser Warping-Pfad enthält die Zuordnung $(4.2, 5.1)$, die in Abbildung 4.8 mit einem Pfeil markiert wurde. Verglichen mit der Umgebung wurden an dieser Stelle geringere Kosten durch das Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CN}$ vergeben. Ursache dafür ist, dass in der Wellenform bei 4.2 Sekunden und in den MIDI-Daten bei 5.1 Sekunden ein Noteneinsatz durch die CN-Merkmale erkannt wurde, deren Kombination eine Kostensenkung im Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CN}$ bewirkt hat. Vergleicht man diese Angaben aber mit Abbildung 4.7, so fällt auf, dass durch die Zuordnung $(4.2, 5.1)$ semantisch unterschiedliche Notenereignisse aufeinander abgebildet werden. So wird in der

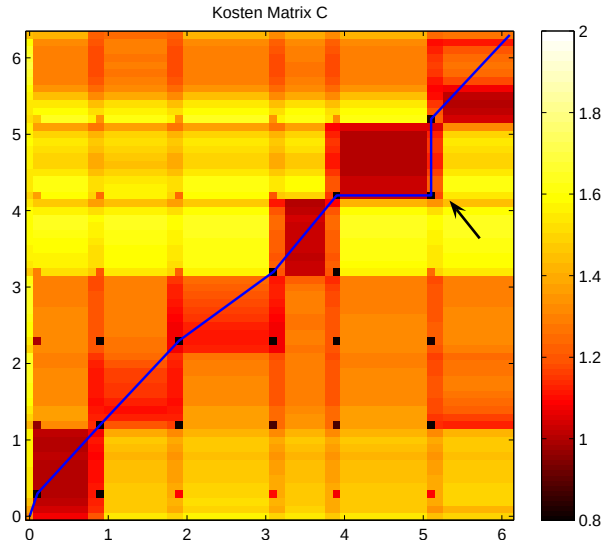


Abbildung 4.8: Kostenmatrix mit stilisiertem Warping-Pfad unter Verwendung des Kostenmaßes $\mathbf{c}^{CN}_{\alpha,\beta}$. Vertikal aufgetragen ist die Folge von CN-Merkmalen zu der Wellenform aus Abbildung 4.7(a), horizontal die Folge von CN-Merkmalen zu den MIDI-Daten aus Abbildung 4.7(c).

Wellenform bei 4.2 Sekunden ein C4/A4 Akkord angespielt, während in den MIDI-Daten ein C4/B4 Akkord bei 5.1 Sekunden gespeichert ist. Durch die Kostensenkung bei $(4.2, 5.1)$ wird somit lokal ein unerwünschter Pfadverlauf begünstigt.

In Experimenten zeigte sich, dass degenerierte Warping-Pfade, wie der in Abbildung 4.8 stilisiert dargestellte, praktisch relevant sind. Verursacht wird dies allgemein durch das Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$, bei dem pauschal eine Kostensenkung vergeben wird, wenn in zwei CN-Merkmalen ein Noteneinsatz erkannt wurde. Eine detailliertere Untersuchung dazu findet sich in Kapitel 6. Um die Bildung solcher degenerierten Warping-Pfade zu verhindern, können verschiedene Strategien verfolgt werden:

1. Ein optimaler Warping-Pfad bezüglich $\mathbf{c}^{CN}_{\alpha,\beta}$ kann als eine korrigierte Version eines optimalen Warping-Pfads bezüglich c_α verstanden werden. Ist bereits ein Warping-Pfad bezüglich c_α bekannt, so kann man ausnutzen, dass ein optimaler Warping-Pfad bezüglich $\mathbf{c}^{CN}_{\alpha,\beta}$ von diesem nur in eingeschränktem Maße abweichen sollte. Eine Methode dazu wird in Abschnitt 4.2 betrachtet.

2. Ein weiterer Ansatz ist, eine β-Kostensenkung im Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$ nicht mehr pauschal für jede Kombination von Noteneinsätzen zu vergeben, sondern zusätzlich zu prüfen, ob in beiden Fällen die gleichen Tonhöhen angespielt wurden. Kann dies ausgeschlossen werden, soll keine oder eine verminderte β-Kostensenkung vergeben werden.

Der zweite Ansatz wird nun für den Fall einer Audio-MIDI Synchronisation demonstriert. Aus MIDI-Daten kann eine Notenliste erstellt werden, welche einen direkten Zugriff auf Einsatzzeiten und Tonhöhen ermöglicht. Die Extraktion solcher Informationen aus einem Audiosignal ist aber ungleich schwieriger. Im Folgenden werden dazu Onset-Merkmale verwendet, über die jedoch häufig eine zu große Anzahl von Einsatzzeiten erkannt wird, was bereits in Abschnitt

2.3.4 ausgeführt wurde. Dabei ist die Wahrscheinlichkeit eines False-Positives deutlich größer als die eines False-Negatives.

Mit Hilfe eines neuen Kostenmaßes soll nun erreicht werden, dass beim Vergleich zweier CN-Merkmale nur dann eine β-Kostensenkung in vollem Maße für erkannte Einsatzzeiten vergeben wird, wenn zusätzlich alle Tonhöhen, die im MIDI angespielt werden, auch im Audiosignal angeschlagen werden. Zeigen Onset-Merkmale, dass eine bestimmte Tonhöhe nicht angespielt wurde, so kann aufgrund der niedrigen Wahrscheinlichkeit für False-Negatives ausgeschlossen werden, dass sich die Notenanschläge im MIDI und Audiosignal entsprechen.

Diese anschaulich formulierte Idee wird nun durch Definition eines neuen lokalen Kostenmaßes formalisiert, wozu auch die verwendete Merkmalsmenge erweitert werden muss.

Definition 4.3 (Chroma-Novelty-Onset-Merkmale). *Die normierte Chroma-Novelty-Onset Merkmalsmenge ist definiert über:*

$$\mathcal{F}_{CNO} := [0,1]^{12} \times \{0,1\} \times \{0,1\}^{88}$$

Ihre Elemente heißen normierte Chroma-Novelty-Onset (CNO)-Merkmale.

Aus Modellierungssicht werden somit CN-Merkmale um 88 Dimensionen erweitert. Diese weiteren Einträge kodieren durch eine 1, ob in dem Zeitraum, den ein CNO-Merkmal umfasst, ein bestimmter Halbton angespielt wurde.

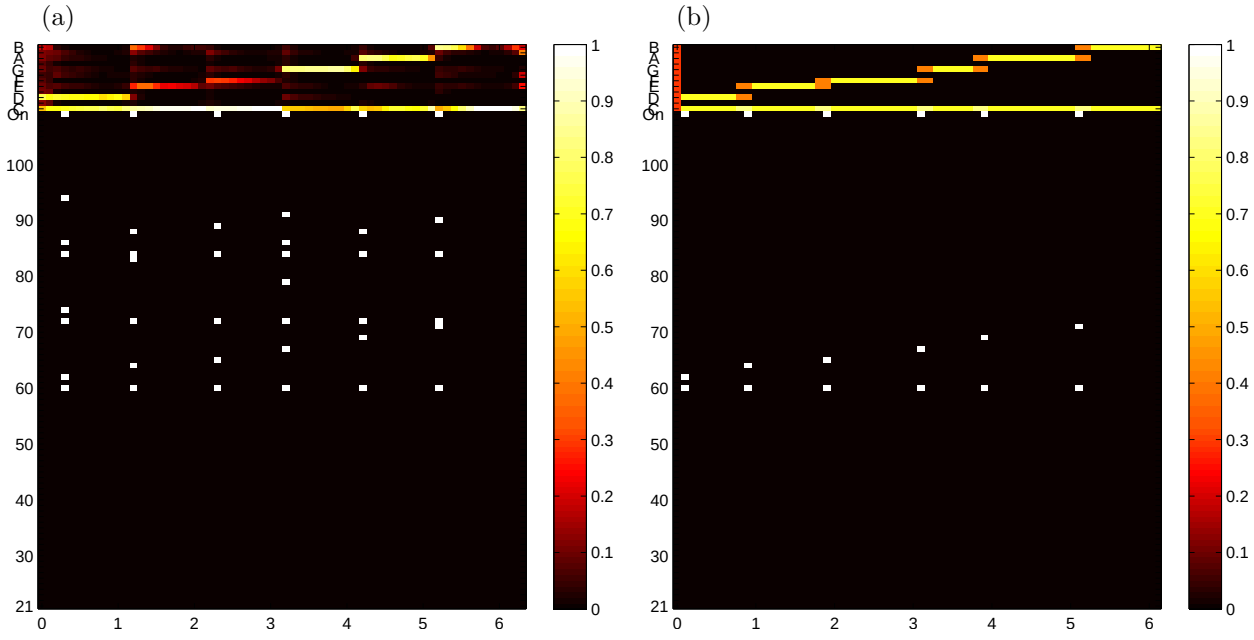


Abbildung 4.9: (a) CNO-Merkmale für die Wellenform aus Abbildung 4.7(a) (b) CNO-Merkmale für die MIDI-Daten dargestellt in Abbildung 4.7(c).

Die Berechnung von CNO-Merkmalen aus einem MIDI-Stück beginnt mit der Erzeugung von normierten Chroma-Novelty-Merkmalen, wie bereits bekannt. Die 88 weiteren Dimensionen werden gesetzt, indem die im MIDI-Stück gespeicherte Notenliste durchlaufen und für jede

enthaltene Note der entsprechende Eintrag in den 88 Dimensionen auf 1 gesetzt wird. Die restlichen Einträge werden mit dem Wert 0 belegt[2]. Abbildung 4.9(b) zeigt das Ergebnis einer Berechnung von CNO-Merkmalen für MIDI-Daten.

Bei der Berechnung der CNO-Merkmale für Audiosignale wird versucht, dem Vorgehen bei MIDI-Daten möglichst analog zu folgen. Die Informationen über gespielte Noten werden dabei aus Onset-Merkmalen gewonnen. Begonnen wird auch hier mit der Berechnung der Chroma-Novelty-Merkmale. Danach wird für jeden durch die Novelty-Merkmale gefundenen Noteneinsatz nach allen Onset-Ereignissen gesucht, die in einer bestimmten zeitlichen Umgebung um den Noteneinsatz liegen. Sobald die logarithmierte Energie eines gefundenen Onset-Ereignisses über einem Schwellwert liegt, wird in dem CNO-Merkmal eine 1 für die entsprechende Tonhöhe vermerkt, ansonsten eine 0. Abbildung 4.9(a) zeigt das Ergebnis der CNO-Merkmalsberechnung für das Audiosignal aus Abbildung 4.7(a).

Damit kann nun formal das lokale Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CNO}$ definiert werden.

Definition 4.4 (lokales Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CNO}$). *Seien:*

- $(c_1, n_1, o_1), (c_2, n_2, o_2) \in \mathcal{F}_{CNO}$ *CNO-Merkmale*
- $\alpha \in \mathbb{R}_{\geq 1}$
- $\beta \in \mathbb{R}_{\geq 0}$

Dann ist das lokale Kostenmaß $\boldsymbol{c}_{\alpha,\beta}^{CNO}$ definiert über:

$$\boldsymbol{c}_{\alpha,\beta}^{CNO}\big((c_1, n_1, o_1), (c_2, n_2, o_2)\big) := \alpha - (1 + n_1 \cdot n_2 \cdot \left(\frac{\langle o_1, o_2\rangle}{\|o_2\|_2^2}\right)^2 \cdot \beta)\langle c_1, c_2\rangle$$

Das Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CNO}$ unterscheidet sich von $\mathbf{c}_{\alpha,\beta}^{CN}$ lediglich durch den Term $\left(\frac{\langle o_1,o_2\rangle}{\|o_2\|_2^2}\right)^2$, mit dem der kostensenkende Faktor β zusätzlich gewichtet wird. $\langle o_1, o_2\rangle$ entspricht dabei der Anzahl der Halbtöne, die in beiden Merkmalen als angespielt erkannt wurden, und $\|o_2\|_2^2$ der Anzahl Halbtöne, die im zweiten Merkmal als angespielt erkannt wurden. Man beachte, dass das lokale Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CNO}$ nicht mehr symmetrisch ist. Man setzt die oben anschaulich formulierte Idee für ein neues Kostenmaß um, indem man für (c_1, n_1, o_1) *die CNO-Merkmale des Audiosignals* und für (c_2, n_2, o_2) *die CNO-Merkmale des MIDI-Stücks* wählt.

Zum Abschluss dieses Abschnitts werden in Abbildung 4.10 die verschiedenen Kostenmaße und Merkmalstypen aus diesem Abschnitt gegenübergestellt. Dargestellt sind die zugehörigen Kostenmatrizen und optimalen Warping-Pfade. Des Weiteren sei noch erwähnt, dass die in diesem Abschnitt benutzen Beispieldaten synthetisch sind und lediglich der Erklärung dienen. In Kapitel 6 wird betrachtet, zu welchen Ergebnissen die vorgestellten Methoden auf realen Daten führen. Anhang A beinhaltet eine kurze Referenz zu den Matlab-Methoden, mit denen diese Kostenmaße und Merkmale umgesetzt wurden.

[2]Die unteren 88 Dimensionen sind dünn besetzt. Deshalb kann die Speicherung aus Effizienzgründen praktisch über geeignete Listen realisiert werden.

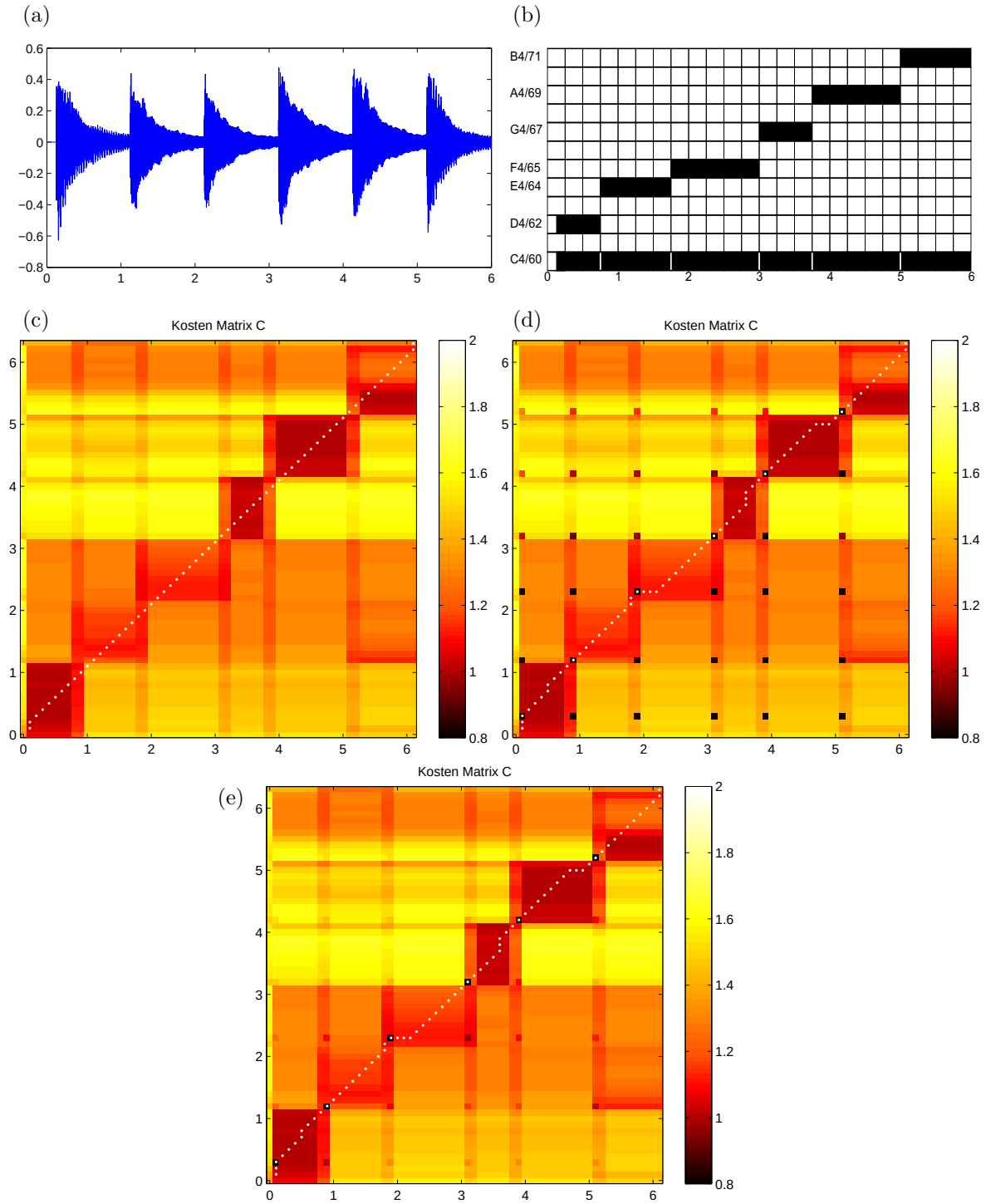


Abbildung 4.10: Kostenmatrizen mit eingezeichneten optimalen Warping-Pfaden. Die Eingabedaten (a) (vertikal aufgetragen) und (b) (horizontal aufgetragen). (c) Ergebnis mit Kostenmaß c_α und normierten Chroma-Merkmalen. (d) Ergebnis mit Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$ und CN-Merkmalen. (e) Ergebnis mit Kostenmaß $\mathbf{c}^{CNO}_{\alpha,\beta}$ und CNO-Merkmalen. Die Parameter (α, β) wurden mit $(2, 1)$ belegt. Alle sonstigen Parameter wurden mit den Standardwerten belegt, die in den entsprechenden Abschnitten definiert wurden.

4.2 Effiziente Umsetzung der erweiterten Methoden

Da sich die MsDTW-Methode bereits durch eine hohe Speicher- und Laufzeiteffizienz auszeichnet, wird im Folgenden untersucht, inwiefern der dort verwendete Multiskalenansatz auch für die in Abschnitt 4.1 eingeführten erweiterten Methoden adaptiert werden kann. Dabei wird auch die algorithmische Komplexität der erweiterten Methoden mit der der MsDTW-Methode verglichen.

Für einen DTW-Multiskalenansatz muss stets eine Methode angegeben werden, mit der die DTW-Merkmalsfolgen geeignet vergröbert werden können. Für die MsDTW-Methode wurden dazu CENS-Merkmale entwickelt, über die verschiedene Auflösungsstufen eines Chromagramms effizient angegeben werden können. Nachdem verschiedene solcher Auflösungsstufen berechnet sind, kann auf jeder Stufe, ausgehend von der gröbsten, ein optimaler Warping-Pfad berechnet werden, der auf der nächst feineren Stufe einen DTW-Einschränkungsbereich definiert. Durch dieses iterierte Vorgehen konnten hohe Steigerungsraten der Speicher- und Laufzeiteffizienz erreicht werden.

Ein erster Gedanke wäre nun, ein Äquivalent zu CENS-Merkmalen für CN- bzw. CNO- Merkmale zu entwickeln, so dass verschiedene Auflösungsstufen der CN- bzw. CNO- Merkmale effizient angegeben werden können. Dabei stellt sich jedoch ein prinzipielles Problem. Da die Erweiterungen aus Abschnitt 4.1 sehr kurzzeitige Ereignisse in Form von Noteneinsatzzeiten modellieren, ist es nicht sinnvoll solche Kurzzeitinformationen in zeitlich vergröberte Merkmale zu übernehmen. So werden im MsDTW-Ansatz CENS-Merkmale mit einer zeitlichen Auflösung von 3 Sekunden eingesetzt, wobei für die Berechnung 12.1 Sekunden des Ausgangssignals berücksichtigt werden. Noteneinsätze können über eine so grobe Zeitauflösung nicht sinnvoll modelliert werden. Man kann nun aber annehmen, dass Informationen über Kurzzeitereignisse wie Noteneinsatzzeiten in zeitlich vergröberten Versionen der CN- bzw. CNO-Merkmale lediglich eine untergeordnete Rolle einnehmen. Aus diesem Grund können CENS-Merkmale nicht nur als vergröberte Version normierter Chroma-Merkmale aufgefasst werden, sondern in dem gleichen Sinne auch für CN- bzw. CNO-Merkmale verwendet werden.

Somit steht nun eine Vergröberungsmethode für die erweiterten Merkmale aus Abschnitt 4.1 zur Verfügung. Dem Vorgehen von MsDTW folgend könnten nun CN- bzw. CNO-Merkmale auf der feinsten und $\mathrm{CENS}_{\mathbf{10}}^{\mathbf{41}}$- und $\mathrm{CENS}_{\mathbf{30}}^{\mathbf{121}}$-Merkmale auf zwei vergröberten Auflösungsstufen in einem Multiskalen-DTW Verfahren eingesetzt werden. Es erweist sich jedoch als günstig, MsDTW wie bisher zu verwenden und den so berechneten Warping-Pfad zur Definition eines Einschränkungsbereichs für DTW unter Verwendung der CN- bzw. CNO-Merkmale einzusetzen (der Ablauf ist in Abbildung 4.11 für das Beispiel aus Abbildung 4.10 dargestellt). Die Gesamtlaufzeit wird durch diesen Zusatzschritt in den meisten Fällen kaum erhöht. Aus Modellierungssicht ergeben sich durch dieses Vorgehen jedoch einige Vorteile:

1. Zur Laufzeit wird ein zusätzlicher Warping-Pfad erzeugt, der weitere Informationen über die Qualität einer Synchronisation liefern kann.

2. Der über CN- bzw. CNO-Merkmale berechnete Warping-Pfad kann als korrigierte Version des von MsDTW berechneten Warping-Pfads aufgefasst werden. Wird nun ein DTW-Einschränkungsbereich über den von MsDTW berechneten Warping-Pfad definiert, kann so über geeignete Heuristiken festgelegt werden, wie weit eine Korrektur abweichen darf. Auf diese Weise kann die Bildung degenerierter Warping-Pfade, wie sie

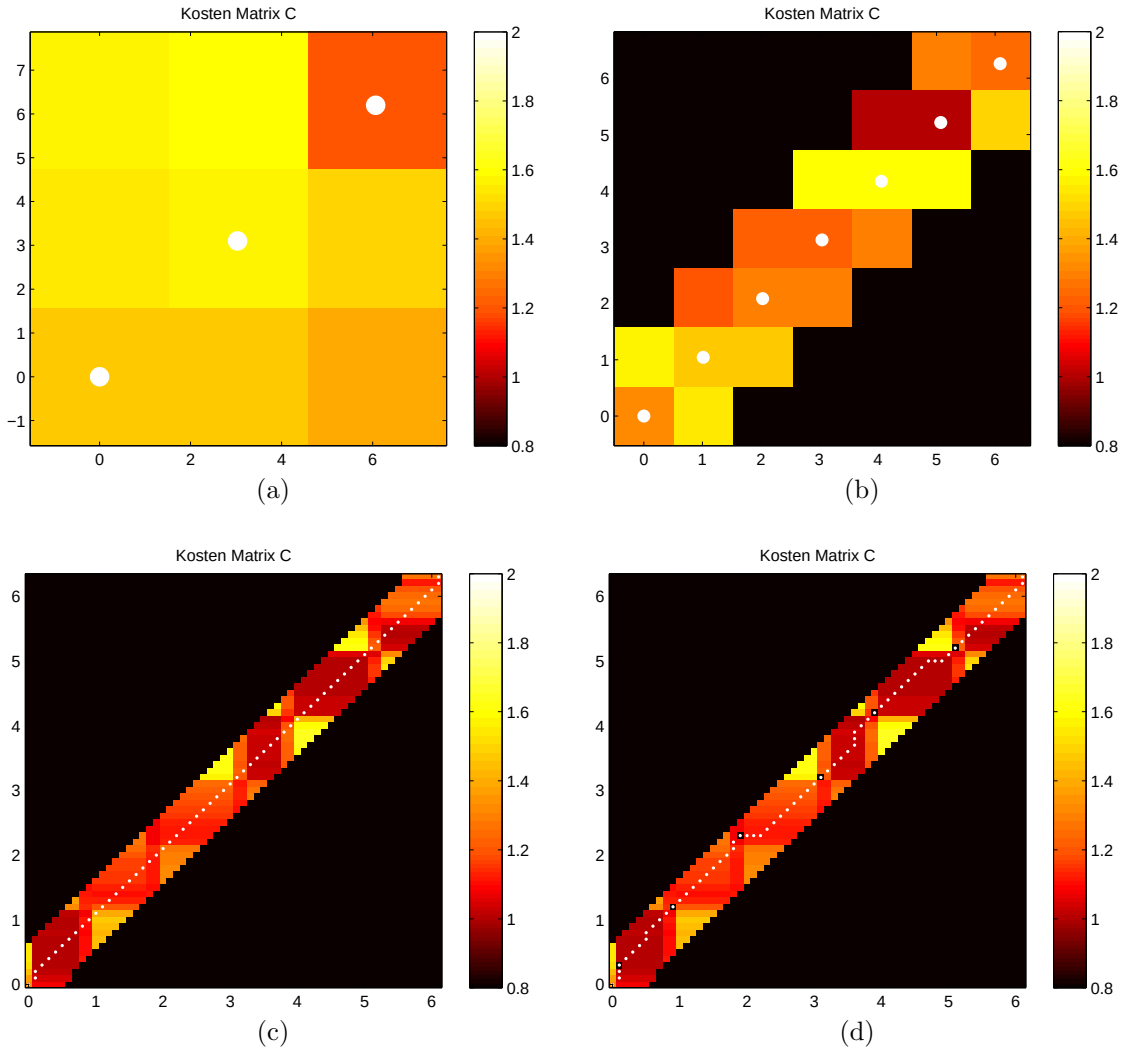


Abbildung 4.11: Effiziente Berechnung eines Warping-Pfads bezüglich CN-Merkmalen:
(a) Schritt 1 aus MsDTW ($\text{CENS}_{\mathbf{30}}^{\mathbf{121}}$-Merkmale).
(b) Schritt 2 aus MsDTW ($\text{CENS}_{\mathbf{10}}^{\mathbf{41}}$-Merkmale).
(c) Schritt 3 aus MsDTW (normierte Chroma-Merkmale).
(d) zusätzlicher Schritt für erweiterte Merkmale (hier: CN-Merkmale).

unter Verwendung von CN-Merkmalen auftreten, oftmals verhindert werden (siehe auch Abschnitt 4.1).

Im Rahmen der vorliegenden Arbeit wurde dieser Ansatz in Form zweier Methoden implementiert. Im Folgenden wird jeweils beschrieben, wie ein Einschränkungsbereich ausgehend von einem gegebenen Warping-Pfad definiert werden kann, sowie die benötigten Anpassungen des klassischen DTW-Algorithmus. Ein Einschränkungsbereich ist dabei formal wie folgt definiert:

Definition 4.5 (Einschränkungsbereich). *Seien N und M die Längen zweier Merkmalsfolgen X und Y. Dann ist ein Einschränkungsbereich $\mathcal{B}$ für DTW zwischen X und Y ein Element der Menge $\{0,1\}^{N\times M}$.*

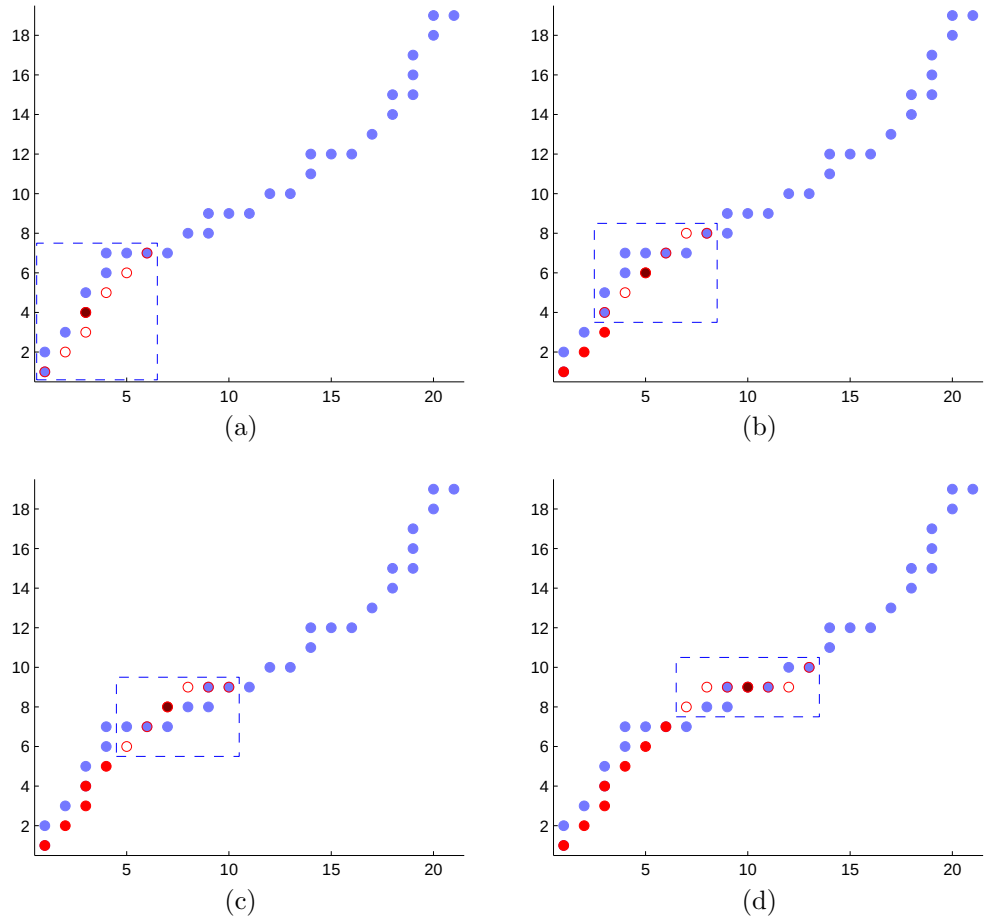


Abbildung 4.12: DTW mit MovingWindow Bereichseinschränkung mit Fensterbreite 7 (in horizontaler bzw. Y-Richtung) in mehreren Schritten. Der vorgegebene Warping-Pfad ist blau, der bis zum aktuellen Schritt berechnete globale Warping-Pfad rot und der im aktuellen Schritt berechnete lokale Warping-Pfad durch rote Kreise gekennzeichnet. Der braune Punkt markiert den Startpunkt (x_s, y_s) des Ausschnitts für den nächsten Schritt. Der aktuelle Aussschnitt ist mit einem Rechteck markiert.

Hat ein Eintrag in $\mathcal{B}$ den Wert 1, so wird der entsprechende Eintrag der Kostenmatrix in die DTW-Berechnung mit einbezogen. Ein Wert von 0 bedeutet analog, dass der entsprechende Kostenmatrixeintrag nicht verwendet wird.

Im weiteren Verlauf wird allgemein der *vorgegebene Warping-Pfad* vom *zu berechnenden Warping-Pfad* unterschieden.

4.2.1 DTW mit MovingWindow-Bereichseinschränkung

Beim Entwurf der MovingWindow Bereichseinschränkung sollten insbesondere zwei Ziele erreicht werden:

1. geringer Speicherbedarf
2. einfache Umsetzbarkeit

Wegen der geforderten einfachen Umsetzbarkeit sollen möglichst wenige Änderungen an bestehenden DTW-Implementationen erforderlich sein. Zudem sollte die Komplexität von zusätzlich benötigten Verwaltungsstrukturen gering sein. Das prinzipielle Vorgehen dazu ist nun, ausgehend von einem kleinen Ausschnitt der Kostenmatrix, einen lokalen Warping-Pfad über eine unveränderte DTW-Implementation zu berechnen. Im nächsten Schritt wird dieser Ausschnitt anhand des vorgegebenen Warping-Pfads verschoben und erneut ein lokaler Warping-Pfad berechnet. Die Verschiebung wird dabei geschickt gewählt, so dass man die lokalen Warping-Pfade auf einfache Weise zu einem globalen Warping-Pfad zusammenfügen kann. Das genaue Vorgehen ist in Algorithmus 4.1 festgehalten und in Abbildung 4.12 anhand eines einfachen Beispiels illustriert.

Algorithmus 4.1 : DTW mit MovingWindow-Bereichseinschränkung

Eingabe : Zeitreihe X der Länge N, Zeitreihe Y der Länge M, Fensterbreite d, vorgegebener Warping-Pfad p zwischen X und Y

Ergebnis : Warping-Pfad $\tilde{p}$ zwischen X und Y

Initialisierung: Startpunkt $(x_s, y_s) := (1, 1)$. $\tilde{p} = \emptyset$

1. Endpunkt (x_e, y_e) neu setzen:
   $y_e = \min(y_s + d, M)$. Suche dann das größte x_e mit $(x_e, y_e) \in p$
2. Berechne lokalen Warping-Pfad über klassisches DTW auf dem Rechteck bestimmt durch Startpunkt und Endpunkt: $p^{lokal} = (p_1^{lokal}, \ldots, p_k^{lokal})$
3. Erweitern des globalen Warping-Pfads: $\tilde{p} = \tilde{p} \cup (p_1^{lokal}, \ldots, p_{\lfloor k/2 \rfloor - 1}^{lokal})$
4. Falls $(x_e, y_e) = (N, M)$, gehe zu Ende
5. Startpunkt auf das mittlere Element in p^{lokal} setzen: $(x_s, y_s) = p_{\lfloor k/2 \rfloor}^{lokal}$
6. Gehe zu 1

Ende: $\tilde{p} = \tilde{p} \cup (p_{\lfloor k/2 \rfloor}^{lokal}, \ldots, p_k^{lokal})$ und Ausgabe $\tilde{p}$

Eine zu beachtende implizite Annahme der MovingWindow-Bereichseinschränkung ist, dass sich der neu zu berechnende Warping-Pfad nicht nur global auf der gesamten Kostenmatrix ähnlich verhält wie der vorgegebene, sondern auch lokal in den Ausschnitten. Dies äußert sich darin, dass man den Endpunkt des Ausschnitts auf dem vorgegebenen Warping-Pfad wandern lässt, wodurch man die lokalen Warping-Pfade zwingt, auf dem vorgegebenen Warping-Pfad zu enden. Man hofft dabei, dass ein lokaler Warping-Pfad innerhalb des Ausschnitts genug Freiheit erhält, um vom vorgegebenen Warping-Pfad abweichen zu können. Aus diesem Grund sollte die Fensterbreite d nicht zu klein gewählt werden.

Abbildungen 4.13 zeigt als Beispiel eine Kostenmatrix mit vorgegebenem Warping-Pfad unter Verwendung von DTW mit MovingWindow-Bereichseinschränkung. Dabei wurden zwei Varianten eines Ausschnitts aus Ludwig van Beethovens Klaviersonate Nr.1 (Opus 2 Nr.1 oder „kleine Appassionata“) verwendet.

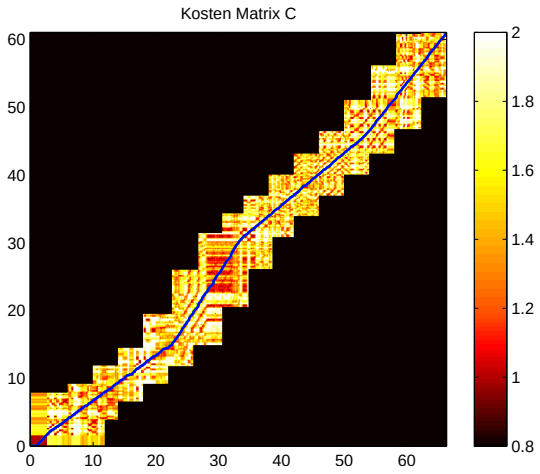


Abbildung 4.13: Kostenmatrix unter MovingWindow-Bereichseinschränkung. Schwarze Bereiche sind nicht Teil des Einschränkungsbereich. Der vorgegebene Warping-Pfad ist blau eingezeichnet.

4.2.2 DTW mit Tube-Bereichseinschränkung

Weicht der zu berechnende Warping-Pfad lokal stark vom vorgegebenen ab, kann dies zu Problemen mit der MovingWindow Bereichseinschränkung unter Verwendung kleiner Fensterbreiten führen. In diesem Fall kann alternativ eine Tube-Bereichseinschränkung eingesetzt werden, womit meist jedoch ein leicht erhöhter Speicherverbrauch sowie Implementierungsaufwand verbunden ist. Eine DTW-Berechnung mit Tube-Bereichseinschränkung verläuft dabei in zwei Schritten:

1. Definition eines Einschränkungsbereichs mittels geeigneter Datenstrukturen.

2. DTW über eine angepasste Version des klassischen Algorithmus.

Zur Definition des Einschränkungsbereichs wird der Ursprung einer so genannten *Erzeugerstruktur* über jedem Zuordnungspunkt des vorgegebenen Warping-Pfads positioniert. Jeder so überdeckte Eintrag der Kostenmatrix wird zum Einschränkungsbereich hinzugefügt. Auf diese Weise bildet sich eine „schlauchartige“ Umgebung um den vorgegebenen Warping-Pfad. Abbildung 4.14 zeigt zwei Beispiele.

Definition 4.6 (Erzeugerstruktur). *Sei* $E \in \{0,1\}^{d \times d}$. *Dann heißt* E *Erzeugerstruktur, falls* $E(\lceil d/2 \rceil, \lceil d/2 \rceil) = 1$. d *heißt die Größe und* $E(\lceil d/2 \rceil, \lceil d/2 \rceil)$ *der Ursprung von* E.

Wenn man beliebige Erzeugerstrukturen zulässt, ergibt sich das Problem, in welcher Datenstruktur der Einschränkungsbereich gespeichert wird. Setzt man Definition 4.5 direkt um, muss für jeden Eintrag der Kostenmatrix gespeichert werden, ob er Teil des Einschränkungsbereichs ist oder nicht. Daraus folgt jedoch ein Speicherbedarf von $O(N \cdot M)$. Bei der Tube-Bereichseinschränkung beschränkt man sich jedoch auf feste Erzeugerstrukturen und kann in diesem Fall den Einschränkungsbereich speichereffizient angeben.

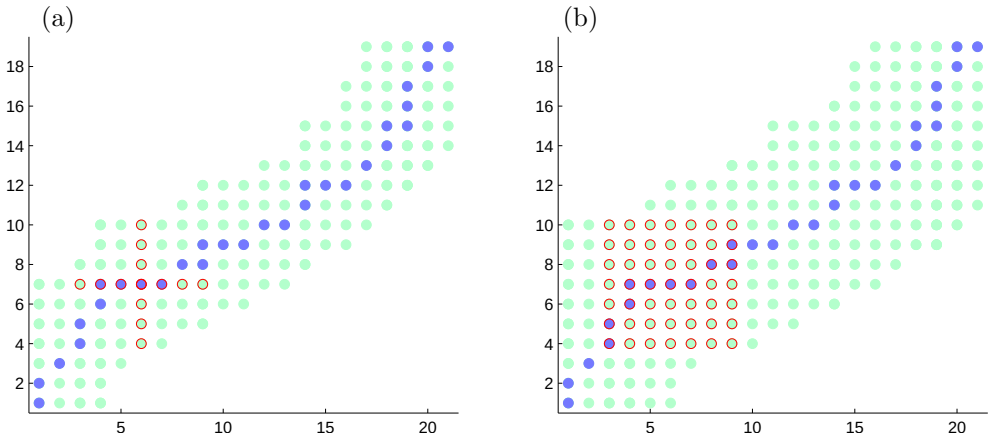


Abbildung 4.14: Tube-Bereichseinschränkung mit vorgegebenem Warping-Pfad (blau). Jeder farbige Eintrag ist Teil des Einschränkungsbereichs. Die rot umkreisten Punkte zeigen die Erzeugerstruktur über einem Zuordnungspunkt des Warping-Pfads. Erzeugerstruktur: (a) „Kreuz“ (Größe 7) (b) „Quadrat“ (Größe 7)

Definition 4.7 (Erzeugerstruktur „Kreuz“). *Die Erzeugerstruktur „Kreuz“* $E^{Kreuz} \in \{0,1\}^{d \times d}$ *der Größe* d *ist definiert über:*

$$E^{Kreuz}_{m,n} := \begin{cases} 1 & \textit{falls} \quad m = \lceil d/2 \rceil \vee n = \lceil d/2 \rceil \\ 0 & \textit{sonst} \end{cases}$$

Definition 4.8 (Erzeugerstruktur „Quadrat“). *Die Erzeugerstruktur „Quadrat“* $E^{Quadrat} \in \{0,1\}^{d \times d}$ *der Größe* d *ist definiert über:*

$$E^{Quadrat}_{m,n} := 1$$

In der MsDTW-Methode wurde eine der hier beschriebenen ähnliche Technik eingesetzt, wobei die Erzeugerstruktur „Kreuz“ gewählt wurde. Der Einschränkungsbereich wird über diese Erzeugerstruktur jedoch nur in Richtung der X und Y Zeitreihen erweitert, jedoch nicht in diagonaler Richtung. Dies führte in Experimenten unter Verwendung kleiner Erzeugerstrukturgrößen zu Problemen. Es zeigte sich, dass der zu berechnende Warping-Pfad oftmals auch in diagonaler Richtung von dem vorgegebenen abwich, was aufgrund der Form des Einschränkungsbereichs aber nicht möglich war. In Abbildung 4.15 wird das Problem visualisiert. Dieses Problem kann umgangen werden, indem die Größe der Erzeugerstruktur „Kreuz“ stark vergrößert wird, was sich aber negativ auf die Effizienz auswirkt. Für die Tube-Methode wird deshalb die Erzeugerstruktur „Quadrat“ verwendet, bei der solche Probleme nicht auftreten.

Zur Erklärung der benötigten Datenstrukturen sei ohne Beschränkung der Allgemeinheit angenommen, dass zwei Zeitreihen X und Y der Längen N und M aneinander ausgerichtet werden sollen. Bezogen auf eine Kostenmatrix soll X dabei vertikal und Y horizontal in einem kartesischen Koordinatensystem aufgetragen sein (siehe Abbildung 4.16).

Der Einschränkungsbereich unter Verwendung der Erzeugerstruktur „Quadrat“ kann nun über ein Paar von Vektoren $v^{\downarrow}, v^{\uparrow} \in \{1, \ldots, N\}^M$ spaltenweise definiert werden. Das Paar

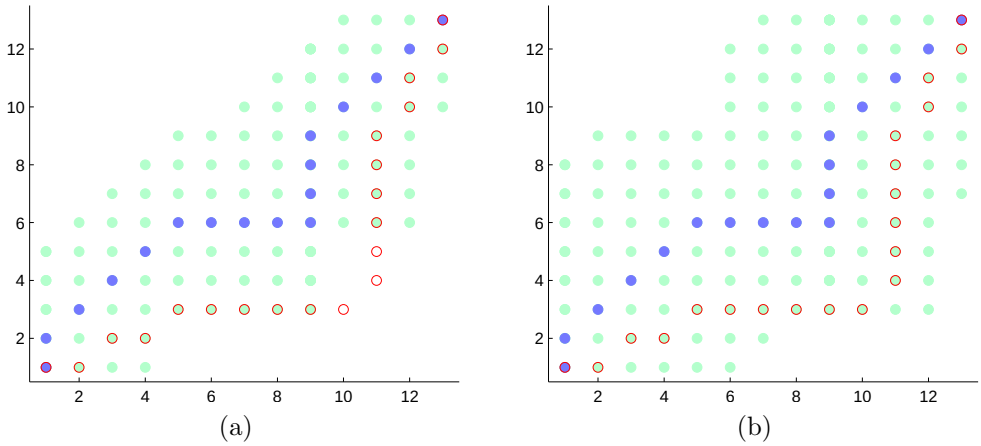


Abbildung 4.15: Problem mit Erzeugerstruktur „Kreuz“: In (a) müsste die Größe der Erzeugerstruktur „Kreuz“ wesentlich vergrößert werden, um den durch rote Kreise gekennzeichneten Warping-Pfad zu ermöglichen. In (b) wurde die Erzeugerstruktur „Quadrat“ verwendet, die auch in Diagonalrichtung erweitert und dieses Problem nicht aufweist.

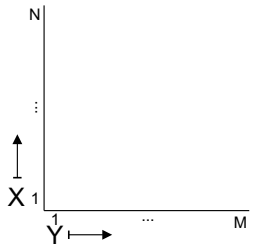


Abbildung 4.16: Zwei Zeitreihen werden in einem kartesischen Koordinatensystem aufgetragen.

$\left(v^{\downarrow}(m), v^{\uparrow}(m)\right)$ legt dabei die untere bzw. obere Grenze des Einschränkungsbereichs in Spalte m fest. Für den Einschränkungsbereich aus Abbildung 4.14(b) würde so zum Beispiel gelten: $\left(v^{\downarrow}(10), v^{\uparrow}(10)\right) = (4, 13)$. Aufgrund der Schrittweitenbedingung eines Warping-Pfads gilt folgender Satz, der zusichert, dass diese Datenstruktur ausreichend ist.

Satz 4.1. *Der von der Tube-Methode erzeugte Einschränkungsbereich unter Verwendung der Erzeugerstruktur $E^{Quadrat}$ und eines beliebigen vorgegebenen Warping-Pfads ist in jeder Spalte zusammenhängend, d.h.*

$$\forall m \in \{1, \ldots, M\} \quad \exists i, j \in \{1, \ldots, N\}, i < j \quad \forall n \in \{1, \ldots, N\} :$$

$$\mathcal{B}(n, m) = \begin{cases} 0 & \textit{falls} \quad n < i \\ 1 & \textit{falls} \quad i \leq n \leq j \\ 0 & \textit{falls} \quad n > j \end{cases}$$

Der klassische DTW-Algorithmus muss angepasst werden, damit ein Einschränkungsbereich in Form von $\left(v^{\downarrow}, v^{\uparrow}\right)$ verwendet werden kann. Man bestimmt dazu zunächst die maximale vertikale Ausdehnung der durch $\left(v^{\downarrow}, v^{\uparrow}\right)$ definierten Umgebung:

$$v_{max} := \max_{m=1,\ldots,M} (v^{\uparrow}(m) - v^{\downarrow}(m))$$

Für den Einschränkungsbereich aus Abbildung 4.14(b) gilt beispielsweise $v_{max} = 13$.

Anschließend verwendet man anstatt der akkumulierten Kostenmatrix $D \in \mathbb{R}^{N \times M}$ die Matrix $\tilde{D} \in \mathbb{R}^{v_{max} \times M}$. Der Wert $D(n, m)$ wird dann in $\tilde{D}(n - v^{\downarrow}(m) + 1, m)$ gespeichert. Auf diese Weise erwartet man für eine Erzeugerstruktur fester Größe d einen in M linearen Speicherbedarf, da v_{max} aber nur durch N beschränkt ist, kann dies nicht garantiert werden.

In Algorithmus 4.2 wird gezeigt, wie mit Hilfe von $\tilde{D}$ eine bereichsbeschränkte Version der akkumulierten Kostenmatrix D berechnet werden kann. Bis auf zusätzliche Prüfungen bezüglich des Einschränkungsbereichs verläuft die Rechnung wie im klassischen DTW-Algorithmus.

Algorithmus 4.2 : DTW mit Tube-Bereichseinschränkung

Eingabe : Zeitreihe X, Zeitreihe Y, $v^{\downarrow}$, $v^{\uparrow}$, Kostenmaß c
Ergebnis : WarpingPfad $\tilde{p}$

$\tilde{D}(1,1) := c(1,1)$
for $n = 2$ **to** $v^{\uparrow}(1)$ **do**
  $\tilde{D}(n,1) := \tilde{D}(n-1,1) + w_x \cdot c(n,1)$
end
for $m = 2$ **to** M **do**
  for $n = v^{\downarrow}(m)$ **to** $v^{\uparrow}(m)$ **do**
    $d_x := d_y := d_{xy} := \infty$
    if $n \neq v^{\downarrow}(m)$ **then** $d_x := \tilde{D}(n - v^{\downarrow}(m) + 1, m-1) + w_x \cdot c(n,m)$
    if $v^{\downarrow}(m-1) \leq n \leq v^{\uparrow}(m-1)$ **then** $d_y := \tilde{D}(n - v^{\downarrow}(m), m) + w_y \cdot c(n,m)$
    if $v^{\downarrow}(m-1) \leq n-1 \leq v^{\uparrow}(m-1)$ **then** $d_{xy} := \tilde{D}(n - v^{\downarrow}(m), m-1) + w_{xy} \cdot c(n,m)$
    $\tilde{D}(n - v^{\downarrow}(m) + 1, m) := \min(d_x, d_y, d_{xy})$
  end
end

Analog zum klassischen DTW kann über ein Backtracking über die Minimumsbildung in Algorithmus 4.2 ein optimaler Warping-Pfad berechnet werden. Dabei müssen wie im Algorithmus die Matrixkoordinaten umgesetzt werden, d.h. man startet das Backtracking ausgehend von $\tilde{D}(N - v^{\downarrow}(M) + 1, M) = D(N, M)$.

Zum Abschluss dieses Abschnitts dient wie unter Abschnitt 4.2.1 ein Ausschnitt aus Beethovens „kleine Appassionata“ als Beispiel.

4.2.3 Algorithmische Komplexität und Laufzeit

In [Mat06] wurde eine detaillierte Analyse der algorithmischen Komplexität und zu erwartenden Laufzeit der MsDTW-Methode vorgestellt, die an dieser Stelle nicht wiederholt werden soll. Stattdessen werden die relevanten Unterschiede beschrieben. Dazu sei im Folgenden angenommen, dass zwei Merkmalsfolgen X und Y der Längen N und M synchronisiert werden sollen. Für die MsDTW-Methode werden Chroma-Merkmale mit dem Kostenmaß $c_\alpha(c_l^X, c_k^Y) = \alpha - \langle c_l^X, c_k^Y \rangle$ verwendet. Zur Auswertung von c_α werden 12 Multiplikationen und Additionen benötigt. Unter Verwendung von CN-Merkmalen und dem Kostenmaß

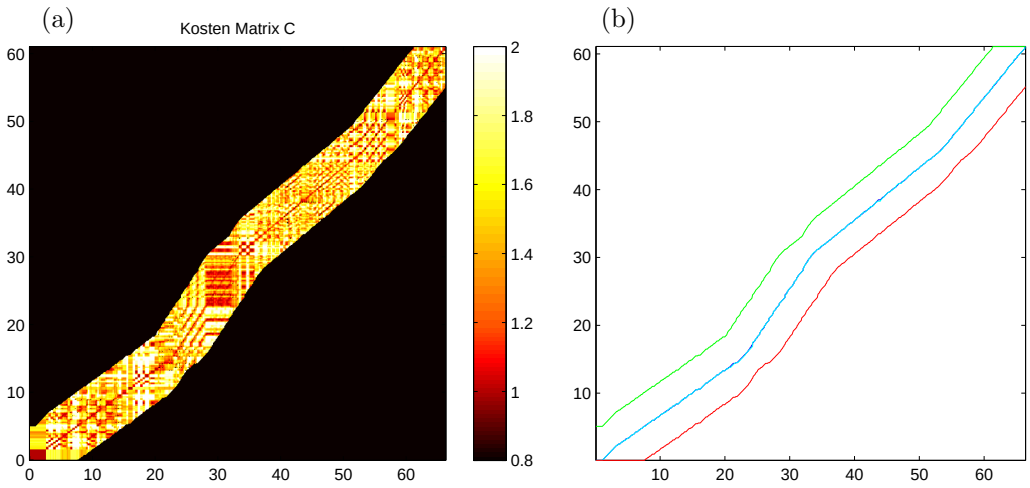


Abbildung 4.17: (a) Kostenmatrix unter Tube-Bereichseinschränkung. Schwarze Bereiche sind nicht Teil des Einschränkungsbereich. (b) Cyan: vorgegebener Warping-Pfad. Grün/Rot: Graph von $(m, v^{\uparrow}(m))$ bzw. $(m, v^{\downarrow}(m))$ für $m \in \{1, \ldots, M\}$.

$\mathbf{c}^{CN}_{\alpha,\beta}\big((c^X_l, n^X_l), (c^Y_k, n^Y_k)\big) = \alpha - (1 + n^X_l \cdot n^Y_k \cdot \beta)\langle c^X_l, c^Y_k\rangle$ werden drei Multiplikationen und eine Addition zusätzlich benötigt. Die Auswertung des Kostenmaßes

$$\mathbf{c}^{CNO}_{\alpha,\beta}\big((c^X_l, n^X_l, o^X_l), (c^Y_k, n^Y_k, o^Y_k)\big) = \alpha - (1 + n^X_l \cdot n^Y_k \cdot \left(\frac{\langle o^X_l, o^Y_k\rangle}{\|o^Y_k\|_2^2}\right)^2 \cdot \beta)\langle c^X_l, c^Y_k\rangle$$

benötigt formal durch den Zusatzterm $\left(\frac{\langle o^X_l, o^Y_k\rangle}{\|o^Y_k\|_2^2}\right)^2$ weitere 179 Multiplikationen und 174 Additionen. Praktisch wird dieser Term jedoch nur ausgewertet, wenn $n^X_l \cdot n^Y_k = 1$, weshalb die Gesamtlaufzeit unter Verwendung des Kostenmaßes $\mathbf{c}^{CNO}_{\alpha,\beta}$ nur geringfügig verlängert wird.

Alle verwendeten Kostenmaße sind unabhängig von N und M. Bezogen auf die algorithmische Komplexität gelten somit für die erweiterten Methoden aus diesem Kapitel unter Verwendung der MovingWindow- bzw. Tube-Bereichseinschränkung die gleichen Aussagen wie für die MsDTW-Methode. Dies bedeutet insbesondere, dass trotz effizienter Verfahren stets $O(N \cdot M)$ arithmetische Operationen zur DTW-Berechnung benötigt werden, da lediglich eine konstante Anzahl von Stufen im Multiskalenansatz verwendet wird.

Der Speicherbedarf im letzten Schritt des hier vorgestellten Multiskalen-Ansatzes unterscheidet sich jedoch zwischen der MovingWindow- und der Tube-Bereichseinschränkung und ist stark vom Verlauf des vorgegebenen Warping-Pfads abhängig. Die Bezeichner werden wie in den entsprechenden Abschnitten gewählt. Der günstigste Fall wird für den Fall eines diagonal verlaufenden vorgegebenen Warping-Pfads angenommen. In diesem Fall wird bei der MovingWindow-Bereichseinschränkung ein Speicherplatz in $O(d^2)$ benötigt und bei der Tube-Bereichseinschränkung in $O(d \cdot M)$. Verläuft der vorgegebene Warping-Pfad jedoch hauptsächlich vertikal bzw. horizontal, liegt der Speicherbedarf bei der MovingWindow-Bereichseinschränkung in $O(N \cdot d)$ und bei der Tube-Bereichseinschränkung in $O(N \cdot M)$. Bei fester Erzeugerstrukturgröße d garantiert die MovingWindow-Bereichseinschränkung somit im letzten Schritt einen in N linearen Speicherverbrauch.

Im nächsten Kapitel werden die vorgestellten Methoden um komplementäre Verfahren ergänzt. Dabei kann ausgehend von einem Warping-Pfad oftmals eine Erhöhung der Synchronisationsgenauigkeit durch den Einsatz verschiedener Methoden zur Nachverarbeitung erreicht werden.

4.2.4 Verwandte Arbeiten

In diesem Abschnitt wurden effiziente Methoden zur Musiksynchronisation vorgestellt. Das Grundprinzip zur Effizienzsteigerung bildete dabei der in Abschnitt 3.2 beschriebene allgemeine Multiskalenansatz für DTW. In anderen Arbeiten wurden weitere Strategien zur effizienten Berechnung von DTW vorgeschlagen. So wird in [DW05] eine Online-Version von DTW zur Musiksynchronisation eingesetzt. Der Einschränkungsbereich wird dabei über eine Greedy-Strategie während der Berechnung der akkumulierten Kostenmatrix bestimmt. Dazu werden in jedem Schritt die Werte der akkumulierten Kostenmatrix innerhalb eines so genannten *aktiven Gebiets* bestimmt. Anhand des kleinsten Wertes innerhalb des aktiven Gebiets wird im Anschluss geschätzt, wie ein optimaler Warping-Pfad wahrscheinlich verläuft.

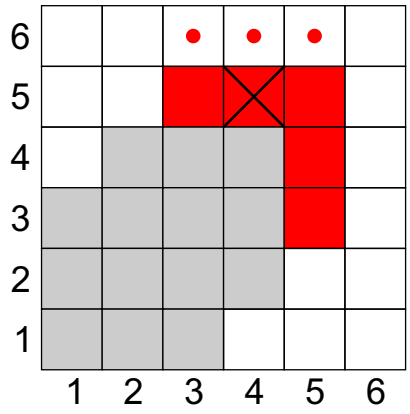


Abbildung 4.18: Definition eines DTW-Einschränkungsbereich bei Online-DTW

Abbildung 4.18 zeigt ein Beispiel. Alle farblich hinterlegten Matrixeinträge sind im aktuellen Schritt bereits Teil des Einschränkungsbereichs. Das aktive Gebiet entspricht der äußersten Reihe bzw. Spalte des aktuellen Einschränkungsbereichs. Die vertikale bzw. horizontale Ausdehnung des aktiven Gebiets kann über einen Parameter definiert werden (im Beispiel wurde dieser Parameter mit 3 belegt). Sind die Werte der akkumulierten Kostenmatrix im aktiven Gebiet berechnet, wird der Eintrag mit dem kleinsten Wert bestimmt. Im Beispiel ist dieser Eintrag mit einem Kreuz markiert. Man nimmt nun an, dass ein optimaler Warping-Pfad durch diesen Eintrag geringster Kosten verlaufen wird. In Abhängigkeit von der Position dieses Eintrags versucht man nun einzuschätzen, ob der Warping-Pfad an dieser Stelle horizontal, vertikal oder diagonal verlaufen wird und erweitert den Einschränkungsbereich entsprechend dieser Einschätzung. Im Beispiel liegt dieser Eintrag in der äußersten Zeile, weshalb der Einschränkungsbereich im nächsten Schritt um eine weitere Zeile ergänzt wird (markiert durch rote Punkte). Läge der Eintrag mit den geringsten Kosten in der äußersten

Spalte, so würde eine weitere Spalte zum Einschränkungsbereich hinzugefügt. Ist der komplette Einschränkungsbereich bestimmt, wird ein optimaler Warping-Pfad wie bekannt via Backtracking bestimmt.

Kapitel 5

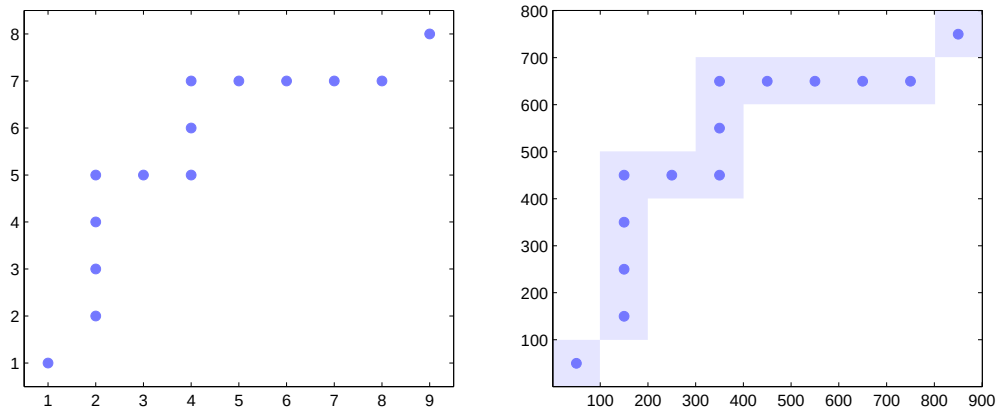
Ergänzende Methoden zur Erhöhung der zeitlichen Auflösung

In Kapitel 4 wurde die MsDTW-Synchronisationmethode untersucht. Über diese Methode werden sich entsprechende Positionen der zu synchronisierenden Varianten eines Musikstücks einander zugeordnet. Es wurde jedoch festgestellt, dass dabei in bestimmten Fällen unerwünschte Zuordnungen berechnet werden. Mit Hilfe zusätzlicher Informationen über Noteneinsatzzeiten wurden die bestehenden Methoden jedoch so erweitert, dass diese unerwünschten Zuordnungen in vielen Fällen vermieden werden können. Auf diese Weise werden sich entsprechende Positionen einander genauer zugeordnet und die zur Verfügung stehende Zeitauflösung wird effektiver genutzt. Die Zeitauflösung selbst wird dadurch jedoch nicht verändert.

In diesem Kapitel werden Strategien und Methoden vorgestellt, mittels derer die bei der Synchronisation verwendete Zeitauflösung erhöht werden kann. Ausgangspunkt ist dabei ein Warping-Pfad, wie er von den Methoden aus Kapitel 4 erzeugt wird. Durch Nachverarbeitung der durch den Warping-Pfad kodierten Zuordnungsinformationen kann die erzielte Zeitgenauigkeit erhöht werden.

5.1 Zeitliche Interpretation des Warping-Pfads

Eine automatische Annotation von Musikstücken ist mit aktuellen Methoden nur eingeschränkt möglich. Liegen jedoch bereits Noteninformationen zu einer Audioaufnahme vor (z.B. in Form einer MIDI-Datei), so können diese mit Hilfe von Synchronisationstechniken zur automatischen Annotation der Aufnahme verwendet werden. Dazu müssen die Noteneinsatzzeiten, die in MIDI-Daten gespeichert sind, so angepasst werden, dass sie möglichst genau den physikalischen Einsatzzeiten der Audioaufnahme entsprechen. Ausgangspunkt ist dabei ein Warping-Pfad, wie er von den Synchronisationsverfahren aus Kapitel 4 erzeugt wird. Anschaulich verwendet man einen Warping-Pfad, um bestimmten Positionen innerhalb einer Variante eines Musikstücks die entsprechenden Positionen innerhalb einer anderen Variante zuzuordnen. Ist also ein bestimmter Zeitpunkt in der ersten Variante vorgegeben (z.B. die Einsatzzeit einer MIDI-Note), so lässt sich anhand des Warping-Pfads die entsprechende Position innerhalb der zweiten Variante finden (die zugehörige physikalische Einsatzzeit in der Audioaufnahme). In diesem Abschnitt wird im Detail beschrieben, wie die Anpassung der MIDI-Einsatzzeiten bisher durchgeführt wurde, welche Probleme dabei auftreten und wie diese mit einem neuen Verfahren vermieden werden können.



(a) (b)

Abbildung 5.1: (a) Warping-Pfad **p** dargestellt in kartesischen Koordinaten (vertikal aufgetragen: X, horizontal: Y) (b) Warping-Pfad **p** wird als Zuordnung von Zeitbereichen interpretiert. Mit jedem Merkmal ist ein Zeitbereich von 100 ms assoziiert.

Aus der soeben formulierten Aufgabe, aus einem Warping-Pfad eine Zuordnung von Zeitpunkten abzulesen, ergeben sich einige Schwierigkeiten. Zunächst kodiert ein Warping-Pfad keine Zuordnung von Zeitpunkten, sondern eine Zuordnung von Elementen zweier Merkmalsfolgen. Werden bei der Berechnung eines Merkmals jedoch Informationen aus einem bestimmten Zeitbereich der zugrunde liegenden Daten verwendet, kann das Merkmal mit diesem Zeitbereich assoziiert werden. Infolgedessen kann ein Warping-Pfad auch als Zuordnung von Zeitbereichen interpretiert werden, was aber noch nicht der geforderten Zuordnung von Zeitpunkten entspricht. Im Verlauf dieses Abschnitts wird der Warping-Pfad **p** zwischen zwei Merkmalsfolgen $X = (x_1, \ldots, x_8)$ und $Y = (y_1, \ldots, y_9)$ als laufendes Beispiel verwendet.

$$\mathbf{p} = \big((1,1),(2,2),(3,2),(4,2),(5,2),(5,3),(5,4),(6,4),(7,4),(7,5),(7,6),(7,7),(7,8),(8,9)\big)$$

Abbildung 5.1(a) stellt den Warping-Pfad **p** grafisch dar. In Abbildung 5.1(b) wurde **p** als Zuordnung von Zeitbereichen interpretiert und entsprechend visualisiert. Dabei wurde angenommen, dass jedes Merkmal mit einem Zeitbereich von 100 ms assoziiert ist.

Für die im Folgenden beschriebenen Verfahren wird angenommen, dass ein Warping-Pfad zwischen den Merkmalsfolgen $X = (x_1, \ldots, x_N)$ und $Y = (y_1, \ldots, y_M)$ vorliegt. Für die Darstellung wird festgelegt, dass X auf der vertikalen Achse und Y auf der horizontalen Achse eines kartesischen Koordinatensystems aufgetragen wird. Ferner sei mit dem m-ten Merkmal einer Folge der Zeitbereich $[(m-1)\cdot T, m\cdot T)$ assoziiert, wobei $T \in \mathbb{R}$. Die Aufgabe besteht aus der Angabe einer Funktion, die anhand eines Warping-Pfads jedem Zeitpunkt bezüglich Y einen Zeitpunkt bezüglich X zuordnet. Genauer soll für einen Warping-Pfad p eine Funktion

$$f_p : [0, M\cdot T) \subset \mathbb{R} \to [0, N\cdot T) \subset \mathbb{R}$$

angegeben werden, die monoton steigend ist und folgende Eigenschaft (Pfadkompatibilität) erfüllt:

$$\forall\, m \in \{1,\ldots,M\} \quad \exists\, n \in \{1,\ldots,N\} :$$
$$t \in \big[(m-1)\cdot T, m\cdot T\big) \quad \Rightarrow \quad f(t) \in \big[(n-1)\cdot T, n\cdot T\big) \ \wedge\ (n,m) \in p$$

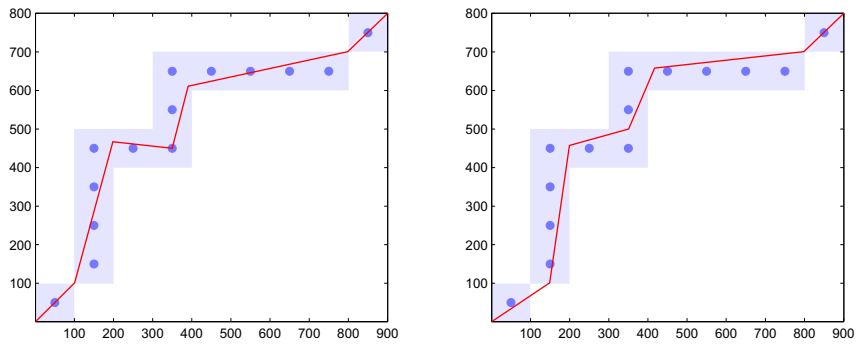


Abbildung 5.2: Ungültige Zeitzuordnungsfunktionen (rot) (a) Monotoniebedingung nicht erfüllt (b) Pfadkompatibilität nicht erfüllt.

Eine Funktion mit dieser Eigenschaft heißt *Zeitzuordnungsfunktion (für den Warping-Pfad p)*. Abbildung 5.2 zeigt zwei Beipspiele, bei denen diese Eigenschaften nicht erfüllt sind. In den nachfolgenden Unterabschnitten werden nun gültige Zeitzuordnungsfunktionen beschrieben.

5.1.1 Das Verfahren WarpTime 1

In bisherigen Methoden wurde ein Warping-Pfad nicht als Zuordnung von Zeitbereichen, sondern von Zeitpunkten interpretiert. Beispielsweise wurde die Zuordnung $(3, 2) \in \mathbf{p}$ so interpretiert, dass der Zeitpunkt 200 ms bezüglich der Merkmalsfolge Y dem Zeitpunkt 300 ms bezüglich X zugeordnet wird. Zeitpunkte zwischen diesen Punktzuordnungen wurden mittels einfacher Strategien, wie z.B. einer Treppenfunktion, abgebildet. Der WarpTime 1 Algorithmus (W_1) beschreibt ein Beispiel für eine solche Zeitzuordnungsfunktion. Abbildung 5.3 zeigt den Graph der WarpTime 1 Funktion für den Beispiel-Warping-Pfad $\mathbf{p}$.

Algorithmus 5.1 : WarpTime 1 (W_1)

Eingabe : Zeitpunkt t, WarpingPfad p zwischen den Merkmalsfolgen $X = (x_1, \ldots, x_N)$ und $Y = (y_1, \ldots, y_M)$, Länge T eines mit einem Merkmal assoziierten Zeitbereichs

Ergebnis : Zugeordneter Zeitpunkt $W_1(t, p, T)$

1. Bestimme $m \in \{1, \ldots, M\}$ mit: $(m-1) \cdot T \leq t < m \cdot T$
2. Bestimme $n_{max} = max(\{\, n \,|\, (n, m) \in p\})$
3. Ausgabe: $W_1(t, p, T) = \left(n_{max} - \frac{1}{2}\right) \cdot T$

Exemplarisch wird nun $W_1(t = 380, p = \mathbf{p}, T = 100)$ für den Beispiel-Warping-Pfad $\mathbf{p}$ angegeben.

1. Es gilt: $300 \leq t = 380 < 400$. Also ist $m = 4$.

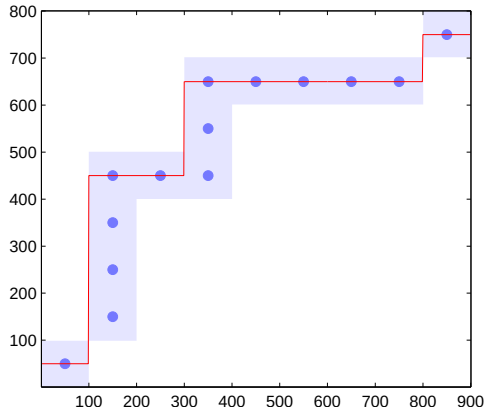


Abbildung 5.3: Graph der WarpTime 1 Zeitzuordnungsfunktion (rot) für den Warping-Pfad **p**. Die rote Kurve gibt die berechnete Zuordnung von Zeitpunkten an.

2. Merkmal 4 aus Y wird über **p** den Merkmalen 5, 6 und 7 zugeordnet. Also ist $n_{max} = 7$.
3. Ausgabe: $W_1(380, \mathbf{p}, 100) = \left(7 - \frac{1}{2}\right) \cdot 100 = 650$

Die recht einfache Vorgehensweise bei WarpTime 1 bedingt einige Eigenschaften der entstehenden Funktion.

1. Die von WarpTime 1 beschriebene Funktion ist eine gültige Zeitzuordnungsfunktion.
2. Die Funktion $W_1(\cdot, p, T)$ ist stückweise konstant. Viele Zeitpunkte t werden somit auf den selben Zeitpunkt $W_1(t, p, T)$ abgebildet. So gilt zum Beispiel $W_1(310, \mathbf{p}, 100) = W_1(380, \mathbf{p}, 100)$ für den Warping-Pfad **p**.
3. Sei $\tilde{p}$ ein Warping-Pfad zwischen den Merkmalsfolgen Y und X, der über

$$\tilde{p} := \{(m, n) | (n, m) \in p\}$$

definiert ist. $\tilde{p}$ legt somit dieselben Elementzuordnungen zwischen X und Y wie der Warping-Pfad p fest, kodiert diese jedoch in umgekehrter Reihenfolge. Wird $\tilde{p}$ im WarpTime 1 Algorithmus verwendet, werden die Rollen von X und Y im Algorithmus vertauscht. Die Funktion W_1 erfüllt folgende Eigenschaft im Allgemeinen aber nicht:

$$\forall\, t \in [0, M \cdot T)\ \ \forall\, \tilde{t} \in [0, N \cdot T) : \quad W_1(t, p, T) = \tilde{t} \ \Rightarrow\ W_1(\tilde{t}, \tilde{p}, T) = t$$

Somit ordnet WarpTime 1 unter Verwendung von $\tilde{p}$ nicht dieselben Zeitpunkte einander zu, wie unter Verwendung von p, obwohl beide Warping-Pfade die selben Elementzuordnungen beschreiben. Dies kann in einigen Anwendungsfällen zu Inkonsistenzen führen.

Aufgrund der zweiten Eigenschaft entstehen technische Probleme, wenn WarpTime 1 verwendet wird, um die Einsatzzeiten, die in einer MIDI-Datei gespeichert sind, an die physikalischen Einsatzzeiten eines Audiostücks anzugleichen. So kann es zu Inkonsistenzen in einer MIDI-Datei kommen, wenn zuvor unterschiedliche Zeitpunkte nach der Anpassung durch WarpTime 1 identisch sind. Eine MIDI-Datei kann dadurch unbrauchbar werden.

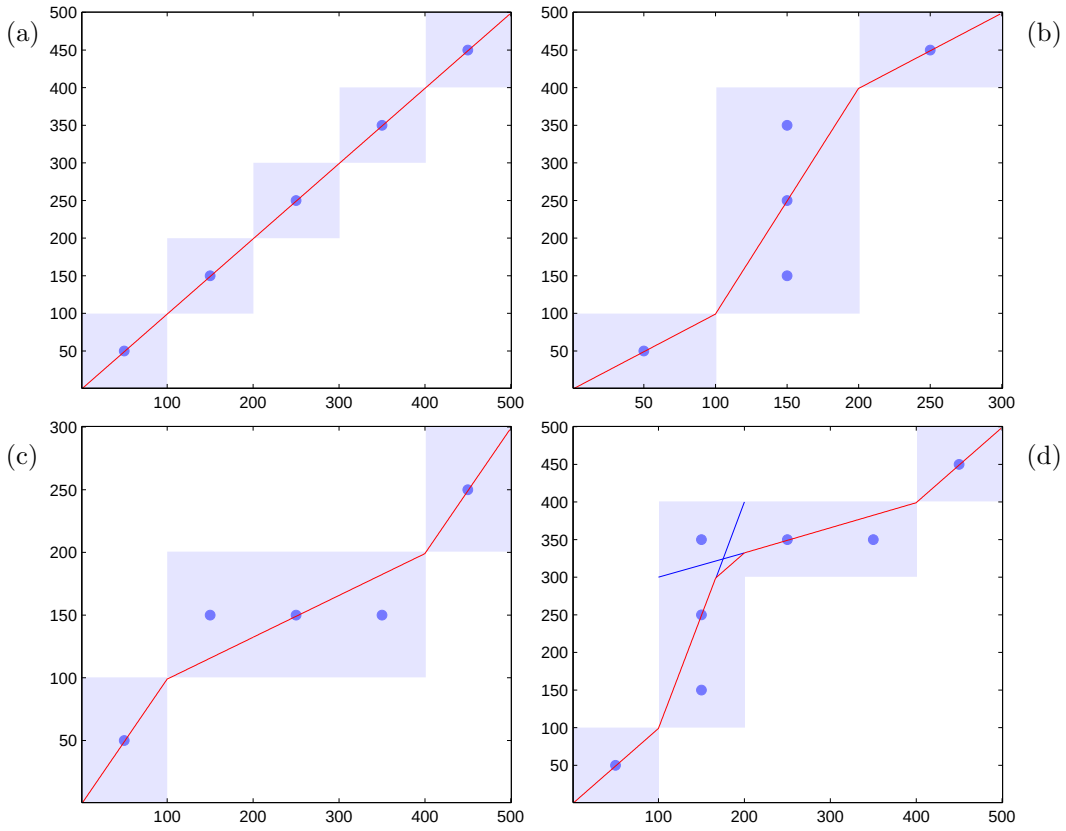


Abbildung 5.4: Erwünschtes Verhalten einer Zeitzuordnungsfunktion (rot) auf einigen exemplarisch ausgewählten Warping-Pfaden. Mit jedem Merkmal wird ein Zeitbereich von 100 ms assoziiert.

5.1.2 Das Verfahren WarpTime 2

Werden zwei Zeitbereiche über einen Warping-Pfad einander zugeordnet, stellt sich bei der Entwicklung einer alternativen Zeitzuordnungsfunktion zunächst die Frage, welche Zuordnung von Zeitpunkten innerhalb dieser Zeitbereiche denn wünschenswert ist. Im Folgenden wird dazu angenommen, dass es günstig ist, wenn eine Zeitzuordnungsfunktion zugeordnete Zeitbereiche linear durchläuft. Anhand einiger Beispiele soll verdeutlicht werden, was dies im Detail bedeutet. So soll die Funktion innerhalb eines Zeitbereichs linear verlaufen, falls der Warping-Pfad wie in Abbildung 5.4(a) diagonal verläuft. Verläuft der Warping-Pfad wie in Abbildung 5.4(b) bzw. 5.4(c) vertikal bzw. horizontal, so sollen mehrfach zugeordnete Zeitbereiche ebenfalls linear durchlaufen werden. Falls jedoch auf einen vertikal verlaufenden Abschnitt ein horizontal verlaufender folgt, kommt es zu einem Problem, das in Abbildung 5.4(d) dargestellt wird. Durch die Überschneidung der beiden Abschnitte kann die Zeit nicht in beiden Abschnitten durchgängig linear verlaufen. In Abbildung 5.4(d) wurde im Überschneidungsbereich zur Verdeutlichung in blau nachgezeichnet, wie sich ein linearer Verlauf der Zeit in beiden Abschnitten darstellen würde. Im Überschneidungsbereich wird deshalb ein linearer Übergang verwendet werden. Entsprechendes gilt für den Fall eines von horizontal auf vertikal wechselnden Verlaufs.

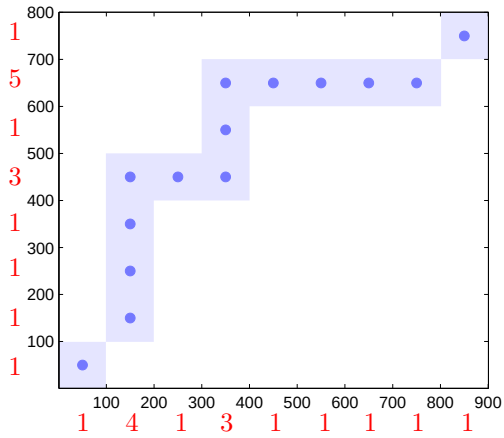


Abbildung 5.5: Zuordnungsgrad von Zeitbereichen bezüglich des Warping-Pfads **p** in roter Schrift. $\Psi_{\mathbf{p}}^{Y}$ (horizontal), $\Psi_{\mathbf{p}}^{X}$ (vertikal)

Nachfolgend wird eine Zeitzuordnungsfunktion formal definiert, die diesen anschaulich formulierten Anforderungen genügt. Dabei stellt es sich als vorteilhaft heraus, zunächst eine Funktion zu definieren, welche die Steigung der Zeitzuordnungsfunktion angibt. Über das bestimmte Integral dieser Funktion kann dann die Zeitzuordnungsfunktion selbst definiert werden. Zur Motivation betrachten wir Abbildung 5.4(b). Die Zeitzuordnungsfunktion verläuft im Bereich $[100, 200)$ mit der Steigung 3. Man erkennt einen Zusammenhang zwischen der Steigung und der Anzahl der Merkmale aus X, die dem zweiten Merkmal aus Y zugeordnet werden: Beide haben in diesem Fall den Wert 3. Die allgemeinen Zusammenhänge zwischen dem Warping-Pfad und der Steigung der Zeitzuordnungsfunktion werden in Algorithmus 5.2 weiter unten beschrieben.

Im weiteren Verlauf wird die Zeitzuordnungsfunktion von der *lokalen Verzerrungsfunktion* unterschieden, welche die Steigung der Zeitzuordnungsfunktion angibt. Die Bezeichnung soll verdeutlichen, dass die lokale Verzerrungsfunktion angibt, um welchen Faktor die Zeit lokal durch die Zeitzuordnungsfunktion gestaucht oder gestreckt wird. So wird die Zeit im Intervall $[100, 200)$ in Abbildung 5.4(b) um den Faktor 3 auf den Zeitbereich $[100, 400)$ gestreckt. Zur Definition einer geeigneten lokalen Verzerrungsfunktion wird der Begriff des Zuordnungsgrads benötigt.

Definition 5.1 (Zuordnungsgrad). *Es sei:*

- *p ein Warping-Pfad zwischen den Merkmalsfolgen X und Y*
- *Menge der Merkmale aus X, die durch p dem m-ten Merkmal aus Y zugeordnet werden:* $\{\, n \mid (n, m) \in p\}$

Dann ist der Zuordnungsgrad des m-ten Merkmals aus Y bezüglich des Warping-Pfads p über die Kardinalität von $\{\, n \mid (n, m) \in p\}$ *definiert:*

$$\Psi_p^Y(m) := |\{\, n \mid (n, m) \in p\}|$$

Entsprechend wird Ψ_p^X definiert. Da mit Merkmalen Zeitbereiche assoziiert sind, lässt sich auch ein Zuordnungsgrad dieser Zeitbereiche angeben. In Abbildung 5.5 ist exemplarisch der Zuordnungsgrad $\Psi_{\mathbf{p}}^Y$ horizontal und der Zuordnungsgrad $\Psi_{\mathbf{p}}^X$ vertikal für den Beispiel-Warping-Pfad $\mathbf{p}$ angegeben. Mit Hilfe des Zuordnungsgrads kann nun in Algorithmus 5.2 eine geeignete lokale Verzerrungsfunktion angegeben werden, welche die in Abbildung 5.4 anschaulich dargestellten Anforderungen erfüllt. Die beschriebene Funktion heißt *lokale Verzerrungsfunktion für lineare Interpolation* f. Abbildung 5.6 zeigt als Beispiel den Graph von $f(\cdot, \mathbf{p}, T = 100)$ für den Beispiel-Warping-Pfad $\mathbf{p}$.

Algorithmus 5.2 : Lokale Verzerrungsfunktion für lineare Interpolation: f

---

Eingabe : Zeitpunkt t, WarpingPfad p zwischen den Merkmalsfolgen $X = (x_1, \ldots, x_N)$ und $Y = (y_1, \ldots, y_M)$, Länge T eines mit einem Merkmal assoziierten Zeitbereichs

Ergebnis : Zugeordnete lokale Verzerrungsfunktion der Zeit $f(t, p, T)$

1. Für $t \notin [0, M \cdot T)$ definiere $f(t, p, T) = 1$ und gehe zu Ende.
2. Bestimme $m \in \{1, \ldots, M\}$ mit: $(m-1) \cdot T \leq t < m \cdot T$
3. Bestimme $\Psi_p^Y(m)$ und die Menge $\{\, n \,|\, (n, m) \in p\}$. Ordne die Menge aufsteigend: $(n_1, \ldots, n_{\Psi_p^Y(m)})$
4. Bestimme in welchem Unterzeitbereich t liegt: $k = \lfloor \frac{t-(m-1)*T}{T} \cdot \Psi_p^Y(m) \rfloor + 1$
5. Bestimme $\Psi_p^X(n_k)$
6. Ausgabe: $f(t, p, T) = \quad \Psi_p^Y(m) \;/\; \Psi_p^X(n_k)$

Ende

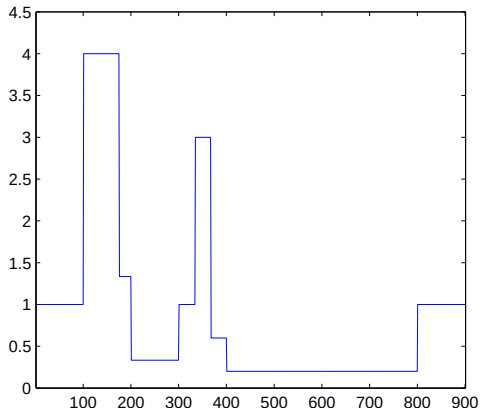


Abbildung 5.6: Lokale Verzerrungsfunktion für lineare Interpolation: Graph von f für den Warping-Pfad $\mathbf{p}$ mit $T = 100$

Exemplarisch wird nun $f(t = 380, p = \mathbf{p}, T = 100)$ für den Beispiel-Warping-Pfad $\mathbf{p}$ angegeben (vergleiche auch Abbildung 5.6).

1. Es gilt: $380 \in [0, 900)$. Somit kann mit Schritt 2 fortgefahren werden.

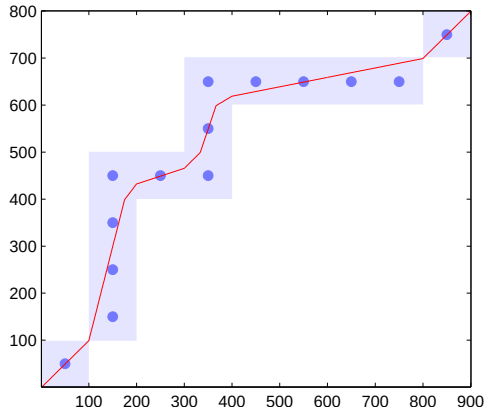


Abbildung 5.7: Graph der WarpTime 2 Zeitzuordnungsfunktion (rot) für den Warping-Pfad **p**. Die rote Kurve gibt die berechnete Zuordnung von Zeitpunkten an.

2. 380 fällt in den Zeitbereich $[300, 400)$. Also ist $m = 4$.
3. Über **p** wird das Merkmal $m = 4$ aus Y den Merkmalen $(n_1, \ldots, n_3) = (5, 6, 7)$ aus X zugeordnet und damit ist $\Psi_p^Y(4) = 3$ (vergleiche auch Abbildung 5.5).
4. Der Zeitbereich $[300, 400)$ wird in $\Psi_p^Y(4) = 3$ Unterzeitbereiche unterteilt:

   $T_1 = [300, 333)$, $T_2 = [333, 366)$, $T_3 = [366, 400)$. Das bedeutet 380 liegt in T_3. Betrachten wir, ob dieser Wert auch für k mit der Formel oben berechnet wird:

   $k = \lfloor \frac{t-(m-1)*T}{T} \cdot \Psi_p^Y(m) \rfloor + 1 = \lfloor \frac{380-(4-1)*100}{100} \cdot 3 \rfloor + 1 = 3.$
5. $\Psi_p^X(n_k) = \Psi_p^X(7) = 5$ (vergleiche auch Abbildung 5.5).
6. Ausgabe: $f(380, \mathbf{p}, 100) = 3/5$.

Im nächsten Schritt wird in Algorithmus 5.3 die Zeitzuordnungsfunktion WarpTime 2 (W_2) als das bestimmte Integral der lokalen Verzerrungsfunktion f festgelegt, die in Algorithmus 5.2 beschrieben wurde. Abbildung 5.7 zeigt den Graph von W_2 anhand des Beispiel-Warping-Pfads **p**. Im Anschluss wird in Satz 5.1 überprüft, welche Eigenschaften diese Funktion aufweist.

Algorithmus 5.3 : WarpTime 2 (W_2)

Eingabe : Zeitpunkt t, WarpingPfad p, Länge T eines mit einem Merkmal assoziierten Zeitbereichs

Ergebnis : Zugeordneter Zeitpunkt $W_2(t, p, T)$

$$W_2(t, p, T) = \int_0^t f(x, p, T)dx$$

Satz 5.1 (Eigenschaften von W_2). *Es sei:*

- *p ein beliebiger, aber fester Warping-Pfad zwischen den Merkmalsfolgen $X = (x_1, \ldots, x_N)$ und $Y = (y_1, \ldots, y_M)$*
- *T die Länge eines mit einem Merkmal assoziierten Zeitbereichs, d.h. mit dem m-ten Merkmal einer Folge ist der Zeitbereich $[(m-1) \cdot T, m \cdot T)$ assoziiert.*
- $\tilde{p} := \{\, (m,n) \,|\, (n,m) \in p\}$

Dann gilt:

1. $W_2(\cdot\,, p, T)$ *ist stückweise (affin-)linear*
2. $W_2(\cdot\,, p, T)$ *ist streng monoton steigend und stetig auf* $[0, M \cdot T]$
3. $W_2(0, p, T) = 0$ *und* $W_2(M \cdot T, p, T) = N \cdot T$
4. $W_2(\cdot\,, p, T)$ *ist eine bijektive Funktion von* $[0, M \cdot T] \to [0, N \cdot T]$
5. $W_2(\cdot\,, p, T)$ *erfüllt die Definition einer Zeitzuordnungsfunktion*
6. $\forall\, t \in [0, M \cdot T] :\ W_2\big(W_2(t, p, T), \tilde{p}, T\big) = t$

Beweis

zu 1.)
Es gilt: Die Stammfunktion einer stückweise konstanten Funktion ist stückweise (affin-)linear (Additivität des Integrationsintervals und Stammfunktion konstanter Funktionen). Es bleibt zu zeigen: $f(\cdot\,, p, T)$ ist stückweise konstant. Mit jedem Merkmal wird ein Zeitbereich assoziiert. Da die Länge der Merkmalsfolgen als endlich angenommen wird, gibt es nur endlich viele Zeitbereiche. Jeder dieser Zeitbereiche wird in Algorithmus 5.2 nochmals mit Hilfe des Zuordnungsgrads in Unterzeitbereiche eingeteilt. Da die Größe des Zuordnungsgrads jedoch ebenfalls durch die Länge der Zeitreihen beschränkt ist, kann jeder der endlich vielen Zeitbereiche nur in endlich viele Unterzeitbereiche eingeteilt werden. Betrachten wir weiter die Definition von f. In Schritt 4 wird t einem Unterzeitbereich T_i zugeordnet. Allen Zeitpunkten, die in diesem Unterzeitbereich liegen, wird derselbe Wert zugeordnet. Damit gibt es endlich viele Unterzeitbereiche, auf denen die lokale Verzerrungsfunktion f einen konstanten Wert annimmt.

zu 2.)
Betrachten wir die Definition von f. Aus der Definition des Zuordnungsgrads und eines Warping-Pfads folgt $f(\cdot\,, p, T) > 0$. Da f im Allgemeinen nicht stetig ist, kann hier nicht über den Hauptsatz der Analysis und den Zusammenhang zwischen Differentiation und Monotonie argumentiert werden. Stattdessen kann aber verwendet werden, dass positive riemannintegrierbare Funktionen eine streng monoton steigende Stammfunktion besitzen (siehe auch nächster Punkt bezüglich Integral von Treppenfunktionen). Die Stetigkeit von $W_2(\cdot\,, p, T)$ ist eine Eigenschaft des Integrals.

zu 3.)
$W_2(0, p, T) = 0$ folgt direkt aus der Definition des bestimmten Integrals.

zu $W_2(M \cdot T, p, T) = N \cdot T$:

Zu Punkt 1 wurde festgestellt, dass f eine stückweise konstante Funktion mit endlich vielen verschiedenen Werten ist. Es sei daran erinnert, dass das Integral einer solchen Treppenfunktion als Summe angegeben werden kann. Bezeichne dabei $\{\alpha_1, \ldots, \alpha_U\}$ die verschiedenen Werte, die eine Treppenfunktion f auf einem Integrationsgebiet Ω annimmt. Dann gilt:

$$\int_\Omega f(t)dt = \sum_{u=1}^{U} \alpha_u \cdot \mu\big(f^{-1}(\alpha_u) \cap \Omega\big) \quad ,$$

wobei μ das Lebesgue-Maß bezeichnet. Für die Funktion f gilt entsprechend:

$$\begin{aligned}
W_2(M \cdot T, p, T) &= \sum_{m=1}^{M} \sum_{(n,m)\in p} \frac{\Psi_p^Y(m)}{\Psi_p^X(n)} \left(\frac{1}{\Psi_p^Y(m)} \cdot T \right) \\
&= \sum_{(n,m)\in p} \frac{\Psi_p^Y(m)}{\Psi_p^X(n)} \left(\frac{1}{\Psi_p^Y(m)} \cdot T \right) \\
&= T \cdot \sum_{(n,m)\in p} \frac{1}{\Psi_p^X(n)} \\
&= T \cdot \sum_{n=1}^{N} \Psi_p^X(n) \cdot \frac{1}{\Psi_p^X(n)} \quad (*) \\
&= T \cdot N
\end{aligned}$$

$(*)$: jedes n kommt $\Psi_p^X(n)$-mal in einem Paar (n, m) in p vor (nach Definition von $\Psi_p^X(n)$).

zu 4.)
Folgt direkt aus 2 und 3.

zu 5.) und 6.)
Ohne Beweis.

□

Die problematischen Eigenschaften, die bei der Betrachtung der Zeitzuordnungsfunktion WarpTime 1 festgestellt wurden, gelten für WarpTime 2 nicht mehr. Genauer gelten für WarpTime 2 folgende Eigenschaften (vergleiche auch den entsprechenden Abschnitt zu Eigenschaften von WarpTime 1):

1. Die von WarpTime 2 beschriebene Funktion ist eine gültige Zeitzuordnungsfunktion.
2. Im Gegensatz zu WarpTime 1 ist WarpTime 2 nicht stückweise konstant. Die Zuordnung von Zeitpunkten ist bei WarpTime 2 aufgrund der Bijektivität eindeutig.
3. Mit $\tilde{p} := \{(m, n) | (n, m) \in p\}$ gilt für WarpTime 2 im Gegensatz zu WarpTime 1 folgende Eigenschaft:

$$\forall\, t \in [0, M \cdot T) \;\; \forall\, \tilde{t} \in [0, N \cdot T) : \quad W_1(t, p, T) = \tilde{t} \;\Rightarrow\; W_1(\tilde{t}, \tilde{p}, T) = t$$

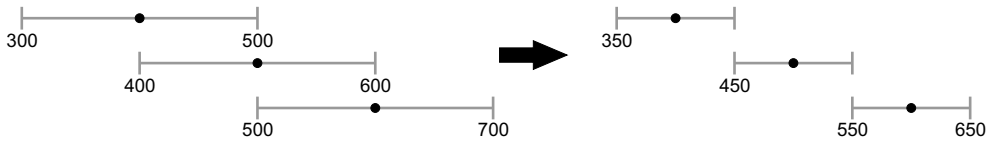


Abbildung 5.8: Links: Überlappende Zeitbereiche von Merkmalen. Rechts: Für WarpTime-Methoden angepasste Zeitbereiche. Die schwarzen Punkte kennzeichnen das Zentrum der Zeitbereiche.

Somit ordnet WarpTime 2 unter Verwendung von $\tilde{p}$ dieselben Zeitpunkte einander zu, wie unter Verwendung von p. Dies ist sinnvoll, da beide Warping-Pfade die selben Elementzuordnungen beschreiben und nur anders kodieren.

Um nun WarpTime 2 praktisch einsetzen zu können, muss zunächst überprüft werden, ob mit dem m-ten Merkmal einer Merkmalsfolge der Zeitbereich $[(m-1)\cdot T, m\cdot T)$ assoziiert ist. Daraus folgt auch, dass Zeitbereiche sich nicht überlappen dürfen. Unter Verwendung von Chroma-, CN- bzw. CNO-Merkmalen ergibt sich somit ein Problem, da diese aus überlappenden Zeitbereichen einer Audioaufnahme berechnet werden. Um die beschriebenen Methoden dennoch mit diesen Merkmalstypen einsetzen zu können, legt man sich auf eine Vereinfachung fest und assoziiert mit jedem Merkmal einen verkleinerten Zeitbereich. Dazu wird jeder Zeitbereich soweit um sein Zentrum gestaucht, bis keine Überlappung mehr vorliegt. Abbildung 5.8 stellt diesen Vorgang grafisch dar. Ob mit WarpTime 2 trotz dieser Vereinfachung befriedigende Resultate erzielt werden können, wird in Kapitel 6 untersucht.

5.1.3 Verwandte Arbeiten

In diesem Abschnitt wurden Methoden beschrieben, anhand derer eine Zuordnung von Elementen zweier Merkmalsfolgen, die ein Warping-Pfad beschreibt, zu einer Zuordnung von Zeitpunkten erweitert werden kann. Voraussetzung hierfür ist, dass mit jedem Merkmal ein Zeitbereich assoziiert werden kann.

In anderen Arbeiten wurden vergleichbare Techniken eingesetzt, wobei die Elemente der DTW-Zeitreihen jedoch nicht mit Zeitbereichen, sondern mit Zeitpunkten assoziiert sind. So werden in [KG03] Techniken beschrieben, um zwei Folgen von 3D-Bewegungsdaten sanft in einander überzuführen („Motion Blending“). Die Elemente einer solchen Folge entsprechen dabei Momentaufnahmen einer Bewegung, die über ein so genanntes Motion-Capturing-Verfahren ähnlich einem Film aufgenommen wurden. Dazu wird eine Person mit bestimmten Markierungen versehen, deren Position im Raum mittels spezieller Sensoren gemessen werden kann. Ein wichtiger Schritt der in [KG03] beschriebenen Techniken ist die Zuordnung der Elemente dieser beiden Folgen mit Hilfe von DTW. Da jedes dieser Elemente einer Momentaufnahme entspricht, ordnet der so berechnete Warping-Pfad einzelne Zeitpunkte einander zu. Um jedoch sanfte Übergänge zwischen solchen Folgen von 3D-Bewegungsdaten zu erzeugen, wird eine kontinuierliche Zuordnung von Zeitpunkten benötigt.

In bisherigen Verfahren wurden dazu die Zeitpunktzuordnungen linear interpoliert, die unter Verwendung von DTW berechnet wurden. In [KG03] werden Splines zur Interpolation eingesetzt. Durch die mit Splineinterpolation verbundene Glattheit der entstehenden Funktion

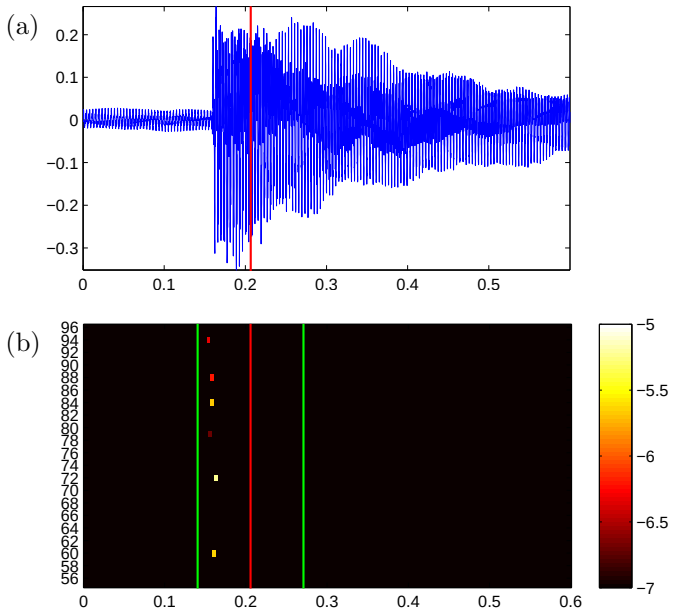


Abbildung 5.9: (a) Wellenform einer Aufnahme einer C4-Note. Mit einer roten Linie ist die Einsatzzeit einer C4-MIDI-Note markiert, die über die beschriebenen Verfahren grob an die physikalische Einsatzzeit der Aufnahme angepasst wurde. (b) Onset-Merkmale zu der Aufnahme. Zusätzlich ist in grün der Zeitbereich um die MIDI-Einsatzzeit markiert, der bei der Snapping-Methode nach Onset-Merkmalen durchsucht wird.

können sanftere Übergänge beim Motion Blending erzielt werden als bei linearer Interpolation. Ein Nachteil dieser Methode ist jedoch, dass nicht garantiert werden kann, dass die Zeitzuordnungsfunktion streng monoton steigend ist. Somit ist die Zuordnung von Zeitpunkten im Allgemeinen nicht eindeutig.

5.2 Nachverarbeitung mittels Onset-Merkmalen – Die Snapping-Methode

Zur Motivation der Snapping-Methode sei der Inhalt der bisherigen Abschnitte kurz zusammengefasst. In Kapitel 4 wurden Erweiterungen der MsDTW-Methode vorgestellt. Wie sich in Kapitel 6 zeigen wird, kann die Genauigkeit der Synchronisation mit diesen erweiterten Methoden gegenüber der MsDTW-Methode erhöht werden. Die Zeitauflösung der eingesetzten Merkmale und damit die theoretisch erreichbare Genauigkeit bleiben jedoch unverändert. In Abschnitt 5.1 wurde eine neuartige Zeitzuordnungsfunktion vorgestellt, über die eine Synchronisation zwischen einer Audioaufnahme und einer MIDI-Version zur automatischen Annotation der Audioaufnahme herangezogen werden kann. Dabei werden die Zuordnungsinformationen eines gegebenen Warping-Pfads durch die Zeitzuordnungsfunktion interpoliert. Die erwartete Genauigkeit der automatischen Annotation hängt somit von der Zeitauflösung der Merkmale ab, die bei der Synchronisation eingesetzt wurden. Unter den Standardparametern, die in Abschnitt 2.3 eingeführt wurden, liegt die Zeitauflösung der Chroma-, CN-

bzw. CNO-Merkmale bei etwa 100 ms. In diesem Abschnitt wird nun ein Verfahren beschrieben, mit dem die Genauigkeit einer automatischen Annotation durch Nachverarbeitung der MIDI-Einsatzzeiten erhöht werden kann. Die dabei eingesetzten Onset-Merkmale weisen eine Auflösung von 2.3 ms bis 11.9 ms auf.

Im Folgenden sei angenommen, dass eine Audioaufnahme und MIDI-Daten mit Hilfe der MsDTW-Methode oder der erweiterten Methoden aus Kapitel 4 synchronisiert und die MIDI-Einsatzzeiten unter Verwendung einer Zeitzuordnungsfunktion an die physikalischen Einsatzzeiten der Audioaufnahme angepasst wurden. Abbildung 5.9(a) zeigt dazu ein Beispiel. Dargestellt ist ein Ausschnitt der Wellenform einer Audioaufnahme, in der ein C4 angespielt wird. Die Einsatzzeit der an die Audioaufnahme angepassten MIDI-Note C4 wurde durch eine rote Linie markiert. Man erkennt anhand der Wellenform, dass die Einsatzzeit in der Audioaufnahme und die der MIDI-Note grob übereinstimmen. Daneben sind in Abbildung 5.9(b) zusätzlich die Onset-Merkmale zu der Wellenform dargestellt. Es lässt sich erkennen, dass die Onset-Ereignisse die Noteneinsatzzeit in der Audioaufnahme sehr genau wiedergegeben.

Bei der Snapping-Methode geht man davon aus, dass die angepassten MIDI-Einsatzzeiten und die physikalischen Einsatzzeiten grob übereinstimmen. Unter dieser Annahme wird für jede MIDI-Note ein Zeitbereich um ihre Einsatzzeit nach Onset-Ereignissen durchsucht. Wird ein Onset-Ereignis gefunden, dessen Tonhöhe mit der MIDI-Note übereinstimmt, so ersetzt man die Einsatzzeit der MIDI-Note mit der des Onset-Ereignisses. Werden mehrere passende Onset-Ereignisse in dem Zeitbereich gefunden, so wird das Onset-Ereignis verwendet, das am nächsten an der Einsatzzeit der MIDI-Note liegt. Falls kein passendes Onset-Merkmal gefunden wird, bleibt die MIDI-Note unverändert.

Dieser Vorgang wird in Abbildung 5.9(b) dargestellt. Für die MIDI-Note C4 (MIDI-Tonhöhe 60) wird im grün markierten Bereich nach Onset-Ereignissen gesucht. Bei 0.16 Sekunden wird ein passendes Onset-Ereignis mit Tonhöhe 60 gefunden und die Noteneinsatzzeit der MIDI-Note infolgedessen angepasst. Dieses „Einrasten“ an die Einsatzzeiten von Onset-Ereignissen wird im Folgenden als *Snapping-Methode* bezeichnet.

5.3 Weitere Ansätze und Ausblick

5.3.1 DTW basierend auf Onset-Merkmalen

Die Snapping-Methode erlaubt auf einfache Weise das „Nachjustieren“ der berechneten MIDI-Einsatzzeiten mit Hilfe hochaufgelöster Merkmale. Durch die dabei verwendete einfache Heuristik entstehen jedoch auch Nachteile. Man stelle sich vor, dass wie in Abschnitt 5.2 eine Audio-Aufnahme und MIDI-Daten miteinander synchronisiert wurden und die Einsatzzeiten der MIDI-Daten mittels einer Zeitzuordnungsfunktion angepasst wurden. Durch Anwendung der Snapping-Methode können nun Notengruppen, die zu einem Akkord gehören und gleichzeitig gespielt werden sollten, „auseinander gezogen“ werden.

Dies soll anhand eines Beispiels verdeutlicht werden. Abbildung 5.10(a) zeigt die Wellenform einer Aufnahme eines C4-F4 Akkords. Analog zu Abbildung 5.9 ist die Einsatzzeit der mittels Zeitzuordnungsfunktion angepassten MIDI-Noten C4 (MIDI-Tonhöhe 60) und F4 (MIDI-Tonhöhe 65) durch eine rote Linie markiert. Die Onset-Merkmale zu der Aufnahme sind in

Abbildung 5.10(b) dargestellt. Es fällt auf, dass der Einsatz der F4-Note nicht durch die Onset-Merkmale erkannt wurde. Ein möglicher Grund ist, dass die Hauptenergie nicht notwendigerweise in der Grundfrequenz liegt, sondern in den ersten Obertönen. Unter Umständen reicht die Energie in der Grundfrequenz deshalb nicht zur Erkennung durch Onset-Merkmale aus. Des Weiteren wurde in Abbildung 5.10(b) der Suchbereich um die Einsatzzeit der MIDI-Noten, welcher bei der Snapping Methode verwendet wird, mit Hilfe grüner Linien markiert. Im Suchbereich findet sich somit ein Onset-Ereignis für die MIDI-Note C4, aber kein Onset-Ereignis für die MIDI-Note F4. Aus diesem Grund wird die Einsatzzeit der C4-MIDI-Note angepasst, aber nicht die der F4-MIDI-Note. Die Noten des C4-F4 Akkords werden somit „auseinandergezogen“, obwohl die Noten sowohl in der Audioaufnahme als auch in den MIDI-Daten exakt gleichzeitig angespielt werden.

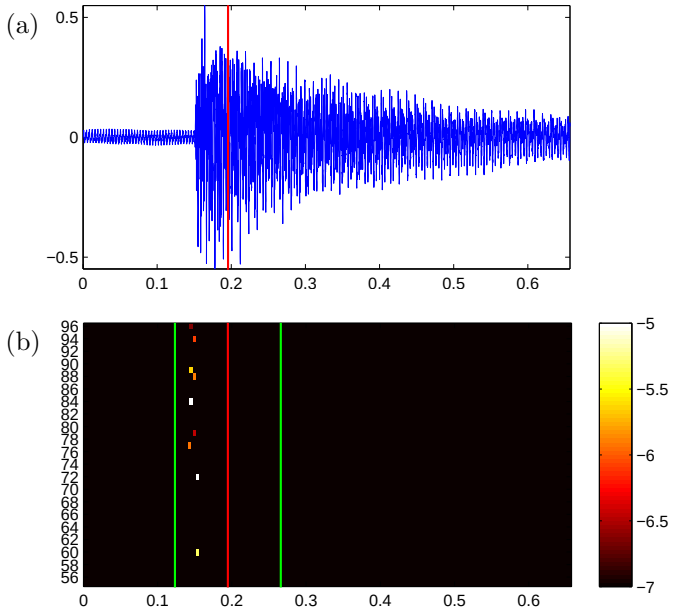


Abbildung 5.10: Die Abbildung ist analog zu Abbildung 5.9 aufgebaut. Gezeigt wird die Aufnahme eines C4-F4 Akkords.

In [MKR04] wurde eine DTW-Variante beschrieben, die in angepasster Form bei der Umgehung dieses Effekts hilfreich sein könnte. Dieses Verfahren geht auf die Besonderheiten ein, die unter Verwendung von DTW in Verbindung mit Onset-Merkmalen auftreten. Zu diesen Besonderheiten zählt die Definition eines Kostenmaßes. Da über Onset-Merkmale Noteneinsatzzeiten erkannt werden, kodieren sie Informationen zu einzelnen Zeitpunkten. Wird eine Aufnahme nun in äquidistante Zeitabschnitte unterteilt, werden in vielen Zeitbereichen keine Einsatzzeiten liegen. Um die Zeitabschnitte vergleichen und eine Synchronisation berechnen zu können, stellt es sich als günstig heraus, kein Kostenmaß zwischen den Zeitbereichen zu definieren, sondern ein Ähnlichkeitsmaß. Das DTW-Verfahren wird anschließend so abgewandelt, dass keine Kostenminimierung erreicht werden soll, sondern eine Ähnlichkeitsmaximierung. Auf diese Weise erreicht man, dass Zeitabschnitte der zu synchronisierenden Varianten nur dann verglichen werden müssen, wenn in beiden Zeitabschnitten eine Einsatzzeit erkannt wurde.

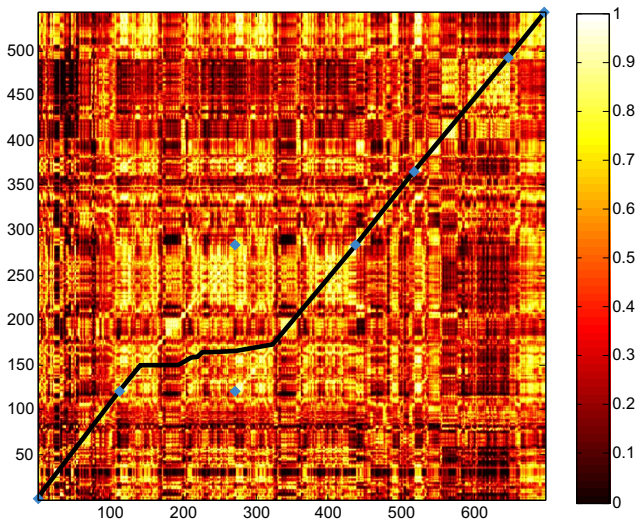


Abbildung 5.11: Kostenmatrix und optimaler Warping-Pfad. Berechnet über die MsDTW-Methode.

Praktisch zeigt sich dieses Verfahren jedoch häufig weniger robust gegenüber Klangfarbe und Instrumentierung als die MsDTW-Methode. Liegt jedoch eine grobe Synchronisation vor, die unter Verwendung einer robusten Methode bestimmt wurde, so ist über diese grobe Vorgabe die Definition eines DTW-Einschränkungsbereichs für die Methode aus [MKR04] möglich. Auf diese Weise sind Zeitauflösungen wie bei der Snapping-Methode zu erreichen, jedoch ohne die damit verbundenen Nachteile.

5.3.2 Partielle Synchronisation

Bisherige Verfahren zur Musiksynchronisation setzen voraus, dass sich zu synchronisierende Varianten eines Musikstücks bis auf interpretatorische Unterschiede in Dynamik, Klang, und Tempoverlauf im Wesentlichen entsprechen. Dazu zählen auch alle in dieser Arbeit vorgestellten Synchronisationsverfahren. Vergleicht man jedoch die Interpretationen verschiedener Künstler, so werden oftmals strukturelle Unterschiede deutlich. Zum Beispiel werden ganze Wiederholungen ausgelassen, zusätzliche Soli oder Kadenzen hinzugefügt, oder auch Strophen und Refrains abgewandelt oder umgestellt.

In einem zur Zeit noch unveröffentlichten Beitrag ([MA07]) wird ein neues Verfahren beschrieben, über das auch dann eine semantisch sinnvolle Synchronisation erzielt werden kann, falls solche strukturellen Abweichungen vorliegen. Dieses Verfahren wird nun in seinen Grundzügen anhand eines einfachen Beispiels umrissen. Dabei sollen zwei Varianten von Antonín Dvořáks Symphonie 9 „From the New World“ (op. 95) synchronisiert werden. Die beiden Varianten unterscheiden sich strukturell in ihrer Abfolge. Unterteilt man das Stück in musikalisch aussagekräftige Abschnitte, so werden in der ersten Variante die Abschnitte I (Einleitung), E (Exposition), D (Durchführung), R (Reprise) und C (Coda) gespielt. In der zweiten Variante finden sich die Abschnitte I, E, E, D, R und C. Die Exposition E wird somit wiederholt. In

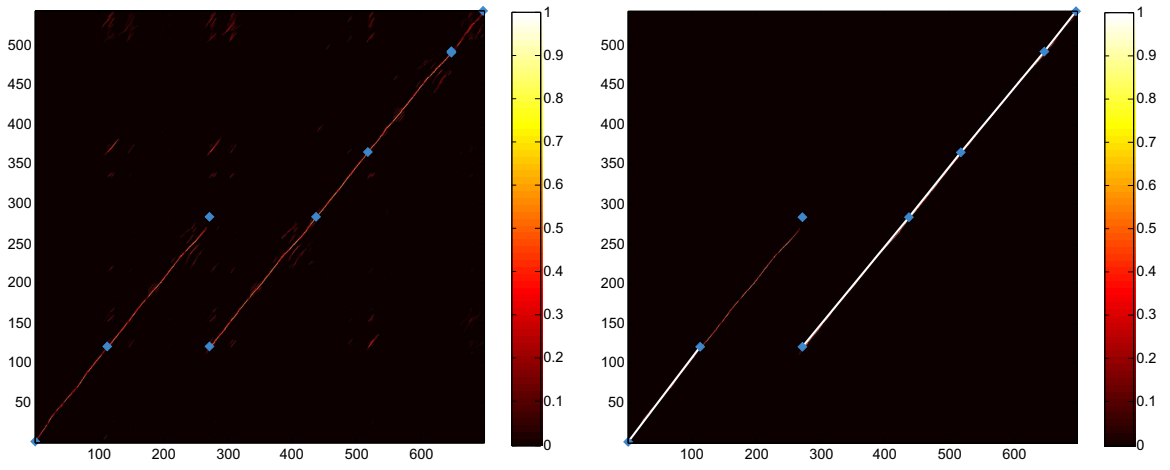


Abbildung 5.12: (a) Auf Pfade niedriger Kosten reduzierte Version der Kostenmatrix. (b) Kostenmatrix mit Warping-Pfad nach weiteren Optimierungsschritten.

Abbildung 5.11 wird dargestellt, welches Ergebnis die MsDTW-Methode in diesem Fall liefert, wobei die Kostenmatrix und ein optimaler Warping-Pfad dargestellt werden. Zusätzlich wurden manuell blaue Markierungen hinzugefügt, die Anfang und Ende der musikalisch aussagekräftigen Abschnitte kennzeichnen. Die erste Variante (I, E, D, R, C) ist dabei vertikal und die zweite Variante (I, E, E, D, R, C) horizontal aufgetragen. Man erkennt, dass der Warping-Pfad die Abschnitte I, D, R und C korrekt aufeinander abbildet. Da die Wiederholung des Abschnitts E in der zweiten Variante aber keine Entsprechung in der ersten Variante findet, kommt es in diesem Bereich zu sinnlosen Zuordnungen.

In [MA07] werden nun so genannte pfadbeschränkte Ähnlichkeitsmatrizen als Lösung vorgeschlagen. Dabei werden einige Techniken der Audiostrukturanalyse mit denen „klassischer“ Synchronisationsverfahren zusammengeführt. Bei der Audiostrukturanalyse untersucht man die Selbstähnlichkeit eines Stücks und versucht dabei, ähnlich zum DTW-Verfahren, bestimmte Pfade niedriger Kosten in einer Ähnlichkeitsmatrix zu identifizieren. Diese Techniken werden in [MA07] benutzt, um die Kostenmatrix, wie sie in Abbildung 5.11 zu sehen ist, auf bestimmte Pfade niedriger Kosten zu reduzieren. Abbildung 5.12(a) zeigt eine auf diese Weise pfadreduzierte Kostenmatrix. Man erkennt, dass auf diese Weise eine große Anzahl von teilweise sehr kurzen Pfaden berechnet wird. Mit Hilfe weiterer Optimierungsschritte reduziert man deren Anzahl weiter und konstruiert im Anschluss einen Pfad wie in Abbildung 5.12(b).

Jedes Segment des so konstruierten Pfads (im Beispiel oben erkennt man ein kürzeres und ein längeres Segment) kodiert ein Paar von sich entsprechenden Abschnitten. Aus Robustheitsgründen werden dabei jedoch zeitlich sehr grob aufgelöste Merkmale eingesetzt, weshalb die Zuordnungen innerhalb der Segmente relativ ungenau ausfallen. Zur Verfeinerung der Auflösung können nun aber klassische Synchronisationstechniken verwendet werden, um die Segmente jeweils einzeln mit höherer Auflösung „nachzusynchronisieren“.

Kapitel 6

Evaluation

In den vorangegangenen Kapiteln wurden neuartige Methoden und Strategien zur Synchronisation von Musik beschrieben. Die Qualität dieser Methoden kann jedoch anhand der einfachen Beispiele, die bei ihrer Beschreibung verwendet wurden, nur sehr eingeschränkt beurteilt werden. Aus diesem Grund werden die Verfahren aus Kapitel 4 und 5 in diesem Abschnitt auf einer größeren Testdatenbank evaluiert.

6.1 Automatische Evaluation von Synchronisationsmethoden

In diesem Abschnitt wird eine Methode zur automatischen Evaluation von Synchronisationsmethoden beschrieben. Dazu wäre es wünschenswert, die Genauigkeit einer Synchronisation angegeben zu können. Es stellt sich jedoch als äußerst schwierig heraus, diese Genauigkeit im Allgemeinen exakt zu definieren. Die nachfolgend beschriebene Evaluationsmethode beschränkt sich aus dieser Schwierigkeit heraus auf das Szenario einer Synchronisation von MIDI-Daten und Audioaufnahmen. In diesem Fall kann eine Synchronisation als automatische Annotation verstanden und ein optimales Ergebnis in Form einer manuell erstellten Annotation vorgegeben werden. Vergleicht man anschließend das über die Synchronisation berechnete Ergebnis mit dem vorgegebenen, kann die Abweichung quantitativ angegeben werden.

Genauer wird nun angenommen, dass Paare von MIDI-Daten und Audioaufnahmen vorliegen, wobei für jedes Paar die Einsatzzeiten der MIDI-Noten manuell an die physikalischen Einsatzzeiten der zugehörigen Audioaufnahme angepasst wurden. Für die vorliegende Arbeit wurden solche Paare unter anderem von einer musikalisch geschulten Person manuell erstellt. Da diese Arbeit jedoch sehr aufwendig ist, wurden weitere Paare gebildet, indem Audioaufnahmen mit Hilfe einer geeigneten Synthesesoftware aus MIDI-Daten erzeugt wurden.

Liegen solche Paare von MIDI-Daten und Audioaufnahmen vor, kann eine Synchronisationsmethode über folgendes Verfahren automatisch evaluiert werden:

1. **Verzerren der MIDI-Zeitinformationen**. Unter Verwendung einer geeigneten Methode werden die Einsatzzeit und Dauer der MIDI-Noten so verändert, dass sie nicht mehr mit der physikalischen Einsatzzeit und Dauer in der Audioaufnahme übereinstimmen. Diese Methode wird weiter unten separat beschrieben.
2. **Synchronisation der Audioaufnahmen und verzerrten MIDI-Daten**. Zur Synchronisation wird die MsDTW-Methode bzw. DTW mit CN- oder CNO-Merkmalen verwendet. Das Ergebnis ist ein Warping-Pfad.

3. **Anpassen der Noteneinsatzzeiten in den verzerrten MIDI-Daten an die physikalischen Einsatzzeiten der Audioaufnahme**. Dazu wird eine Zeitzuordnungsfunktion (WarpTime 1 bzw. WarpTime 2) in Verbindung mit dem Warping-Pfad eingesetzt, der in Schritt 2 berechnet wurde. Optional werden die angepassten Noteneinsatzzeiten mit Hilfe der Snapping-Methode nachbearbeitet.

4. **Vergleich der so berechneten MIDI-Daten mit den unverzerrten Orginal MIDI-Daten**. Details zum Vergleich finden sich weiter unten.

Abbildung 6.1 stellt den Ablauf einer automatischen Evaluation grafisch dar.

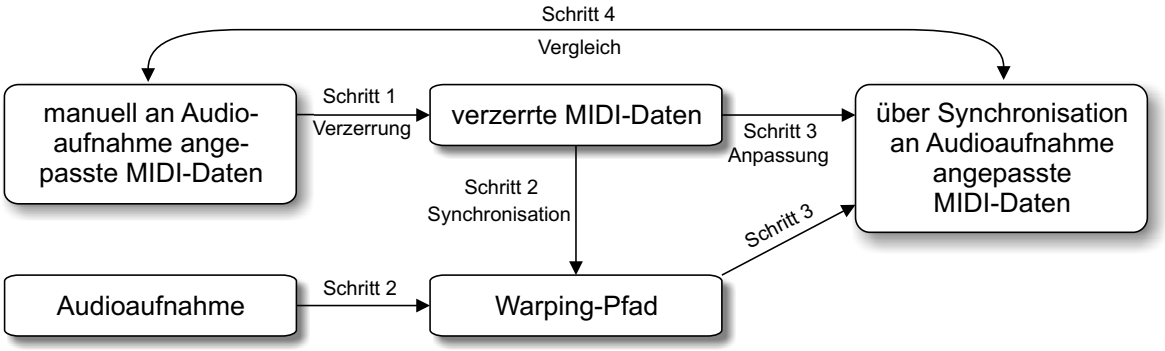


Abbildung 6.1: Ablauf einer automatischen Evaluation

Die in Schritt 1 verwendete Verzerrung der MIDI-Daten soll nun kurz beschrieben werden. Dazu müssen die Einsatzzeit und Dauer der MIDI-Noten geeignet verändert werden. Bezeichne dazu T die Länge des unverzerrten MIDI-Musikstücks in Millisekunden. Die Verzerrung wird dann über einen Vektor $v \in \mathbb{R}^L_{>0}$ und $L \in \mathbb{N}$ spezifiziert. Dazu wird T in L gleich große Zeitabschnitte unterteilt. Für $\ell \in \{1, \ldots, L\}$ wird die Zeit im ℓ-ten Zeitabschnitt mittels des multiplikativen Faktors $v(\ell)$ gestaucht bzw. gestreckt. Auf diese Weise kann die Einsatzzeit und Dauer jeder MIDI-Note verändert werden, worauf aber nicht im Detail eingegangen werden soll. Abbildung 6.2 skizziert ein Beispiel für den Vektor $v = (0.6, 0.8, 1.5) \in \mathbb{R}^3_{>0}$. L ist in diesem Beispiel somit 3.

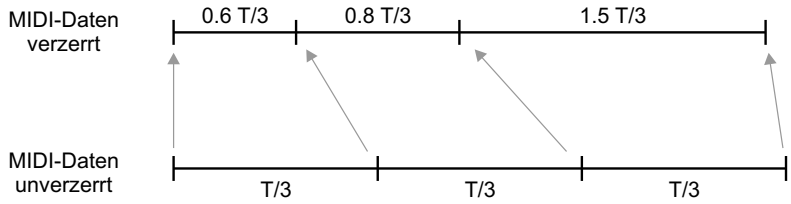


Abbildung 6.2: Zeitinformationen in MIDI-Daten der Länge T werden anhand des Vektors $v = (0.6, 0.8, 1.5)$ verzerrt.

Zum Vergleich der MIDI-Daten in Schritt 4 des automatischen Evaluationsverfahrens interpretiert man eine MIDI-Datei als eine Liste von Notenereignissen. Mit jedem Notenereignis ist dabei eine Einsatzzeit assoziiert. In Schritt 1 und Schritt 3 werden aus den vorgegebenen MIDI-Daten neue erzeugt, indem diese Einsatzzeiten verändert werden. Dabei ist wichtig, dass

die Reihenfolge der Notenereignisse innerhalb der Liste von dieser Veränderung nicht betroffen ist. Aus diesem Grund kann in Schritt 4 die unveränderte und die veränderte MIDI-Noten-Liste parallel sequentiell durchlaufen werden. Dabei kann für jede Note eine Abweichung der veränderten von der vorgegebenen Einsatzzeit berechnet werden.

Der Durchschnitt der Absolutwerte dieser Abweichungen in einem Stück wird im Folgenden mit $\varnothing_{abs}$ bezeichnet. Ein relativ großer Wert für $\varnothing_{abs}$ wird dahingehend interpretiert, dass die berechnete Synchronisation relativ ungenau ist. $\varnothing_{abs}$ wird im weiteren Verlauf auch *durchschnittliche Absolutabweichung* genannt. Des Weiteren werden die betragsmäßig größte positive bzw. negative Abweichung innerhalb eines Stücks mit $\mathbf{max}_{\text{pos}}$ bzw. $\mathbf{max}_{\text{neg}}$ bezeichnet.

6.2 Beschreibung der Testdatenbank

Die in den Experimenten verwendeten Testdaten bestehen aus 18 Paaren von MIDI-Daten und Audioaufnahmen. Bei der Auswahl der Stücke wurden verschiedene Komponisten der klassischen Musik berücksichtigt. Diese 18 Stücke sind Teil einer größeren Testdatenbank von über 300 Stücken, wobei sich die 18 Stücke, bezogen auf die Testergebnisse, als repräsentativ erwiesen haben. Alle Stücke sind reine Klavierlieder, bis auf ein Stück von Händel, bei dem Streicher zum Einsatz kommen und ein Stück von Mendelssohn, das neben einem Klavier auch eine Orchesterbegleitung enthält. Der Großteil der Audioaufnahmen wurde mit Hilfe der Synthesesoftware „Timidity“ ([IT04]) aus den MIDI-Daten erzeugt, was aufgrund eines Mangels an exakt annotierten Testdaten notwendig war. Die Verwendung synthetisch generierter Testdaten kann die Aussagekraft einer Evaluation jedoch mindern, da die Gefahr besteht, dass Messergebnisse nur von bestimmten Eigenschaften der Synthesemethode und nicht der Teststücke selbst abhängen. Dennoch lassen sich anhand synthetischer Daten oftmals Tendenzen erkennen, auch wenn sich Abweichungen der Messwerte unter Verwendung realer Daten ergeben können.

Unter den ausgewählten Teststücken befinden sich weiterhin zwei Stücke, die unter nichtsynthetischen Bedingungen aufgenommen wurden. Das erste ist ein Stück von Burgmüller, das von Meinard Müller eingespielt wurde. Das zweite ist ein Stück von Beethoven in einer Fassung von Daniel Barenboim. Beide wurden anhand einer Spektrogrammanalyse manuell annotiert.

Als Testdaten wurden desweiteren Stücke aus der von Masataka Goto initiierten RWC-Datenbank erwägt ([Got02]). Diese enthält Paare von Audioaufnahmen und MIDI-Daten. Zur Zeit dieses Schreibens sind die Einsatzzeiten in den MIDI-Daten jedoch nicht ausreichend genau an die Einsatzzeiten der Audioaufnahmen angepasst. So konnten lediglich ein paar Stücke des Unterbereichs RWC-Pop verwendet werden, der Popmusik enthält. Auf diese Weise konnten die Verfahren auf einem Musikgenre getestet werden, das bei der Entwicklung der Synchronisationsmethoden nicht im Vordergrund stand.

6.3 Experimente

Für die oben beschriebene automatische Evaluationsmethode müssen MIDI-Daten zeitlich verzerrt werden, wobei die Verzerrung anhand eines Vektors $v \in \mathbb{R}^L_{>0}$ spezifiziert wird. Für die nun beschriebenen Experimente wurde folgender Verzerrungsvektor verwendet, der unter Verwendung einer Zufallsfunktion erzeugt wurde:

$$v = (0.612, 1.197, 0.956, 1.345, 0.972, 0.934, 1.276, 1.020, 0.762, 1.137) \in \mathbb{R}^{10}_{>0}$$

Alle in dieser Arbeit beschriebenen Methoden wurden in der Programmierumgebung Matlab implementiert. In Abschnitt A findet sich eine Übersicht der verwendeten Funktionen. Die im Folgenden dargestellten Messwerte wurden über diese Implementation mit Matlab 2006b auf einem Rechner mit Intel Core 2 Duo E6600 Prozessor und 4GB Arbeitsspeicher unter Windows XP bestimmt.

6.3.1 Experiment 1 - Vergleich der MsDTW Methode mit den erweiterten Methoden aus Kapitel 4.1

Musikstück	**MsDTW**	**CN**	**CNO**
Burgmüller op02 no1 (Meinard Müller)	89	54	33
Beethoven op002 no1 1 Sonate 1 (Barenboim)	46	32	40
Bach BWV772 no1 Invention	36	39	29
Bach BWV787 no1 Sinfonia	52	50	34
Bach BWV988 no1 Gold Variationen	45	59	30
Beethoven op002 no1 1 Sonate 1	47	57	39
Beethoven op007 no1 Sonate 4	52	48	46
Beethoven op010 no1 1 Sonate 5	51	52	38
Burgmueller op100 no1	42	34	34
Chopin op10 no1 Etueden	76	113	71
Chopin op25 no1 Etueden	93	105	48
Haendel op6n07m1	137	113	96
Mendelssohn op25a	69	144	91
Mozart kv449m1 Pianosonate 14	42	45	40
Mozart kv37m1 Pianosonate 1	41	61	38
Schubert D911 no1 Winterreise	82	58	55
Schubert D899 1 Impromptus	54	70	43
Sor op1 no1	93	76	41
Durchschnitt über alle Teststücke	64	67	47

Tabelle 6.1: Vergleich der MsDTW-Methode, DTW mit CN-Merkmalen und DTW mit CNO-Merkmalen bezüglich $\varnothing_{abs}$.

In diesem Experiment wird die durchschnittliche Absolutabweichung $\varnothing_{abs}$ unter Verwendung von drei Synchronisationsmethoden miteinander verglichen.

1. MsDTW
2. DTW mit CN-Merkmalen

3. DTW mit CNO-Merkmalen

Die Parameter (α, β) der Kostenmaße $\mathbf{c}^{CN}_{\alpha,\beta}$ und $\mathbf{c}^{CNO}_{\alpha,\beta}$ wurden mit $(2, 1)$ belegt. Bei der Berechnung von DTW wurden keine effizienzsteigernden Methoden wie eine MovingWindow- bzw. Tube-Bereichseinschränkung verwendet. Des Weiteren wurde in allen drei Fällen WarpTime 2 in Schritt 3 der automatischen Evaluation verwendet. Die Snapping-Methode kam nicht zum Einsatz. Alle sonstigen Parameter der Methoden und Merkmale wurden mit den Standardparametern belegt, die in den entsprechenden Abschnitten festgelegt wurden. Tabelle 6.1 zeigt die Werte der durchschnittlichen Absolutabweichung $\varnothing_{abs}$ für alle drei Verfahren.

Man beobachtet, dass die Werte von $\varnothing_{abs}$ unter Verwendung von DTW mit CN-Merkmalen oftmals größer als unter Verwendung der MsDTW-Methode ausfallen. Verursacht wird dies durch degenerierte Warping-Pfade, wie sie in Abschnitt 4.1.3 beschrieben wurden. Auf großen Testdatenbeständen zeigte sich, dass beide Methoden ähnliche Werte für $\varnothing_{abs}$ liefern. Unter Verwendung von DTW mit CNO-Merkmalen erhält man jedoch deutlich kleinere Werte für $\varnothing_{abs}$. So ist der in Tabelle 6.1 angegebene „Durchschnitt über alle Teststücke“ unter Verwendung von DTW mit CNO-Merkmalen 27% kleiner als unter Verwendung der MsDTW-Methode. Des Weiteren fällt auf, dass die Werte von $\varnothing_{abs}$ kleiner als die Zeitauflösung der eingesetzen Merkmale sind (etwa 100 ms). Ursache dafür ist der Einsatz von WarpTime 2, das die Zuordnungsinformationen des berechneten Warping-Pfads interpoliert und dadurch die Genauigkeit erhöht.

6.3.2 Experiment 2 - Fortsetzung von Experiment 1

Musikstück	**MsDTW**		**CN**		**CNO**	
Burgmüller op02 no1 (Meinard Müller)	454	-50	215	-459	82	-160
Beethoven op002 no1 1 Sonate 1 (Barenboim)	386	-102	274	-89	274	-89
Bach BWV772 no1 Invention	173	-117	184	-198	87	-106
Bach BWV787 no1 Sinfonia	245	-197	260	-247	128	-147
Bach BWV988 no1 Gold Variationen	417	-217	417	-245	123	-132
Beethoven op002 no1 1 Sonate 1	242	-411	278	-691	208	-411
Beethoven op007 no1 Sonate 4	475	-413	1921	-1343	492	-472
Beethoven op010 no1 1 Sonate 5	305	-230	772	-464	201	-165
Burgmueller op100 no1	246	-99	102	-99	102	-99
Chopin op10 no1 Etueden	517	-773	1891	-1719	554	-623
Chopin op25 no1 Etueden	884	-669	2788	-1103	474	-264
Händel op6n07m1	326	-1445	326	-1589	126	-1589
Mendelssohn op25a	486	-569	1177	-1910	486	-1910
Mozart kv449m1 Pianosonate 14	536	-264	912	-292	415	-223
Mozart kv37m1 Pianosonate 1	405	-180	504	-373	230	-190
Schubert D911 no1 Winterreise	685	-502	685	-501	839	-501
Schubert D899 no1 Impromptus	430	-741	515	-641	430	-741
Sor op1 no1	449	-271	404	-231	273	-207
Durchschnitt über alle Teststücke	426	-403	757	-677	307	-446

Tabelle 6.2: Vergleich der MsDTW-Methode, DTW mit CN-Merkmalen und DTW mit CNO-Merkmalen bezüglich $\mathbf{max}_{\text{pos}}$ und $\mathbf{max}_{\text{neg}}$.

In Tabelle 6.2 werden weitere Ergebnisse von Experiment 1 dargestellt. Für jedes der drei Verfahren sind die Werte von $\mathbf{max}_{\text{pos}}$ in der linken Spalte und die Werte von $\mathbf{max}_{\text{neg}}$ in der rechten Spalte aufgetragen. Sowohl DTW mit CN-Merkmalen als auch mit CNO-Merkmalen liefern betragsmäßig für einige Stücke deutlich größere Werte als die MsDTW-Methode. Ursache sind degenerierte Warping-Pfade wie sie in Abschnitt 4.1 beschrieben wurden.

6.3.3 Experiment 3 - Einfluss des kostensenkenden Faktors β

Mit diesem Experiment wird der Einfluss des kostensenkenden Faktors β im Kostenmaß $\mathbf{c}_{\alpha,\beta}^{CNO}$ auf die durchschnittliche Absolutabweichung $\varnothing_{abs}$ untersucht. Dazu wurde wiederholt DTW mit CNO-Merkmalen berechnet, wobei jeweils der Parameter β variiert wurde. Alle sonstigen Parameter sind im Vergleich zu Experiment 1 unverändert. Tabelle 6.3 zeigt die auf diese Weise berechneten Werte von $\varnothing_{abs}$ in Abhängigkeit von β.

Musikstück	$\beta =$ 0.25	$\beta =$ 0.5	$\beta =$ 0.75	$\beta =$ 1	$\beta =$ 1.5	$\beta =$ 2
Burgmüller op02 no1 (Meinard Müller)	71	40	36	33	32	32
Beethoven op002 no1 1 Sonate 1 (Barenboim)	45	42	42	40	44	42
Bach BWV772 no1 Invention	29	30	30	29	31	31
Bach BWV787 no1 Sinfonia	34	32	31	34	34	36
Bach BWV988 no1 Gold Variationen	31	29	29	30	32	33
Beethoven op002 no1 1 Sonate 1	38	37	37	39	44	45
Beethoven op007 no1 Sonate 4	43	42	42	46	85	107
Beethoven op010 no1 1 Sonate 5	38	36	36	38	38	40
Burgmueller op100 no1	32	30	32	34	36	37
Chopin op10 no1 Etueden	65	63	67	71	98	119
Chopin op25 no1 Etueden	46	41	42	48	132	248
Händel op6n07m1	113	106	96	96	98	97
Mendelssohn op25a	60	59	66	91	119	141
Mozart kv449m1 Pianosonate 14	36	36	39	40	52	57
Mozart kv37m1 Pianosonate 1	35	34	36	38	42	46
Schubert D911 no1 Winterreise	65	59	56	55	53	50
Schubert D899 no1 Impromptus	41	41	42	43	45	204
Sor op1 no1	54	47	43	41	41	40
Durchschnitt über alle Teststücke	49	45	45	47	59	78

Tabelle 6.3: Vergleich von DTW mit CNO-Merkmalen anhand der durchschnittlichen Absolutabweichung $\varnothing_{abs}$ unter Verwendung verschiedener Werte für den Parameter β des lokalen Kostenmaßes $\mathbf{c}_{\alpha,\beta}^{CNO}$.

Es zeigt sich, dass DTW mit CNO-Merkmalen mit $\beta \in [0.5, 1]$ relativ kleine Werte für $\varnothing_{abs}$ liefert. Kleinere Werte für β bewirken, dass die Kostensenkung zu wenig Einfluss erhält. Zu große Werte für β bewirken, dass degenerierte Warping-Pfade, wie unter Abschnitt 4.1 beschrieben, zu sehr begünstigt werden.

6.3.4 Experiment 4 - Einfluss der Tube-Bereichseinschränkung

In diesem Experiment wird untersucht, welche Auswirkung die Tube-Bereichseinschränkung auf die Laufzeit von DTW und den Verlauf des berechneten Warping-Pfads hat. Da die Laufzeit bei klassischem DTW von der Länge der beiden zu synchronisierenden Varianten abhängt, wurde die Länge der Audioaufnahme in der zweiten Spalte in Sekunden angegeben. Die Länge der MIDI-Daten entspricht trotz der Verzerrung in Schritt 1 des automatischen Evaluationsverfahrens in etwa der Länge der Audioaufnahme und wird nicht eigens angegeben. In den folgenden Spalten ist die Laufzeit von DTW mit CNO-Merkmalen in Millisekunden angegeben. Dabei wurde für die dritte Spalte keine DTW-Bereichseinschränkung verwendet. In der vierten bis sechsten Spalte wurde eine Tube-Bereichseinschränkung eingesetzt, wobei verschiedene Größen der Erzeugerstruktur „Quadrat“ verwendet wurden. Die Erzeugerstrukturgröße ist dabei wie in Abschnitt 4.2.2 definiert angegeben (Anzahl Merkmale). Weiterhin stellte sich die Frage, ob und wie weit die Laufzeit durch Optimierung der Implementierung verkürzt werden kann. Zum Vergleich mit den Matlab Implementierungen wird deshalb in der Spalte „DLL“ angegeben, welche Laufzeit in Millisekunden eine in der Programmiersprache C geschriebene klassische DTW-Implementation benötigte. Alle Ergebnisse wurden auf einem Rechner mit Intel Core 2 Duo E6600 Prozessor mit 4GB Arbeitsspeicher unter Windows XP mit Matlab R2006b berechnet.

Musikstück	**Länge**	–	**300**	**20**	**10**	**DLL**
Burgmüller op02 no1 (Meinard Müller)	18	298	307	56	48	5
Beethoven op002 no1 1 Sonate 1 (Barenboim)	21	416	393	64	57	6
Bach BWV772 no1 Invention	68	4401	1894	219	211	66
Bach BWV787 no1 Sinfonia	74	5140	2019	241	221	80
Bach BWV988 no1 Gold Variationen	109	12393	3350	416	350	166
Beethoven op002 no1 1 Sonate 1	213	44338	6632	749	748	612
Beethoven op007 no1 Sonate 4	489	234802	16309	2061	2220	3242
Beethoven op010 no1 1 Sonate 5	309	92358	9862	1045	821	1292
Burgmueller op100 no1	78	5598	2128	240	156	89
Chopin op10 no1 Etueden	117	15089	3924	429	282	189
Chopin op25 no1 Etueden	143	21696	4724	509	341	276
Händel op6n07m1	60	3094	1471	176	121	49
Mendelssohn op25a	365	135190	12284	1439	1022	1817
Mozart kv449m1 Pianosonate 14	424	193322	14092	1617	1253	2485
Mozart kv37m1 Pianosonate 1	277	80242	9226	973	725	1073
Schubert D911 no1 Winterreise	330	94886	9427	1062	932	1485
Schubert D899 no1 Impromptus	445	191713	14145	1734	1389	2741
Sor op1 no1	76	5175	1962	231	150	81

Tabelle 6.4: Vergleich der Laufzeit von DTW mit Tube-Bereichseinschränkung.

Man erkennt die quadratische Abhängigkeit zwischen der Länge des Stücks in Spalte 2 und der Laufzeit in Spalte 3. Ein Vergleich der dritten Spalte mit der Spalte „DLL“ zeigt, dass die Matlab Implementierung deutlich ineffizienter bezüglich der Laufzeit arbeitet. Anhand der Spalten „300“, „20“ und „10“ ist ersichtlich, dass die Laufzeit unter Verwendung einer Tube-Bereichseinschränkung deutlich verkürzt werden kann.

Neben der Laufzeit wurde in diesem Experiment der Einfluss der Tube-Bereichseinschränkung

auf den Verlauf des berechneten Warping-Pfads betrachtet. Es zeigte sich für alle Stücke, dass unter Verwendung von Erzeugerstrukturen, die größer als 10 waren, dieselben Warping-Pfade berechnet wurden, wie im uneingeschränkten Fall. Nun sei daran erinnert, dass der Einschränkungsbereich bei der Tube-Methode anhand des Warping-Pfads festgelegt wird, der von der MsDTW-Methode berechnet wird. Dass eine Erzeugerstrukturgröße größer 10 ausreicht, lässt erkennen, dass der Warping-Pfad, der über DTW mit CNO-Merkmalen berechnet wird, relativ genau dem Verlauf des MsDTW-Warping-Pfads folgt. Anders formuliert bestätigt sich damit experimentell die Intuition, dass die Methoden aus Abschnitt 4.1 lediglich kleinere Korrekturen am Verlauf des Warping-Pfads bedingen, der über die MsDTW-Methode berechnet wird.

6.3.5 Experiment 5 - Vergleich der Zeitzuordnungsfunktionen WarpTime 1 und WarpTime 2

In allen bisherigen Experimenten wurde WarpTime 2 als Zeitzuordnungsfunktion in Schritt 3 der automatischen Evaluation eingesetzt. Mit diesem Experiment wird betrachtet, welchen Einfluss die Zeitzuordnungsfunktion auf die durchschnittliche Absolutabweichung $\varnothing_{abs}$ hat. Dazu wurde DTW mit CNO-Merkmalen zweimal über die automatische Evaluation ausgewertet, wobei die Zeitzuordnungsfunktion in Schritt 3 zwischen WarpTime 1 und WarpTime 2 variiert wurde. Der Parameter β des Kostenmaßes $\mathbf{c}_{\alpha,\beta}^{CNO}$ wurde mit 0.75 belegt. Alle sonstigen Parameter sind identisch mit denen aus Experiment 1. Tabelle 6.5 zeigt die auf diese Weise berechneten Werte von $\varnothing_{abs}$ in Abhängigkeit von der verwendeten Zeitzuordnungsfunktion.

Musikstück	**WarpTime 1**	**WarpTime 2**
Burgmüller op02 no1 (Meinard Müller)	50	36
Beethoven op002 no1 1 Sonate 1 (Barenboim)	54	42
Bach BWV772 no1 Invention	59	30
Bach BWV787 no1 Sinfonia	60	31
Bach BWV988 no1 Gold Variationen	59	29
Beethoven op002 no1 1 Sonate 1	69	38
Beethoven op007 no1 Sonate 4	73	42
Beethoven op010 no1 1 Sonate 5	63	37
Burgmueller op100 no1	63	32
Chopin op10 no1 Etueden	111	67
Chopin op25 no1 Etueden	83	43
Händel op6n07m1	141	96
Mendelssohn op25a	106	64
Mozart kv449m1 Pianosonate 14	70	39
Mozart kv37m1 Pianosonate 1	68	36
Schubert D911 no1 Winterreise	72	56
Schubert D899 no1 Impromptus	67	42
Sor op1 no1	64	43
Durchschnitt über alle Teststücke	74	45

Tabelle 6.5: Vergleich der Zeitzuordnungsfunktionen WarpTime 1 und WarpTime 2 anhand der durchschnittlichen Absolutabweichung $\varnothing_{abs}$.

Es zeigt sich, dass die Werte von $\varnothing_{abs}$ unter Verwendung von WarpTime 1 deutlich größer als unter WarpTime 2 ausfallen. Bezogen auf den in Tabelle 6.5 angegebenen „Durchschnitt

über alle Teststücke“ ist der Wert von $\varnothing_{abs}$ 39% kleiner unter Verwendung von WarpTime 2, als unter Verwendung von WarpTime 1.

6.3.6 Experiment 6 - Einfluss der Snapping-Methode auf die Synchronisationsgenauigkeit

In Schritt 3 der automatischen Evaluation werden die Noteneinsatzzeiten der MIDI-Daten mittels einer Zeitzuordnungsfunktion an die physikalischen Einsatzzeiten der Audioaufnahme angepasst. Bei den bisherigen Experimenten wurde dabei auf eine Nachverarbeitung verzichtet. In diesem Experiment wird untersucht, welchen Einfluss die Snapping-Methode aus Abschnitt 5.2 auf die Synchronisationsgenauigkeit hat. Dazu wurden die Werte von $\varnothing_{abs}$ einmal mit und einmal ohne Snapping-Methode berechnet, wobei jeweils DTW mit CNO-Merkmalen in Schritt 2 der automatischen Evaluation verwendet wurde. Der Parameter β des Kostenmaßes $\mathbf{c}_{\alpha,\beta}^{CNO}$ wurde mit 0.75 belegt. Alle sonstigen Parameter sind im Vergleich zu Experiment 1 unverändert. Tabelle 6.6 zeigt in der zweiten Spalte die Werte von $\varnothing_{abs}$ ohne und in der dritten Spalte mit nachgeschalteter Snapping-Methode.

Musikstück	**CNO**	**CNO + Snapping**
Burgmüller op02 no1 (Meinard Müller)	36	28
Beethoven op002 no1 1 Sonate 1 (Barenboim)	42	34
Bach BWV772 no1 Invention	30	17
Bach BWV787 no1 Sinfonia	31	19
Bach BWV988 no1 Gold Variationen	29	20
Beethoven op002 no1 1 Sonate 1	38	27
Beethoven op007 no1 Sonate 4	42	32
Beethoven op010 no1 1 Sonate 5	37	24
Burgmueller op100 no1	32	12
Chopin op10 no1 Etueden	67	58
Chopin op25 no1 Etueden	43	32
Händel op6n07m1	96	103
Mendelssohn op25a	64	61
Mozart kv449m1 Pianosonate 14	39	27
Mozart kv37m1 Pianosonate 1	36	22
Schubert D911 no1 Winterreise	56	49
Schubert D899 no1 Impromptus	42	36
Sor op1 no1	43	33
Durchschnitt über alle Teststücke	45	35

Tabelle 6.6: Vergleich von DTW mit CNO-Merkmalen mit und ohne Nachverarbeitung durch die Snapping-Methode anhand der durchschnittlichen Absolutabweichung $\varnothing_{abs}$.

Wie man erkennt, bewirkt die Nachverarbeitung der Noteneinsatzzeiten über die Snapping-Methode eine Verkleinerung der durchschnittlichen Absolutabweichung $\varnothing_{abs}$. Bezogen auf den „Durchschnitt über alle Teststücke“ zeigt sich, dass der Wert von $\varnothing_{abs}$ unter Verwendung der Snapping-Methode etwa 22% kleiner ist.

6.3.7 Experiment 7 - Einfluss des Musikgenres

Musikstück	MsDTW	CN	CNO
RM-P030	927	970	932
RM-P031	756	830	756
RM-P032	742	763	732
RM-P033	241	237	210
RM-P034	228	224	222
RM-P035	117	119	104
RM-P036	119	173	119
RM-P037	183	198	174
RM-P038	131	216	135
RM-P039	146	147	113
RM-P040	493	572	526
Durchschnitt über alle Teststücke	371	404	365

Tabelle 6.7: Vergleich der MsDTW-Methode, DTW mit CN-Merkmalen und DTW mit CNO-Merkmalen bezüglich $\varnothing_{abs}$ anhand von Popularmusik.

Während bei klassischer Musik in den meisten Fällen der Harmonieverlauf im Vordergrund steht, wird ein aktuelles Popstück häufig eher rhythmusbezogen komponiert. Daraus ergibt sich auch die häufige Verwendung perkussiver Instrumente, wie Schlagzeug oder Effektinstrumenten. In manchen Genres finden sich Stücke, in denen kein melodischer Anteil mehr vorhanden ist. So beschränken sich einige HipHop-Stücke auf Percussion-Elemente und atonalen Sprechgesang.

Mit diesem Experiment wird untersucht, welche Ergebnisse mit den vorgestellten Synchronsationsmethoden auf Popmusik erzielt werden können. Als Testdaten dienen einige Beispielstücke aus dem Unterbereich „Pop“ der RWC-Datenbank. In Tabelle 6.7 werden die Werte von $\varnothing_{abs}$ unter Verwendung der MsDTW-Methode, DTW mit CN-Merkmalen und DTW mit CNO-Merkmalen verglichen. Dabei wurde der Parameter β der Kostenmaße $\mathbf{c}^{CN}_{\alpha,\beta}$ und $\mathbf{c}^{CNO}_{\alpha,\beta}$ mit 0.75 belegt. In Schritt 3 der automatischen Evaluation wurde in allen Fällen WarpTime 2 verwendet. Die Snapping-Methode wurde nicht verwendet. Bei DTW mit CN- bzw. CNO-Merkmalen wurde eine Tube-Bereichseinschränkung der Größe 25 verwendet. Alle sonstigen Parameter entsprechen den Standardparametern, wie sie in den entsprechenden Kapiteln eingeführt wurden.

Vom Aufbau ist dieses Experiment mit dem ersten vergleichbar, jedoch fallen die Werte von $\varnothing_{abs}$ in diesem Experiment deutlich größer aus als in Experiment 1. So beträgt die durchschnittliche Absolutabweichung beim Beispiel „RM-P030“ fast eine Sekunde. Jede berechnete Noteneinsatzzeit weicht damit im Durchschnitt etwa eine Sekunde vom vorgegebenen Zeitpunkt ab. Eine wichtige Ursache dafür ist, dass die Singstimme in einigen RWC-Stücken bei der Annotation ausgelassen wurde. Eine andere Ursache ist, dass einige RWC-Stücke relativ einfach produziert sind und sich ähnelnde Abschnitte vielfach wiederholt werden. Durch den entstehenden monotonen Harmonieverlauf tritt der Effekt auf, dass zwei Abschnitte einander durch die Synchronisation zugeordnet werden, die zwar ähnlich sind, sich semantisch aber nicht entsprechen.

Bei einigen Stücken treten jedoch noch zusätzliche Effekte auf. Beim Stück „RM-P030“, das

dem HipHop-Genre zuzuordnen ist, wird die Komposition von perkussiven Elementen dominiert. Einem perkussiven Instrument lässt sich für gewöhnlich keine Tonhöhe zuordnen. Solche Stücke können deshalb nicht sinnvoll über Chroma-Merkmale beschrieben werden, wodurch eine Synchronisation auf Basis dieser Merkmale nahezu beliebige Ergebnisse liefert. Die Stücke „RM-P031“ und „RM-P032“ zeigen ähnlich große Werte für $\varnothing_{abs}$. Beim Stück „RM-P031“ werden wie bei „RM-P030“ hauptsächlich perkussive Instrumente eingesetzt. Dazu spielt durchgehend ein tiefer Bass, der an sich zwar tonal ist, jedoch von starken perkussiven Nebeneffekten begleitet ist und zudem nicht sehr lautstark ist. Die Gesangsstimme ist zudem verzerrt, was sie ebenfalls weniger tonal klingen lässt. Auch bei „RM-P032“ finden sich atonale Elemente als Ursache für den großen Wert von $\varnothing_{abs}$.

6.3.8 Detailuntersuchung - Burgmüller Beispiel

In diesem Abschnitt werden die Ergebnisse für das Beispiel „Burgmüller op02 no1 (Meinard Müller)“ im Detail untersucht. Alle Parameter entsprechen den Vorgaben wie in Experiment 1. Der Parameter β der Kostenmaße $\mathbf{c}^{CN}_{\alpha,\beta}$ und $\mathbf{c}^{CNO}_{\alpha,\beta}$ wurde jedoch, davon abweichend, mit 0.75 belegt. Abbildung 6.3 zeigt einen Ausschnitt der Kostenmatrizen und den berechneten Warping-Pfad unter Verwendung von

(a) der MsDTW-Methode,

(b) DTW mit CN-Merkmalen,

(c) DTW mit CNO-Merkmalen.

Vergleicht man die drei Kostenmatrizen, so erkennt man in den CN- und CNO-Kostenmatrizen Einträge mit niedrigeren Kosten im Vergleich zur Umgebung. Diese deuten auf Noteneinsatzzeiten hin, die durch die verwendeten Merkmale erkannt wurden. Betrachten wir den Bereich $[0, 2]$ Sekunden horizontal, sowie $[0, 2.5]$ Sekunden vertikal. Das Stück beginnt mit vier A3 - C4 - E4 Akkorden, die in diesem Bereich durch die CN- bzw. CNO-Merkmale in beiden Varianten des Stücks erkannt und über den berechneten Warping-Pfad korrekt einander zugeordnet wurden. Der Warping-Pfad der MsDTW-Methode durchläuft diesen Bereich weitestgehend diagonal und bildet die Noteneinsätze nicht korrekt aufeinander ab. Im darauf folgenden Bereich von $[2, 3]$ Sekunden horizontal sowie $[2.5, 3.5]$ Sekunden vertikal, findet sich ein Beispiel eines leicht degenerierten Warping-Pfads, der unter Verwendung von CN-Merkmalen auftritt. In diesem Bereich werden verschiedene 1/16 Noten angespielt. In der Kostenmatrix der CN-Merkmale erkennt man anhand der punktuell erniedrigten Kosten, dass die zugehörigen Einsatzzeiten erkannt wurden. Da das Kostenmaß $\mathbf{c}^{CN}_{\alpha,\beta}$ den kostensenkenden Faktor β jedoch in keiner Weise gewichtet, werden die Kosten im betrachteten Bereich zu stark und indifferent gesenkt. Der über CN-Merkmale bestimmte Warping-Pfad nimmt deshalb an dieser Stelle einen unerwünschten Lauf. Der Warping-Pfad unter Verwendung von CNO-Merkmalen ordnet sich entsprechende Stellen der beiden Varianten in diesem Bereich korrekt zu.

Um genauer zu betrachten, wie groß die Abweichung des berechneten vom erwünschten Ergebnis in den einzelnen Zeitbereichen ist, werden nun *Abweichungsgraphen* verwendet. Zur

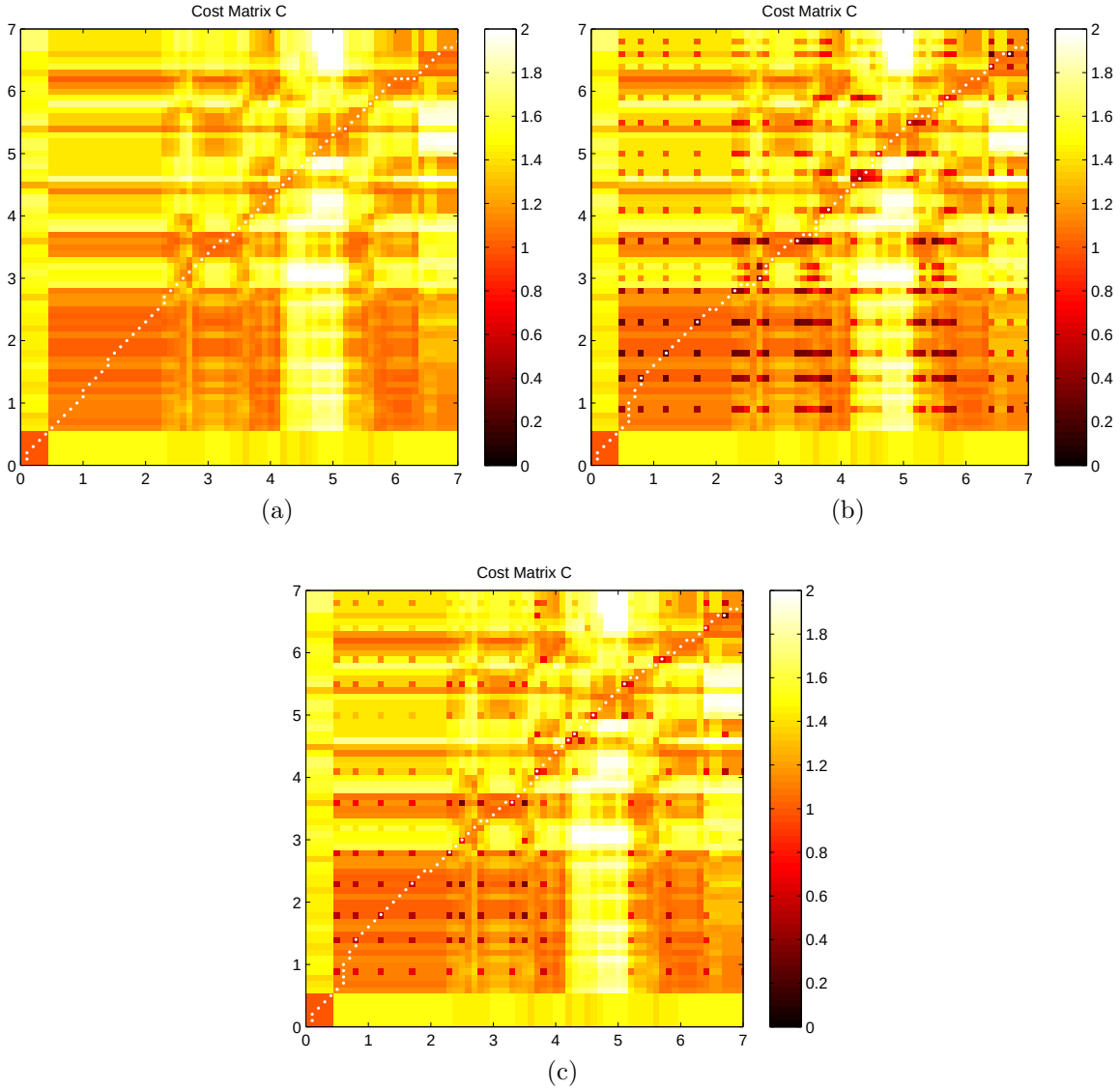


Abbildung 6.3: Ausschnitt der Kostenmatrix unter Verwendung von: (a) der MsDTW-Methode, (b) DTW mit CN-Merkmalen, (c) DTW mit CNO-Merkmalen. Die Audioaufnahme wurde vertikal und die verzerrten MIDI-Daten horizontal aufgetragen.

Erklärung eines Abweichungsgraphen sei daran erinnert, dass über die automatische Evalutionsmethode für jede MIDI-Note bestimmt werden kann, wie weit die berechnete von der vorgegebenen Einsatzzeit abweicht. Die Abweichung wird dabei über die Formel:

$$\Delta_t = t_{ref} - t$$

berechnet, wobei Δ_t die Abweichung, t_{ref} die vorgegebene Einsatzzeit und t die berechnete Einsatzzeit bezeichnet. Ein positiver Wert von Δ_t bedeutet somit, dass die berechnete Einsatzzeit zu klein ist und die Note zu früh angespielt wird. Entsprechend ist ein negativer Wert dahingehend zu interpretieren, dass die Note zu spät angespielt wird. In einem Abweichungsgraphen trägt man für jede MIDI-Note die vorgegebene Noteneinsatzzeit t_{ref} auf

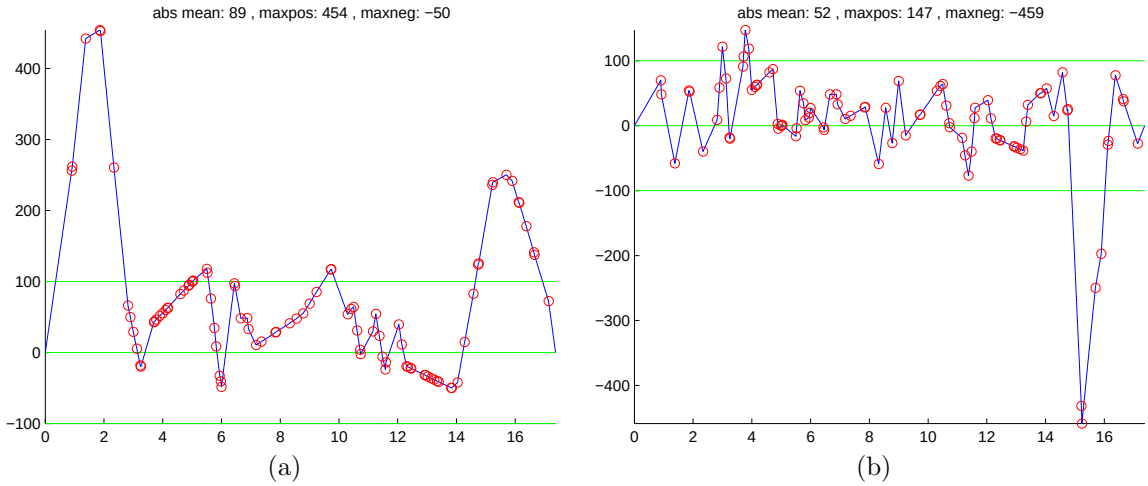


Abbildung 6.4: Abweichungsgraphen, wobei zur Synchronisation verschiedene Methoden verwendet wurden: (a) MsDTW-Methode, (b) DTW mit CN-Merkmalen. Zur Beschreibung siehe Fließtext.

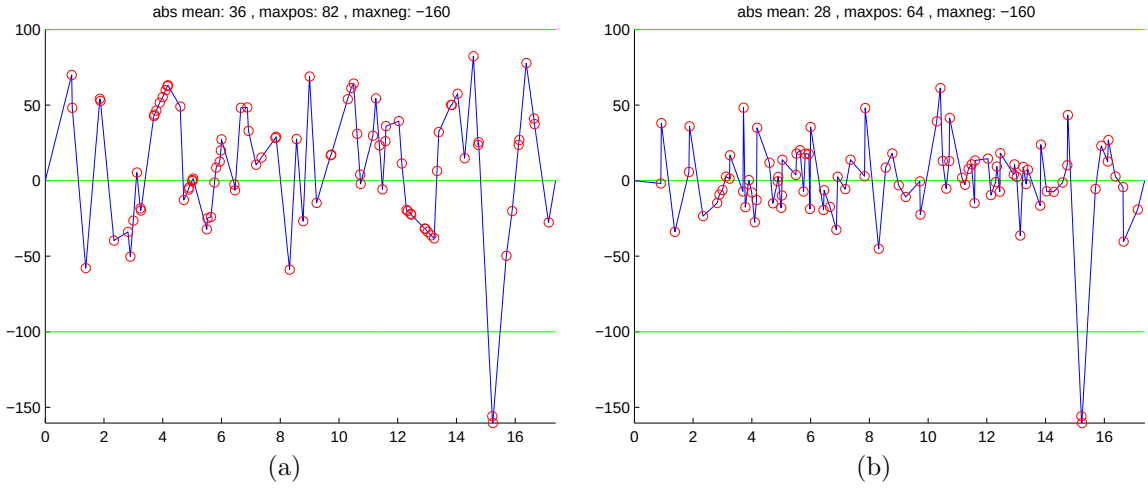


Abbildung 6.5: Abweichungsgraphen, wobei zur Synchronisation verschiedene Methoden verwendet wurden: (a) DTW mit CNO-Merkmalen, (b) DTW mit CNO-Merkmalen und Nachverarbeitung mittels Snapping-Methode. Zur Beschreibung siehe Fließtext.

der X-Achse und die Notenabweichung Δ_t auf der Y-Achse auf. Die Abbildungen 6.4 und 6.5 zeigen die Abweichungsgraphen für das Stück „Burgmüller op02 no1 (Meinard Müller)“, wobei jeweils eine andere Synchronisationsmethode verwendet wurde. Untersucht wurden die MsDTW-Methode, DTW mit CN-Merkmalen und DTW mit CNO-Merkmalen. Zusätzlich wurde in einem vierten Abweichungsgraph DTW mit CNO-Merkmalen mit nachgeschalteter Snapping-Methode untersucht. Dabei wurden die Zuordnungen von Einsatzzeit und Abweichung durch rote Kreise markiert. Der blaue Graph ergibt sich durch lineare Interpolation dieser Punktzuordnungen. Um die Zeitauflösung der Merkmale zu verdeutlichen, die bei der Synchronisation verwendet wurden, wurden zusätzlich drei Hilfslinien bei 100 ms, 0 ms und

-100 ms eingezeichnet.

In Abbildung 6.4(a) fällt eine relativ große Abweichung im Bereich von $[0, 2.5]$ Sekunden auf. Die zugrundeliegende Ursache wurde bereits bei der Untersuchung der Kostenmatrizen weiter oben betrachtet. Des Weiteren lässt sich erkennen, dass die Abweichung, ausgehend von der MsDTW-Methode, über DTW mit CN-Merkmalen und CNO-Merkmalen bis zur Nachverarbeitung mittels der Snapping-Methode, jeweils kleiner wird. Auffällig ist jedoch der Wert der Abweichung im Bereich von etwa 15 Sekunden. Bei allen Methoden außer der MsDTW-Methode treten hier relativ betragsstarke, negative Werte auf. Es stellte sich heraus, dass hier ein Fehler in der manuellen Annotation vorliegt. Zudem wurde in diesem Bereich ein Noteneinsatz fälschlicherweise nicht durch Novelty-Merkmale erkannt. Durch diese Kombination wird in diesem Bereich lokal ein falscher Pfadverlauf begünstigt.

6.3.9 Anmerkung zu den Experimenten

Bei allen hier gezeigten Experimenten wurde derselbe Verzerrungsvektor

$$v = (0.612, 1.197, 0.956, 1.345, 0.972, 0.934, 1.276, 1.020, 0.762, 1.137) \in \mathbb{R}^{10}_{>0}$$

in Schritt 1 der automatischen Evaluation verwendet. Versuche mit anderen Verzerrungsvektoren zeigten teilweise deutliche Abweichungen von den hier gezeigten Ergebnissen. Vor allem extreme Verzerrungen bewirkten eine starke Verschlechterung der Synchronisationsgenauigkeit, wobei eine extreme Verzerrung durch besonders große bzw. kleine Einträge im Verzerrungsvektor v gekennzeichnet ist. Man kann sich überlegen, dass ein Warping-Pfad hauptsächlich horizontal bzw. vertikal verlaufen sollte, wenn sich die beiden zu synchronisierenden Varianten durch eine extreme Verzerrung unterscheiden. Eine Verschlechterung der Synchronisationsgenauigkeit kann nun damit begründet werden, dass ein solcher Verlauf durch die leichte Diagonalpräferenz (DTW-Gewichte) der Verfahren in Teilen verhindert wird.

Kapitel 7

Zusammenfassung und Ausblick

In dieser Arbeit wurden bestehende Verfahren zur Synchronisation harmoniebasierter Musik weiterentwickelt und neuartige vorgeschlagen. Ausgangspunkt war dabei die MsDTW-Methode, die sich in früheren Arbeiten bereits als zuverlässige Synchronisationslösung erwiesen hat. In Experimenten zeigte sich, dass mit den beschriebenen Verfahren in vielen Fällen eine genauere Synchronisation als mit bisherigen Verfahren erzielt werden kann.

Erweiterung der MsDTW-Methode

Zunächst wurde die MsDTW-Methode untersucht. Dabei konnte festgestellt werden, dass eine Synchronisationsmethode, die nur harmoniebasierte Merkmale einsetzt, nicht durchgehend befriedigende Ergebnisse liefern kann. Deshalb wurden CN-Merkmale vorgeschlagen, die zusätzlich zum Harmonieverlauf auch Informationen über Noteneinsatzzeiten kodieren. Zur robusten Erkennung der Einsatzzeiten wurden dabei Novelty-Merkmale eingesetzt. Die Funktion $\mathbf{c}^{CN}_{\alpha,\beta}$, die ein lokales Kostenmaß auf CN-Merkmalen definiert, entspricht in weiten Teilen dem Kostenmaß, das in der MsDTW-Methode eingesetzt wird. Ein wichtiger Unterschied besteht im Faktor β, der eine Kostensenkung bewirkt, wenn in beiden zu vergleichenden CN-Merkmalen eine Noteneinsatzzeit erkannt wurde. Um die MsDTW-Methode mit diesem erweiterten Verfahren vergleichen zu können, wurde eine Methode zur automatischen Evaluation von Synchronisationsmethoden vorgestellt. Mit diesem Verfahren konnte die Genauigkeit einer Synchronisation erfasst werden. In Versuchen zeigte sich, dass die Verwendung von DTW mit CN-Merkmalen überraschenderweise oftmals eine Verschlechterung der Synchronisationsgenauigkeit gegenüber der MsDTW-Methode bewirkt. Als Ursache konnte das Auftreten bestimmter degenerierter Warping-Pfade identifiziert werden, deren Verlauf durch den Einsatz des lokalen Kostenmaßes $\mathbf{c}^{CN}_{\alpha,\beta}$ begünstigt wird. Um diesem Effekt entgegen zu wirken, wurden Onset-Merkmale hinzugezogen, die zusätzliche Informationen über die angespielten Tonhöhen kodieren. Durch Erweiterung der CN- zu CNO-Merkmalen können diese Informationen verwendet werden, um den kostensenkenden Faktor β im Kostenmaß $\mathbf{c}^{CNO}_{\alpha,\beta}$ zu gewichten. In Experimenten zeigte sich, dass eine Synchronisation unter Verwendung von DTW mit CNO-Merkmalen, im Vergleich mit der MsDTW-Methode, in den meisten Fällen genauer ist.

Effiziente Umsetzung von DTW mit CN- und CNO-Merkmalen

Ein wesentlicher Bestandteil der MsDTW-Methode ist der DTW-Multiskalenansatz, der eine hohe Laufzeiteffizienz ermöglicht. Dieser Ansatz konnte für die Verfahren basierend auf CN-

bzw. CNO-Merkmalen adaptiert werden. Grundprinzip dabei war, einen DTW-Einschränkungsbereich anhand eines Warping-Pfads festzulegen, der mit Hilfe der MsDTW-Methode berechnet wurde. Dieses Vorgehen begründete sich in der Beobachtung, dass ein Warping-Pfad, der über DTW mit CN- bzw. CNO-Merkmalen berechnet wurde, typischerweise einen ähnlichen Verlauf wie ein Warping-Pfad nimmt, der mittels der MsDTW-Methode berechnet wurde. Zur Definition eines Einschränkungsbereichs wurden die MovingWindow- und die Tube-Methode vorgestellt. Erstere zeichnet sich durch einen sehr geringen Speicherplatzbedarf aus. Sie setzt jedoch voraus, dass ein zu berechnender optimaler Warping-Pfad nicht nur global, sondern auch lokal einen ähnlichen Verlauf wie der Warping-Pfad nimmt, über den der Einschränkungsbereich definiert wird. Bei der Tube-Methode ist dies nicht notwendig.

Erhöhte Genauigkeit der automatischen Annotation einer Audioaufnahme

Nach der Synchronisation einer Audioaufnahme und MIDI-Daten, können die MIDI-Einsatzzeiten unter Verwendung einer Zeitzuordnungsfunktion an die physikalischen Einsatzzeiten der Audioaufnahme angepasst werden. Auf diese Weise kann ein Synchronisationsergebnis als automatische Annotation von Audioaufnahmen betrachtet werden. Für die Anpassung der Einsatzzeiten wurden bisher Zeitzuordnungsfunktionen verwendet, die lediglich einfache Strategien umsetzen. Die Genauigkeit der Zuordnung von Zeitpunkten entsprach deshalb nur der Auflösung der Merkmale, die bei der Synchronisation eingesetzt wurden. Als Alternative zu diesen einfachen Funktionen wurde WarpTime 2 vorgestellt. Man nutzt bei WarpTime 2, dass Chroma-, CN- bzw. CNO-Merkmale aus bestimmten Abschnitten eines Audiosignals berechnet werden und daher mit jedem Merkmal ein Zeitbereich assoziiert ist. Bei der Beschreibung von WarpTime 2 wurde nun ein Warping-Pfad dahingehend interpretiert, dass er festlegt, wie sehr diese Zeitbereiche gestreckt oder gestaucht werden müssen, damit sich die beiden zu synchronisierenden Musikstücke zeitlich genau entsprechen. Auf Basis dieser Interpretation konnte eine so genannte lokale Verzerrungsfunktion angegeben werden, die für jeden Zeitbereich lokal angibt, wie stark diese Stauchung bzw. Streckung ausfallen muss. Mittels einer Integration dieser Funktion konnte eine Zuordnung von Zeitpunkten angegeben werden. In Experimenten zeigte sich, dass eine automatische Annotation unter Verwendung von WarpTime 2 oftmals genauer ist als mit dem bisher eingesetzten WarpTime 1 Verfahren.

Nachverarbeitung mittels Onset-Merkmalen

Wird eine Synchronisation zur automatischen Annotation einer Audioaufnahme eingesetzt, werden die in MIDI-Daten gespeicherten Einsatzzeiten an die physikalischen Einsatzzeiten der Audioaufnahme angepasst. Die erwartete Genauigkeit dieser Anpassung hängt von der Zeitauflösung der Merkmale ab, die bei der Synchronisation verwendet werden. Über eine Nachverarbeitung der angepassten MIDI-Einsatzzeiten konnte die Genauigkeit der automatischen Annotation weiter erhöht werden. Dazu wurden Onset-Merkmale verwendet, die Noteneinsatzzeiten mit höherer zeitlicher Auflösung als CN- oder CNO-Merkmale kodieren. Man ging nun davon aus, dass die angepassten MIDI-Einsatzzeiten bereits grob mit den physikalischen Einsatzzeiten der Audioaufnahme übereinstimmen. Bei der Nachverarbeitung wurde dann ein Zeitbereich um die angepasste MIDI-Einsatzzeit nach passenden Onset-Ereignissen

der Audioaufnahme durchsucht. Bei einem Treffer wurde die Einsatzzeit der MIDI-Note durch die eines gefundenen Onset-Ereignisses ersetzt.

Ausblick

Die Musiksynchronisation ist ein aktuelles Forschungsgebiet mit noch zahlreichen offenen Fragestellungen. In Abschnitt 5.3 wurde bereits ein Verfahren zur partiellen Synchronisation in Grundzügen beschrieben. Bei dieser Form der Synchronisation berücksichtigt man, dass Varianten von Musikstücken sich nicht nur in Dynamik, Klang und Tempoverlauf unterscheiden können, sondern auch durch strukturelle Änderungen, zum Beispiel in Form von ausgelassenen oder wiederholten Soli, Kadenzen oder Strophen. Für bestimmte Arten von Variationen eines Musikstücks existieren bisher jedoch keine passenden Synchronisationsmethoden. So fand sich in der Literatur keine Methode zur Synchronisation von Varianten eines Musikstücks, die sich aufgrund lokaler Transpositionen unterscheiden. Eine Transposition heißt dabei lokal, wenn die Tonhöhe nicht für ein ganzes Musikstück, sondern nur in bestimmten Abschnitten verändert wird. Des Weiteren zeigte sich in den Experimenten, dass das Musikgenre eine entscheidende Rolle bei der Musiksynchronisation einnimmt. So sind moderne Popstücke häufig eher durch ihren Rhythmus und weniger durch ihren Harmonieverlauf gekennzeichnet. Vorhandene Harmonien sind zudem häufig sehr einfach aufgebaut, werden vielfach wiederholt und charakterisieren deshalb eine bestimmte Position innerhalb eines Musikstücks nur unzureichend. Für solche Stücke liefern die vorgestellten Methoden oft unbefriedigende Ergebnisse. Um diese Stücke synchronisieren zu können, müssen andere Merkmalstypen verwendet werden.

Aber auch die in dieser Arbeit vorgestellten Methoden können weiter verbessert werden. So wurde in Abschnitt 6.3.8 anhand eines Beispiels festgestellt, dass die erzielbare Genauigkeit der hier vorgestellten Methoden stark davon abhängt, ob Novelty- und Onset-Merkmale Noteneinsatzzeiten korrekt erkennen. Eine Erhöhung der Robustheit dieser Merkmalstypen wäre folglich wünschenswert. Weiterhin legen Erkenntnisse aus Abschnitt 5.3.1 nahe, dass man Onset-Merkmale aus Gründen der Robustheit analog zu Chroma-Merkmalen auf zwölf Halbtonklassen beschränken sollte. Auf diese Weise könnten bei fehlerhafter Erkennung von Einsatzzeiten Oberton-bezogene Effekte reduziert oder ausgeschaltet werden.

Anhang A

Quelltext-Referenz

In diesem Abschnitt werden die Deklarationen der wichtigsten Funktionen aufgelistet, die während der vorliegenden Arbeit entstanden oder angepasst wurden. Alle Methoden wurden mit Matlab 2006b entwickelt und getestet.

Merkmalsextraktion

```
function [f_peaks,sideinfo] = audio_to_FBpitchOnsetPeaks(f_audio,
    parameter,sideinfo);
function [f_pitch,sideinfo] = audio_to_FBpitchSTMSP(f_audio,parameter,
    sideinfo);
function [f_novelty,f_noveltyPeaks,f_noveltyTimes,sideinfo] =
    audio_to_STFTnoveltyPeaks(f_audio,parameter,sideinfo);
function pitch_features_midi = midird4_to_pitchSTMSP(midi,
    FB_window_stepsize_ms, transpose);
function [f_CENS_cell,sideinfo] = STMSP_to_CENS(f_pitch,parameter,
    sideinfo);
function [f_chroma_norm,f_chroma,sideinfo] = STMSP_to_chroma(f_pitch,
    parameter,sideinfo);
function [f_audio,sideinfo] = wav_to_audio(dir_abs,dir_rel,wavfilename,
    parameter);
```

Sonifikation

```
function f_audio = sonify_chroma(f_chroma,stepsize_ms,fs,parameter);
function f_audio = sonify_Wav_annotationSine(wav,annotation,match,fs,
    parameter);
function f_audio = sonify_Wavfile1_warpWavfile2_path(wavfile1,wavfile2,
    warpingpath,stepsize_samples,parameter);
function f_audio = sonify_Wavfile_annotationSine(wavfile,annotation,fs,
    parameter);
function varargout = sonify_Wavfile_warpMidifile_path(wavfile,midifile,
    warpingpath,stepsize_ms,parameter);
```

Zeitzuordnungsfunktionen

```
function [midi,ext,tempoList,lyrics] = warpMidi(midi,info,ext,tempoList,
    lyrics, path, stepSize_ms);
function assignedTime_ms = warpTime1(time_ms, path, stepSize_ms);
function assignedTime_ms = warpTime2(time_ms, path, stepSize_ms);
```

Automatische Evaluation

```
function [midi,ext,tempoList,lyrics] = distortMidi(distVector, midi,info,
    ext,tempoList,lyrics);
function [diversions, attimes, absmean, absstd, maxpos, maxneg] =
    reference_comparison2(annotation_ref, annotation, parameter);
function [diversions, attimes, absmean, absstd, maxpos, maxneg] =
    reference_midi_comparison(midi_ref, midi_comp, parameter);
function WriteSpecialTextFileWithSyncResult(outputtextfile,wavfile,
    midifile,warpingpath, stepSize_ms);
function [midi,ext,tempoList,lyrics] = applyTimeOffsetOnMidi(offset_ms,
    length_ms,midi,info,ext,tempoList,lyrics);
function varargout = getOffsetFromAudioForMidi(f_audio,fs,midi,parameter)
    ;
```

Synchronisationsmethoden

```
function midi = snapMidiToNearestPitchOnset(midi, tempoList, f_peaks,
    searchTimeSlice_ms);
function match = snapMatchToNearestPitchOnset(annotation, match, fs,
    f_peaks, searchTimeSlice_ms);
function varargout = advancedSyncDtw(V,W,parameter);
function onsetsPerFeature = midi_to_cn(midi,parameter);
function pitchesPerFeature = midi_to_cno(midi,parameter);
function onsetsPerFeature = novelty_to_cn(novelty_peaks,novelty_times,
    parameter);
function pitchesPerFeature = novelty_onset_to_cno(novelty_peaks,
    novelty_times,f_peaks,parameter);
```

Anhang B

MIDI-Tonhöhen-Tabelle

Halbton ISO	Frequenz Hz	Halbton MIDI	Halbton ISO	Frequenz Hz	Halbton MIDI	Halbton ISO	Frequenz Hz	Halbton MIDI
C0	-	12	C4	261.626	60	C8	4186.009	108
C#0	-	13	C#4	277.183	61	C#8	4434.922	109
D0	-	14	D4	293.665	62	D8	4698.637	110
D#0	-	15	D#4	311.127	63	D#8	4978.032	111
E0	-	16	E4	329.628	64	E8	5274.042	112
F0	-	17	F4	349.228	65	F8	5587.652	113
F#0	-	18	F#4	369.994	66	F#8	5919.912	114
G0	-	19	G4	391.995	67	G8	6271.928	115
G#0	-	20	G#4	415.305	68	G#8	6644.876	116
A0	27.500	21	A4	440.000	69	A8	7040.000	117
A#0	29.135	22	A#4	466.164	70	A#8	7458.620	118
B0	30.868	23	B4	493.883	71	B8	7902.133	119
C1	32.703	24	C5	523.251	72	C9	8372.019	120
C#1	34.648	25	C#5	554.365	73	C#9	8869.845	121
D1	36.708	26	D5	587.330	74	D9	9397.273	122
D#1	38.891	27	D#5	622.254	75	D#9	9956.064	123
E1	41.203	28	E5	659.255	76	E9	10548.083	124
F1	43.654	29	F5	698.457	77	F9	11175.305	125
F#1	46.249	30	F#5	739.989	78	F#9	11839.823	126
G1	48.999	31	G5	783.991	79	G9	12543.855	127
G#1	51.913	32	G#5	830.609	80	G#9	13289.752	-
A1	55.000	33	A5	880.000	81	A9	-	-
A#1	58.270	34	A#5	932.328	82	A#9	-	-
B1	61.735	35	B5	987.767	83	B9	-	-
C2	65.406	36	C6	1046.502	84			
C#2	69.296	37	C#6	1108.731	85			
D2	73.416	38	D6	1174.659	86			
D#2	77.782	39	D#6	1244.508	87			
E2	82.407	40	E6	1318.510	88			
F2	87.307	41	F6	1396.913	89			
F#2	92.499	42	F#6	1479.978	90			
G2	97.999	43	G6	1567.982	91			

Halb-ton ISO	Frequenz Hz	Halb-ton MIDI	Halb-ton ISO	Frequenz Hz	Halb-ton MIDI	Halb-ton ISO	Frequenz Hz	Halb-ton MIDI
G#2	103.826	44	G#6	1661.219	92			
A2	110.000	45	A6	1760.000	93			
A#2	116.541	46	A#6	1864.655	94			
B2	123.471	47	B6	1975.533	95			
C3	130.813	48	C7	2093.005	96			
C#3	138.591	49	C#7	2217.461	97			
D3	146.832	50	D7	2349.318	98			
D#3	155.564	51	D#7	2489.016	99			
E3	164.814	52	E7	2637.021	100			
F3	174.614	53	F7	2793.826	101			
F#3	184.997	54	F#7	2959.956	102			
G3	195.998	55	G7	3135.964	103			
G#3	207.652	56	G#7	3322.438	104			
A3	220.000	57	A7	3520.000	105			
A#3	233.082	58	A#7	3729.310	106			
B3	246.942	59	B7	3951.066	107			

Erklärung der selbständigen Arbeit

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig angefertigt, keine anderen als die angegebenen Hilfsmittel benutzt, sowie Zitate kenntlich gemacht habe.

Bonn, den 19. Dezember 2007

Sebastian Ewert

Literaturverzeichnis

[Ari02] Arifi, Vlora: *Algorithmen zur Synchronisation von Musikdaten in Partitur-, MIDI- und PCM Format*. Dissertationsschrift, 2002.

[Bla98] Blackham, E.: *Die Physik der Musikinstrumente*, Band 2. Akademischer Verlag, 1998.

[BW05] Bartsch, Mark A. und Gregory H. Wakefield: *Audio thumbnailing of popular music using chroma-based representations*. IEEE Transactions on Multimedia, 7(1):96–104, 2005.

[CM01] Clausen, Michael und Meinard Müller: *Zeit-Frequenz-Analyse und Wavelettransformationen*. `http://www-mmdb.iai.uni-bonn.de`, 2001.

[CM07] Clausen, Michael und Meinard Müller: *Inhaltsbasiertes Multimediaretrieval*. Vorlesungsskript, 2007.

[DW05] Dixon, Simon und Gerhard Widmer: *MATCH: A Music Alignment Tool Chest*. In: *ISMIR*, Seiten 492–497, 2005.

[Fol84] Folland, G.: *Real Analysis*. John Wiley & Sons, 1984.

[FR91] Fletcher, N. und T. Rossing: *The Physics of Musical Instruments*. Springer-Verlag, 1991.

[Got02] Goto, Masataka: *AIST RWC-Datenbank*. `http://staff.aist.go.jp/m.goto/RWC-MDB/`, 2002.

[Hem97] Hempel, Christoph: *Neue allgemeine Musiklehre*. Atlantis-Schott-Verlag, 1997.

[IT04] Izumo, Masanao und Tuukka Toivonen: *Timidity MIDI-Synthesesoftware*. `http://timidity.sourceforge.net/`, 2004.

[KG03] Kovar, Lucas und Michael Gleicher: *Flexible automatic motion blending with registration curves*. In: *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, Seiten 214–224, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[KP95] Kleijn, W. und K Paliwal: *Speech and Coding Synthesis*. Elsevier, 1995.

[MA07] Müller, Meinard und Daniel Appelt: *Path-contrained partial music synchronization*. To be published, 2007.

[Mat06] Mattes, Henning: *Effiziente Synchronisation von Musikdatenströmen*. Diplomarbeit, 2006.

[MKC05] Müller, M., F. Kurth und M. Clausen: *Audio Matching via Chroma-based Statistical Features*. In: *Proc. ISMIR, London, GB*, 2005.

[MKR04] Müller, Meinard, Frank Kurth und Tido Röder: *Towards an Efficient Algorithm for Automatic Score-to-Audio Synchronization*. In: *ISMIR*, 2004.

[MMK06] Müller, Meinard, Henning Mattes und Frank Kurth: *An Efficient Multiscale Approach to Audio Synchronization*. In: *ISMIR*, Seiten 192–197, 2006.

[Mül07] Müller, Meinard: *Methods for Robust and Efficient Multimedia Retrieval*. Springer, 2007.

[NHT03] N. Hu, R. Dannenberg und G. Tzanetakis: *Polyphonic audio matching and alignment for music retrieval*. IEEE WASPAA, 2003.

[RJ93] Rabiner, L. und B. Juang: *Fundamentals of speech recognition*. Prentice Hall, 1993.

[SC04] Salvador, S. und P. Chan: *FastDTW: Toward accurate dynamic time warping in linear time and space*. In: *KDD Workshop on Mining Temporal and Sequential Data*, Seiten 70–80, 2004.

[Set05] Sethares, W.: *Tuning, Timbre, Spectrum, Scale*. Springer, 2005.