# Fan Chirp Transformation

Seminararbeit aus Algorithmen in Akustik und Computermusik 2, SE

Josef Hölzl

Betreuung: Dr Franz Zotter, DI Matthias Frank

Graz, 10. April 2011


institut für elektronische musik und akustik

**Zusammenfassung**

In dieser Seminararbeit werden Grundlagen und Methoden der Grundfrequenz-Detektion und die Vorteile der *Fan Chirp Transformation (FChT)* beschrieben. In Bezug auf [6] werden Theorie und praktische Lösungsansätze diskutiert. Darüber hinaus wird eine effiziente Implementation (Analyse und Synthese) in MATLAB präsentiert.

**Abstract**

In this work the main ideas of *Pitch Detection)* and the advantages of the *Fan Chirp Transform (FChT)* will be explained. According to [6], theory and practical solutions are discussed. Moreover, an efficient implementation (analysis and synthesis) in MATLAB is presented.

# Inhaltsverzeichnis

*Keywords:* Fan-Chirp Transform; Chirp Analysis; Voice Activity Detection

# 1 Einleitung

Mehrere Techniken zur Bestimmung und Synthese der Grundfrequenz wurden in den letzten Jahrzehnten entwickelt, sind aber in der Verwendung immer noch limitiert. Dabei steckt die größte Komplexität in der Modellierung mit Teiltonmustern und die dazu benötigte Aufspaltung bzw. Detektion der Grundfrequenz.

Musikalische Signale haben oft ein harmonisches Spektrum mit ein oder mehreren Grundtönen. Die meisten Sprach-Algorithmen basieren auf der *Short Time Fourier Transformation (STFT)*. Da immer nur ein zeitlich begrenzter Abschnitt (*Window*) des Signals betrachtet wird, kann auch eine Aussage über das zeitliche Auftreten einer Frequenz gemacht werden. Die *STFT* setzt voraus, dass in dem Analyse-Frame das Signal stationär ist. Sprache besteht auch aus nicht-stationäre Signalen mit komplexen Charakteristiken. In genau diesen Abschnitten (Beginn, Ende) stecken wichtige Information für die Berechnung.

Für die Analyse nicht-stationärer Signale benötigt man eine hohe Frequenzauflösung in tiefen und mittleren Frequenzen, da besonders in diesem Bereich Harmonische oft vorkommen. Die natürliche Grenze der STFT-Methode ist durch die definierte Fensterbreite gegeben: bei breiten Fenstern ist die zeitliche Lokalisierung schlecht, bei schmalen ist die Schätzung des Spektrums sehr unsicher. Auf der anderen Seite benötigt man breite Fenster zur Auflösung niedriger Frequenzen, während dessen kurze nicht-stationäre Oszillationen nur mit kleinen Fensterbreiten identifizierbar sind und sich bei großen Fenstern weitgehend herausmitteln würden.

Eine Alternative mit einem anderen Ansatz stellt die *Wigner-Ville-Transformation* dar. Diese benutzt keinerlei Fensterfunktionen und zeigt daher auch keinen Fenstereffekt. Ein Nachteil ist hingegen die sogenannte *Kreuzterm-Interferenz*, die unweigerlich wieder zu einer Unschärfe im Zeit-Frequenz Spektrum führt.

Diese inhärente Kopplung von Zeit- und Frequenzauflösung wird mittels der *FChT* besser gelöst und in den folgenden Abschnitten vorgestellt.

Der Vollständigkeit halber muss erwähnt werden, dass es grundsätzlich noch andere Transformationen gibt, die mit dem Wort "Chirp"verbunden werden: *Chirp-Z*, *Chirplet*, *Fractional Fourier* Transformation und auch *warped time operators*. Für das Verständnis dieses Arbeit sind die *Chirplet Transform (CT)* und *warping operator* am relevantesten und werden kurz vorgestellt.

Die CT-Transformation wird als inneres Produkt zwischen Signal und einem Chirplet beschrieben

$$X_\beta(f) = \int_{-\infty}^{\infty} x(t)\ g^*_{\varrho,\tau,f,\beta}(t)dt\ , \tag{1}$$

wobei $g_{\varrho,\tau,f,\beta}(t)$ einem Gausschen Chirplet mit Einheitsenergie entspricht:

$$g_{\varrho,\tau,f,\beta}(t) = \frac{e^{-(1/2)((t-\tau)/\varrho)^2}}{\sqrt[4]{\pi\varrho^2}} e^{j2\pi(f(t-\tau)+(1/2)\beta(t-\tau)^2)} \ . \tag{2}$$

Das Analyseintegral ist definiert als:

$$X_{\psi(\cdot)}(f) = \int_{-\infty}^{\infty} x(\psi(t)) \ \sqrt{|\psi'(t)|} \ e^{-j2\pi ft} dt \ , \tag{3}$$

mit $\psi(t)$ als kontinuierliche Zeitabbildung (*time warping*) und $\psi^{'}(t)$ als dessen Ableitung. Damit eine Fourier-Transformation möglich ist, wird das Zeitsignal zur Vereinfachung mit einem Zeitverzerrungsoperator $\psi(t)$ verwendet.

Durch einen Variabelwechsel von $t = \phi(\tau)$ wir das Integral zu

$$X_{\psi(\cdot)}(f) = \int_{-\infty}^{\infty} x(\tau)) \ \sqrt{|\phi'(\phi(\tau))|} \ e^{-j2\pi f\phi(\tau)} dt \ . \tag{4}$$

Mit einer anderen Schreibweise kann Formel 3 mit $\phi(t) = \psi^{-1}(t)$ auch folgendermaßen dargestellt werden:

$$X(f, \phi(\cdot)) = \int_{-\infty}^{\infty} x(t) \ \sqrt{|\phi'(t)|} \ e^{-j2\pi f\phi t} dt \ . \tag{5}$$

Diese Gleichung beschreibt einer inneres Produkt des Signals $x(t)$ mit einem nichtlinearen Chirp. In weiteren Überlegungen wird $\phi(t)$ als Polynom zweiter Ordnung angenommen. In Abbildung 1 werden drei Transformationen mit verschiedenen Zeitverzerrungen gegenübergestellt.

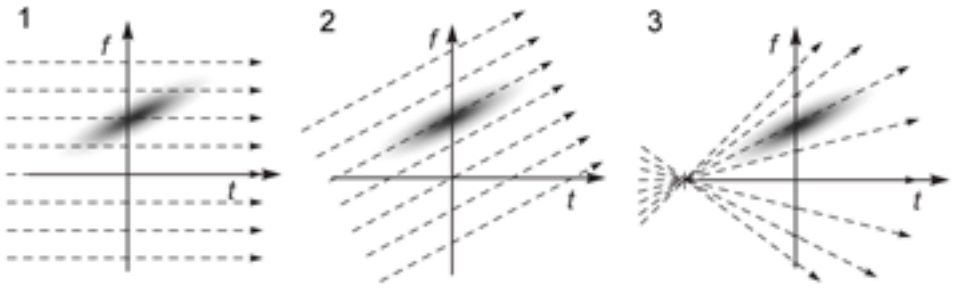


Abbildung 1: Vergleich der Transformationen: (1) Fourier, (2) Chirplet, (3) Fan-Chirp. Der schwarze Bereich repräsentiert die Zeit-Frequenz Energie eines Gausschen Chirplets [3]

# 2 Fan Chirp Transformation

Die Fan Chirp Transformation kann im Ansatz als Zeitfaltung mit anschließender Fourier-Transformation verstanden werden. Die Transformation ist definiert als

$$X(f,\alpha) \triangleq \int_{-\infty}^{\infty} x(t)\ \sqrt{\phi'_\alpha(t)}\ e^{-j2\pi f \phi_\alpha(t)} dt\ , \tag{6}$$

wobei das Polynom $\phi_\alpha(t)$ als *Time Warping* Funktion bezeichnet wird und abhängig von der Chirp Rate $\alpha$ ist:

$$\phi_\alpha(t) = (1 + \frac{1}{2}\alpha\ t)\ t\ . \tag{7}$$

Nach Einsetzten von $\phi_\alpha(t)$ in Formel (6), beinhaltet die FChT das innere Produkt zwischen $x(t)$ und dem komplexen Signalen

$$\xi(t,f,\alpha) = \sqrt{|1+\alpha(t)|}\ e^{-j2\pi f(1+(1/2)\alpha t)t}\ . \tag{8}$$

Die Formel stellt *Chirps* dar, deren momentane Frequenz die Ableitung des Exponent ist und sich linear über die Zeit verändert:

$$f\frac{d\phi_{\alpha(t)}}{dt} = (1+\alpha t)f\ . \tag{9}$$

Das Vorzeichen aller Basiskomponenten ändert sich in dem Moment

$$t = -\frac{1}{\alpha}\ , \tag{10}$$

da hier Formel (9) zu Null wird. Dieser Punkt wird auch als Fokuspunkt (*focal point*) bezeichnet und dessen richtige Wahl ist auch ausschlaggebend für eine gelungene Rekonstruktion des Signals. In den folgenden Überlegungen wir die Chirp Rate $\alpha$ als positiv angenommen.

## 2.1 Rekonstruktion

Das Synthese-Signal $x(t)$ angeschrieben werden mit

$$x(t) = \int_{-\infty}^{\infty} X(f,\alpha)\ \sqrt{|\phi'_\alpha(t)|}\ e^{j2\pi f \phi_\alpha(t)} df\ . \tag{11}$$

Wenn man nun Formel (6) in (11) einsetzt, kommt

$$z(t) = \int_{-\infty}^{\infty} x(\tau)\ \sqrt{|\phi'_\alpha(\tau)\phi'_\alpha(t)|}\ \delta(\phi_\alpha(\tau) - \phi_\alpha(t))\ d\tau\ . \tag{12}$$

Diese Formel kann nach wenigen Schritten mithilfe von Eigenschaften des Dirca-Impulses vereinfacht werden:

$$z(t) = x(t) + x(-t - 2/\alpha)\ . \tag{13}$$

Das wesentliche dieser Synthese Formel besagt, dass das Eingangssignal im Fokuspunkt mit sich selbst überlagert wird. Um $x(t)$ von einer FChT wieder zu rekonstruieren, muss daher laut [3] folgende Bedingung gelten:

$$x(t) = 0 \quad \text{für} \quad t < -\frac{1}{\alpha}\ . \tag{14}$$

In Abbildung 2 wird ein Signal und dessen Synthese gegenübergestellt. Die Chirp-Rate $\alpha = 0,1$, daher liegt der Fokuspunkt bei $t = -10$. Da das Signal vor dem Fokuspunkt ($t < -10$) nicht Null ist, wird das erwünschte Signal ($t > -10$) bei der Rekonstruktion mit sich selbst überlagert. Um diese Artefakte zu vermeiden muss der Fokuspunkt außerhalb der Zeitachse des gewünschten Signals liegen.

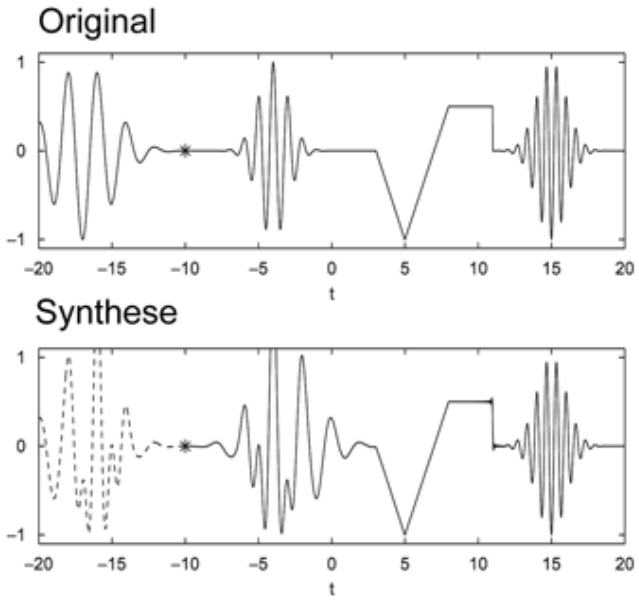


Abbildung 2: Original und rekonstruiertes Signal, Chirp-Rate $\alpha$ = 0,1; Fokuspunkt ($t = -10$) ist durch einen Stern markiert

## 2.2 Berechnung durch zeitliche Vorverzerrung mit FT

Durch die Änderung der Variable $\tau = \phi_\alpha(t)$ wird Analyse-Formel (6) zu:

$$X(f,\alpha) = \int_{-\infty}^{\infty} x(\phi_\alpha^{-1}(\tau))\ e^{-j2\pi f\tau} d\tau\ . \tag{15}$$

Diese Zeitfaltung und anschließende Fourier-Transformation ermöglicht eine effiziente FFT Implementation [4].

Ziel ist es eine Anzahl von linear unabhängigen *Chirps* mit der folgenden Form zu bekommen:

$$x_c(t,f) = e^{j2\pi f\phi_\alpha(t)}\ . \tag{16}$$

Laut [3] repräsentiert nur ein Element dieser Basis den Chirp.

Wenn das Signal gefenstert wird (Fensterfunktion $w(t)$), verliert es die Vollständigkeit und somit ist auch eine ideale Rekonstruktion nicht mehr möglich.

$$X_w(f,\alpha) = \int_{-\infty}^{\infty} x(t)\ w(\phi_\alpha(t))\ \phi'_\alpha\ e^{-j2\pi f\phi_\alpha(t)} dt \tag{17}$$

In [6] wurden die Spektren von *STFT*, *Q-Transformation*[1] und *FChT* gegenübergestellt, wobei mit der letzteren die präzisesten Aussagen über die harmonische Struktur gemacht werden können. Der Grund dafür ist, dass die linearen Chirps mit der Funktion

$$x_hc(t,f_0,L) = \sum_{k=1}^{L} e^{j2\pi f_0\phi_\alpha} \tag{18}$$

alle dieselbe Chirp-Rate $\alpha$ besitzen. Die *Time-Warping* Funktion bildet daher konstante, harmonische Sinus-Komponenten [6].

## 2.3 Eigenschaften

Vor allem die varianten Eigenschaften bzgl. Zeitverschiebung und Zeitskalierung machen die FChT zu einer komplexen Transformation. Als Übersicht werden grundlegende Merkmale vorgestellt:

1. Vorteile der *Q-Transform* sind gute Frequenzauflösung in tiefen Frequenzen und gut Zeitauflösung in hohen Frequenzen. Das wird durch eine veränderliche Fensterbreite erreicht.

| Signal | FChT | Kommentar |
|---|---|---|
| $x^*(t)$ | $X^*(-f, \alpha)$ | |
| $x(t)$ | $X^*(-f, -\alpha)$ | |
| $x(t)$ reell | $X^*(-f, \alpha)$ | |
| $x(t)$ imaginär | $-X^*(-f, \alpha)$ | |
| $ax(t) + by(t)$ | $aX(f, \alpha) + bY(f, \alpha)$ | Linearität |
| $x(t)e^{j2\pi v\phi_\alpha(t)}$ | $X(f - v, \alpha)$ | Chirp-Modulation |
| $\frac{x(t)y(t)}{\sqrt{\lvert 1+\alpha t\rvert}}$ | $X(f, \alpha) * Y(f, \alpha)$ | Fensterung |

Für $\alpha = 0$ können alle Einträge der Tabelle auch der Fourier-Transformation zugeschrieben werden.

# 3 Diskrete Analyse

Wie in Abschnitt 2 beschrieben, kann die *FCht* eines Signals $x(t)$ durch *Fourier-Transformation* eines zeitgefalteten Signals $\tilde{x}(t) = x(\phi_\alpha^{-1}(t))$ berechnet werden.

$$\phi_\alpha{}^{-1}(t) = -\frac{1}{\alpha} + \frac{\sqrt{1 + 2\alpha t}}{\alpha} \tag{19}$$

## 3.1 Reale Sprachsignale

Um harmonische Spektren mit steigender und fallender Grundfrequenz bestmöglich aufzulösen, muss in jedem Segment eine Chirp-Rate $\alpha$ gefunden werden. Es wäre aber auch möglich anhand von *Pitch-Estimation* die Chirp-Rate zu schätzen (z.B. durch Ableitung der Tonhöhe).

Bei polyphonen Signalen reicht ein einzelner Wert für $\alpha$ nicht aus, da sich bei den Harmonischen die Fundamentalfrequenz ($f_0$) im Analyse-Frame ändert. Laut [6] ist ein multidimensionaler Ansatz hier von Vorteil (mehrere *FChT* Instanzen mit verschiedenen Werten für $\alpha$).

Eine einzelne Instanz zeigt für eine spezielle Harmonische ein sehr gutes Frequenzspektrum während die Auflösung für alle anderen Harmonischen schlechter ist. Danach müssen die besten $\alpha$-Werte für jeden Frame ausgewählt werden, zB. mit *Sinusoidal Modeling Techniques* [1]. Die Auswahl der am meisten hervorspringenden Chirp-Rates wird *Pitch Salience* genannt und nun näher erklärt. Eine Möglichkeit besteht darin für jede Fundamentalfrequenz die Summe der Amplituden der Teiltöne zu berechnen.

### 3.1.1 Pitch Salience

Das Ziel dieser Funktion ist einen aussagekräftigen Wert für jede Fundamentalfrequenz zu erhalten. Da jedoch auch an den harmonischen Vielfachen (und Sub) Peaks detektiert werden müssen noch Verbesserungen durchgeführt werden um Mehrdeutigkeit zu vermeiden. Da das menschliche Ohr den Energielevel logarithmisch und nicht linear auflöst, wird die Funktion (*Gathered log-spectrum*) berechnet:

$$\rho_0(f_0) = \frac{1}{n_H} \sum_{i=1}^{n_H} \log |S(if_0)| \ . \tag{20}$$

$|S(f)|$ bezeichnet das *Power-Spectrum* und $n_H$ die Nummer der Harmonischen. Für jede Grundfrequenz wird eine Summe der Teilton-Aplituden berechnet. Das Herausfiltern der falschen Peaks wird durch folgende nicht-lineare Weiterverarbeitung ermöglicht:

$$\rho_1(f_0) = \rho_0(f_0) - \max_{q \in N} \rho_0(\frac{f_0}{q}) \ . \tag{21}$$

Für monophone Signale ist diese Unterdrückung ausreichend, da die Subvielfachen sicherlich eine geringere Amplitude haben als die gewünschte Grundfrequenz. Um auch im polyphonen Fall diese Vielfachen herauszufiltern wird

$$\rho_2(f_0) = \rho_1(f_0) - a_k \rho_1(k f_0) \tag{22}$$

angewendet. Laut [6] genügt es für melodische Inhalte nur den ersten Subvielfachen ($k = 2$) zu löschen. In der Praxis kann es vorkommen, dass eine Grundfrequenz wegen der großen Varianz der Teiltöne in Amplitude und Frequenz nicht richtig detektiert wird. Der Dämpfungsfaktor $a_2$ wird daher nach Erfahrung mit $1/3$ angenommen.

Im letzten Schritt wird $p_2(f_0)$ normalisiert damit die Varianz nicht zu groß wird und es somit nicht zu falschen Detektionen kommen kann.

## 3.2 Schätzung der Chirp rate $\alpha$

Signale mit fächerartigem Spektrum findet man meist nur in kurzen Segmenten von Sprache oder Lauten von Säugetieren.

Entweder sollte man eine Warping-Funktion mit mehreren Freiheitsgraden verwenden um die nicht-lineare Geometrie möglichst gut zu erkennen, oder man unterteilt die Signale in sehr kurze Segmente und analysiert jedes einzelne Segment unabhängig von einander. Die erste Möglichkeit bezieht sich auf einen allgmeinen Warping-Operator und wird in dieser Arbeit nicht näher behandelt. Die zweite Variante bezieht sich auf die FChT als Verallgemeinerung des Spektograms:

$$\xi_x(t, f) = |FChT\{w(\tau)\ x(\tau + t), \alpha(t)\}|^2 \ , \tag{23}$$

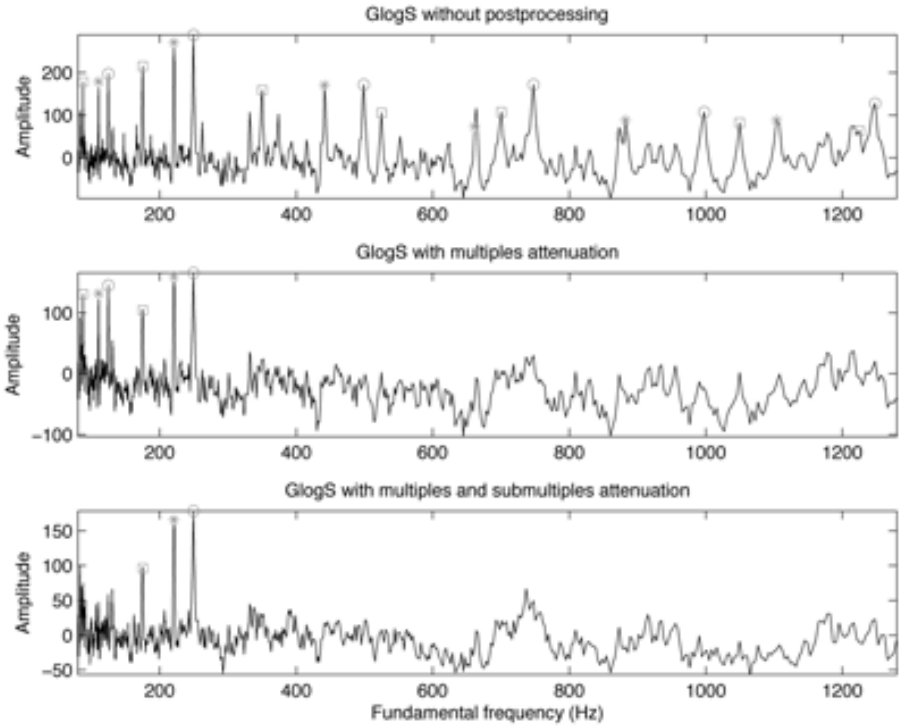


Abbildung 3: *Gathered log spectrum (GlogS):* 3 prominente Stimmen gleichzeitig

$w(t)$ bezeichnet das Analyse-Fenster (Gauss oder Hanning), $\alpha(t)$ die Chirp Rate für das Segment, zentriert auf die Zeit $t$.

Diese segmentweise Verarbeitung hat den Vorteil einer signalverlaufsangepassten Chirp-Rate $\alpha$ (dementsprechend bekommt man die bestmögliche Auflösung für jedes Segement) und wird auch in Echtzeit-Applikationen verwendet. Cross-Term Inteferenzen werden darüber hinaus sehr gut unterdrückt.

Um nun die Chirp Rate $\alpha$ zu schätzen gibt es zwei Methoden.

### 3.2.1 Inter-frame

Ausgehend vom Standpunkt, dass sich das Signal kontinuierlich von seiner Grundfrequenz verändert, kann $\alpha$ mit

$$\alpha(t) = \frac{f_0^{'}(t)}{f_0(t)} \tag{24}$$

geschätzt werden, wobei $f_0^{'}(t)$ die Ableitung von $f_0(t)$ ist. Ziel ist es also die Entwicklung von $f_0(t)$ zu messen um so die Chirp Rate zu erhalten. Im diskreten Fall ($t = nS$) kann die folgende Formel verwendet werden:

$$\alpha[n] = \frac{f_0[n+1] - f_0[n-1]}{2Sf_0[n]} \ , \tag{25}$$

mit dem Verschiebungs-Intervall $S$.

Hier ist jedoch zu beachten, dass die Berechnung des Segments $n$ die Daten des Segments $n+1$ voraussetzt. Mehr Details über diese *nicht-kausale* Methode werden in [5] beschrieben.

### 3.2.2 Intra-frame

In dieser Methode werden zur Berechnung der Daten eines Segments keine Informationen aus anderen Segmenten benötigt. Für die Berechnung wird zunächst ein Diagramm $(\alpha, f)$ herangezogen, mit dem man detaillierte Information über harmonische Zusammenhänge für positive Chirp Rate erhält. Wie in Abbildung 4 zu sehen, hat die vertikale Projektion ein Maximum bei $\alpha \simeq 0,3$. Dieser Wert kann als Pitch Rate angenommen werden.

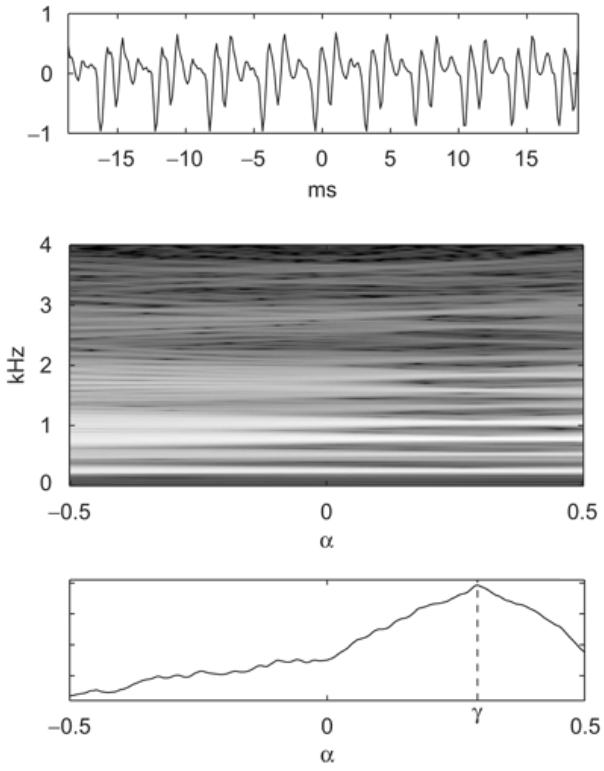


Abbildung 4: Beispiel einer intra-frame Analyse; oben: Analyse-Signal, mitte: $(\alpha, f)$ Diagramm, unten: vertikale Marginalisierung mit Maximum bei $\alpha \simeq 0.3$

Der Nachteil sind jedoch die redundanten Daten des Diagramms. In [2] wird ein Ansatz beschrieben, in dem $L$ verschiedene FChT-Instanzen gleichzeitig berechnet werden und dann danach die Chirplet-Parameter aus den $L$ Instanzen geschätzt werden.

### 3.2.3 Vergleich

Die inter-frame Methode hängt stark von der Präzision der Tonhöhen-Verfolgung (*pitch tracking*) ab. Der Intra-frame Ansatz hängt nicht von zeitlichen Informationen anderer Segmente ab. Obwohl die Berechnung eines $(\alpha, f)$ Diagramms sehr intuitiv erscheint, führt die Redundanz der Daten zu einer großen Rechenlast. Der in [2] beschriebene Ansatz verwendet mehrere FCht-Instanzen und vermeidet aufgrund effizienter Programmierung redundante Information. Trotzdem kann der Algorithmus nicht in Echtzeit verwendet werden.

## 3.3 Constant Q Transformation (CQT)

Wegen der variablen Fensterbreite der *CQT* ist ein grundlegender Vorteil die bessere Frequenzauflösung bei tiefen Frequenzen und gleichzeitig höhere Zeitauflösung bei hohen Frequenzen. Es gibt mehrere verschiedene Implementierungen, die Matlab-Analyse in Abschnitt 3.5 benützt die IIR CQT.

Sprache besteht zu großen Teilen aus nicht-stationären Signalen. Im Gegensatz zur *STFT* werden bei der *CQT* diese höheren Teiltöne nicht so verschmiert dargestellt. Die Chirp-Rate $\alpha$ kann in der Implementierung nur aus diskreten Werten bestehen. Wenn diese Werte nicht genau mit den realen $\alpha$-Werten übereinstimmen, können höhere Teiltöne nach der Ausführung der *Warping-Function* instationär erscheinen und als geräuschhaft interpretiert werden. Die *CQT* mit einem relativ hohen Q-Wert verringert dieses Problem.

## 3.4 Zusammenfassung der digitalen Berechnung

Um eine FChT zu berechnen benötigt man laut [3] vier Grundoperationen:

1. **Normalisierung**: Anzahl der Operationen: 2N
   Das zeitdiskrete Signal $x[n]$ wird mit einem Fenster (von Chirp Rate $\alpha$ abhängig) gewichtet:

$$z[n] = \frac{x[n]}{\sqrt{|1 + \alpha t_n|}} \tag{26}$$

2. **Zeitverzerrung**: Anzahl der Operationen: M
   Das erhaltene Signal muss auf die neue Zeitindizes $\psi_\alpha(t)$ umgetastet werden.

$$\tau_n = \psi_\alpha(\hat{t}_n) \tag{27}$$

3. **Umtastung des Signals**: Anzahl der Operationen: 4M

$$\hat{z}[n] = \sum_l z[l]\ h[t_l - \tau_n] \tag{28}$$

4. **DFT Berechnung** Anzahl der Operationen: M $log$ M (für $M = 2^q, q \in N$)
   Die nachträgliche DFT Berechnung des gesampelten Signals schließt die FChT ab:

$$X[k, \alpha] = DFT\{\hat{z}[n]\} \tag{29}$$

Für die inverse Berechnung lauten die Formeln folgendermaßen:

1. **iDFT**:

$$\hat{z}[n] = iDFT\{X[k, \alpha]\} \tag{30}$$

2. **Zeitverzerrung**:

$$\hat{\tau}_n = \phi_\alpha(t_n) \tag{31}$$

3. **Umtastung des Signals**:

$$\hat{z}[n] = \sum_l \hat{z}[l]\ h[\hat{t}_l - \hat{\tau}_n] \tag{32}$$

4. **Normalisierung**:

$$y[n] = \sqrt{|1 + \alpha t_n|}\ z[n] \tag{33}$$

Die gesamte Berechnung einer FChT beträgt etwa $(logN + 7)N$. Es wird jedoch darauf hingewiesen, dass die Schätzung von der Chirp-Rate $\alpha$ hier nicht in den Berechnungen mit eingeflossen ist, da es in der Implementierung als externer Prozess programmiert vorliegt.

In Abbildung 5 wird ersichtlich, dass die FChT in der Lage ist, für alle Harmonischen die bestmögliche Auflösung zu erzielen. Die anderen drei Transformationen konnten die Auflösung nur für den selektierten Oberton verbessern. Zu erwähnen ist, dass FT und FChT bei reellen Signalen immer ein symmetrisches Spektrum liefern.

## 3.5 Effiziente Analyse in MATLAB

In [6] wird ein schnelles Analysetool in Matlab vorgestellt, mit der auch komplette Ordner mit Sprachdateien z.B. aus der MIREX [2] oder RWC [3] Datenbank analysiert werden können. Es besteht aus C- und Matlab-Funktionen. Der Quellcode steht im Internet [4] zur Verfügung.

Darüber hinaus wurde in der Implementierung auch eine Kombination von FChT mit *Constant-Q-Transformation* vorgestellt um eine noch bessere Auflösung bei harmonischen Signalen mit nicht-linearen Tonhöhenveränderungen zu erzielen.

2. Music Information Retrieval Evaluation eXchange
3. Popular Music Database
4. `http://iie.fing.edu.uy/~pcancela/fcht/`

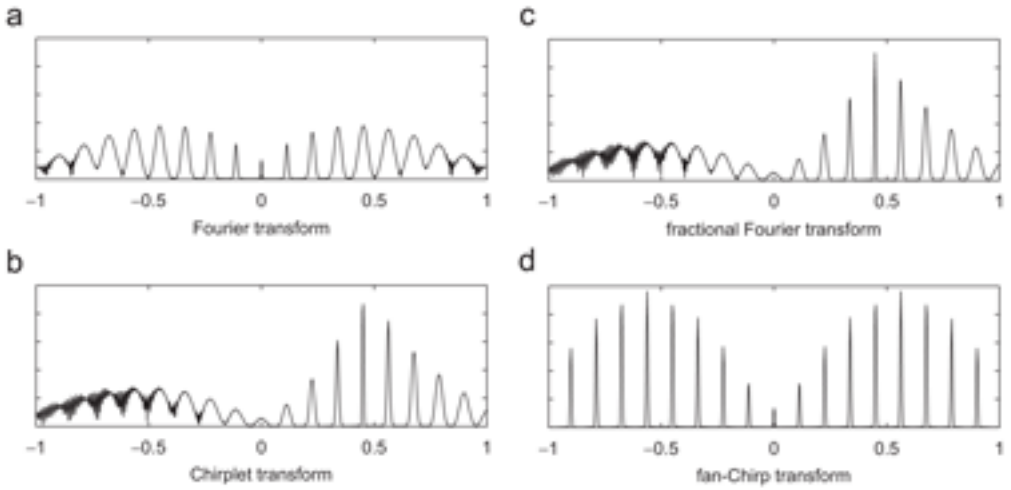


Abbildung 5: Vergleich zwischen den Transformationen; Analyse eines synthetischen Signals, bestehend aus linear modulierten Grundtönen

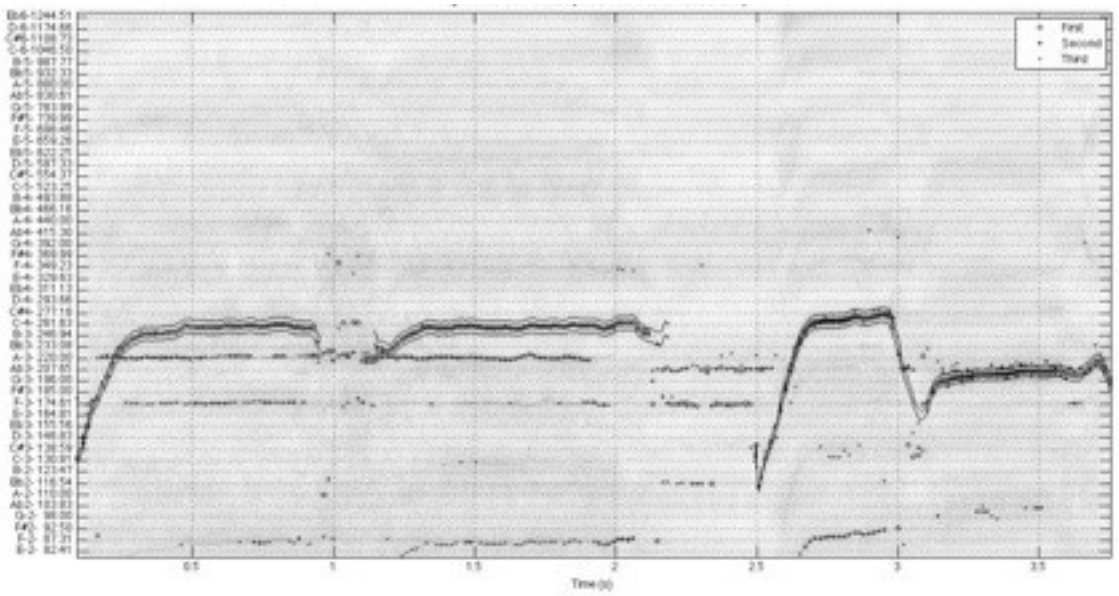

Abbildung 6: F0gram in Matlab eines 4 sek Sample "pop1_long.wav": zeitlicher Verlauf der Tonhöhen prominentester Grundfrequenzen

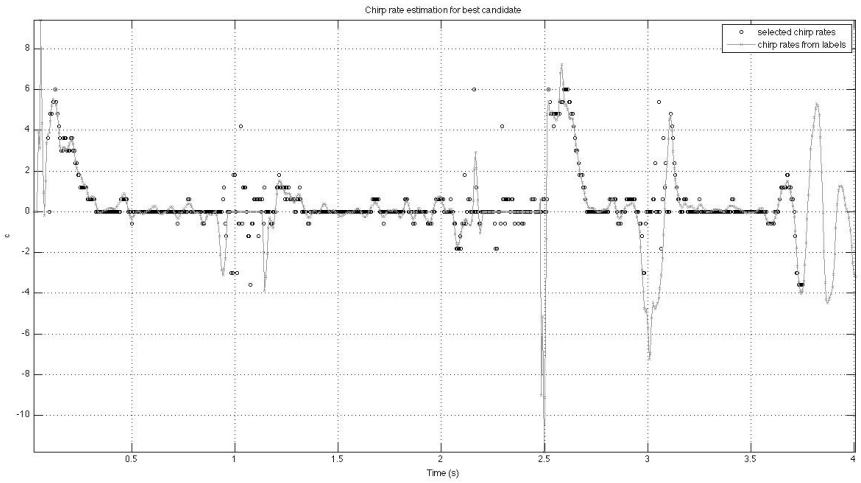


Abbildung 7: Bewertung der Chirp Rate in Matlab

Um gleichzeitig verschiedene Tonhöhen aus einem Signal herauszufiltern, wurde eine Methode anhand *Pitch salience* angewandt und verbessert, um polyphone Musik zu detektieren. In den Abbildungen 6 und 7 sieht man die grafische Ausgabe der Berechnung.

Ein Nachteil dieser Implementierung ist, dass die Phase nicht korrekt berechnet wird. In [7] wird auf dieses Problem eingegangen und ein Ansatz beschrieben. Weiters wird das Eingangssignal wegen der Zeitfaltung interpoliert. Die Wahl der Interpolation beeinflusst jedoch Geschwindigkeit und gleichzeitig die Genauigkeit der Transformation.

# 4 Synthese in MATLAB

In diesem Abschnitt wird eine selbst erstellte Implementierung vorgestellt.

Ausgehend von den bereitgestellten Daten der Matlab-Implementierung sollte ein Beispiel-Signal wiederhergestellt werden. Anhand einer Matrix mit Grundfrequenz- und dazugehörige Zeitwerten kann theoretisch ein Signal synthetisiert werden.

Zuerst wurden der Einfachheit wegen die einzeln detektieren Frequenzen ohne Phaseninterpolation abgespielt. Dazu wurden die ersten 3 wichtigsten Grundfrequenzen ausgewählt. Natürlich sind aufgrund Phasenfehler beim Abhören des Signals die Artefakte dominanter als das Signal selbst.

In der ersten Verfeinerung wurden die Momentanfrequenzender generierten Grundtonspuren beim Abspielen linear interpoliert.

Als zweite Verfeinerung wurden weitere Phasenfehler behoben. Die experimentelle Implementierung speichert die Phase im aktuellen Segment und beginnt das Segment danach beim vorausberechneten Phasenwert.

**Wave-Shaping** In der Analyse-Matrix sind nur die verschiedenen Grundtöne pro Zeiteinheit gespeichert. Um jedoch eine passende Synthese zu erzeugen sollte man auf die Obertöne nicht verzichten, da vor allem die ersten Obertöne zum Klangbild beitragen.

Die Wellenformsynthese *Wave-Shaping* fügt dem Eingangssignal durch nichtlineare Verzerrung Obertöne hinzu. Als Verzerrungsfunktion verwendet man *Chebyshev-Polynome*, wenn harmonische Obertöne gezielt erzeugt werden sollen.

Die ersten *Chebyshev-Polynome* sind folgendermaßen aufgebaut:

$$\begin{aligned} T_0(x) &= 1 \ , \\ T_1(x) &= x \ , \\ T_2(x) &= 2x^2 - 1 \ , \\ T_3(x) &= 4x^3 - 3x \ , \\ T_4(x) &= 8x^4 - 8x^2 + 1 \ . \end{aligned} \tag{34}$$

Diese einzelnen Terme werden durch die rekursive Formel hergeleitet:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x) \ . \tag{35}$$

Wenn man nun $x$ durch die Schwingung $cos\ wt$ ersetzt, erzeugt ein Chebyshev-Polynom einen bestimmten Oberton davon:

$$T_k\ cos(x) = cos(kx) \ . \tag{36}$$

In unserem Fall wird $x$ durch den erzeugten Grundtonverlauf (ohne Amplitude) ersetzt. Um nun einen gewünschten Obertonklang zu erzeugen, können die einzelnen Terme addiert werden. Die Obertöne 1,4,9 im Ampituden-Verhältnis 3:1,5:0,1 ergeben sich aus dieser Formel:

$$3\ T_1(x) + 1,5\ T_4(x) + 0,1\ T_9(x) \ , \tag{37}$$

und wurden zum Experimentieren mit Klangbeispielen verwendet.

# 5 Zusammenfassung

In dieser Arbeit wurden die Grundlagen und Eigenschaften der FChT besprochen. Die Transformation ist sehr gut geeignet für Signale mit fächerartigem Spektrum, wie alle harmonisch zusammengesetzte Klänge mit gemeinsam modulierten Grundton.

Die fächerartige Geometrie wird durch die Chirp Rate $\alpha$ oder deren Inverse, den Fokuspunkt, bestimmt. In der Praxis sollte das Signal in diesem Bereich 0 sein um bei der Rekonstruktion keine Artefakte zu bekommen.

Da das Ergebnis der FChT sehr stark von $\alpha$ abhängt, ist ein wesentlicher Teil die optimale Vorhersage und Schätzung dieses Wertes. Für polyphone Signale müssen für jedes Analyse-Segment mehrere $\alpha$-Werte ausgewählt werden, die den Klang am besten repräsentieren.

Derzeit wird bei Sprachanalyse oft die Bandbreite in zwei Teile unterteilt und diese getrennt betrachtet: tiefe Frequenzen werden als stimmhaft angesehen und hohe als stimmlos. Mit der FChT ist es allerdings möglich den gesamten spektralen Bereich als harmonische Struktur zu analysieren. Darüber hinaus werden *Crossterm-Inteferenzen* weitestgehend unterdrückt.

Ein Nachteil ist hingegen die starre Frequenz-Auflösung in Folge der konstanten Fensterbreite. Weiters wird bei stimmlosen Segmenten kein zuberlässiger Wert ermittelt.

Die experimentelle Implementierung konnte bei einem Klangbeispiele mit drei singenden Männerstimmen jede einzelne Stimme gut wiedergeben. Bei gleichzeitigem Abspielen der drei Stimmen wurden jedoch Artefakte hörbar.

# Literatur

[1] M. Bartkowiak. Application of the fan-chirp transform to hybrid sinusoidal+noise modeling of polyphonic audio. In *16th European Signal Processing Confernce*, 2008.

[2] L. Weruga, M. Kepesi. Self-organizing chirp-sensitive artificial auditory cortical model. In *Proceedings of Interspeech*, 2005.

[3] Luis Weruaga, Marian Képesi. The fan-chirp transform for non stationary harmonic signals. In *Signal Processing 87*, 2007.

[4] Luis Weruga, Milan Sigmund. Time-frequency analysis for voice activity detection. In *Speech Comm*, 2007.

[5] M. Kepesi, L. Weruga. Adaptive chirp-based time frequency analysis of speech signals. In *Speech Commun. 48*, 2006.

[6] Pablo Cancela, Ernesto López, Martin Rocamora. Fan chirp transform for music representation. In *International Conference on Digital Audio Effects, 13th. DAFx-10. Graz, Austria*, 6-10 Sep 2010.

[7] Robert B. Dunn, Thomas F. Quatierei, Nicolas Malyska. Sinewave parameter estimation using the fast fach-chirp transform. In *Speech Com.*, 2010.