ISSN 1688-2806

Universidad de la República
Facultad de Ingeniería

Transient and steady-state component separation for audio signals

Tesis presentada a la Facultad de Ingeniería de la Universidad de la República por

Ignacio Irigaray Bayarres

en cumplimiento parcial de los requerimientos para la obtención del título de Magister en Ingeniería Eléctrica.

Director de Tesis

Dr. Luiz W. P. Biscainho	Universidad Federal de Rio de Janeiro

Tribunal

Dr. Federico Lecumberry	Universidad de la República
Dr. Pablo Belzarena	Universidad de la República
Dr. Álvaro Pardo (Revisor externo)	Universidad Católica del Uruguay

Director Académico

Dr. Pablo Monzón	Universidad de la República

Montevideo
2 de octubre de 2014

Transient and steady-state component separation for audio signals, Ignacio Irigaray Bayarres

ISSN 1688-2806

Esta tesis fue preparada en LaTeX usando la clase iietesis (v1.0).
Contiene un total de 88 páginas.
Compilada el Monday 9th February, 2015.
http://iie.fing.edu.uy/

Whitout analysis, we can never have a synthesis.

Hans-Joachim Koellreutter

Se me ocurre que la verdad profunda de las cosas es necesariamente difusa, imprecisa, inexacta; que el espíritu se alimenta del misterio y huye y se disuelve cuando lo que llamamos precisión o realidad intenta fijar las cosas en una forma determinada - o en un concepto.

Mario Levrero - Desplazamientos

Aside from weighty technical problem of collective coherent thinking, there is a very human, even social need for sympathy from all the members to bend for the common result.

Bill Evans - Improvisation in Jazz

Abstract

In this work the problem of transient and steady-state component separation of an audio signal was addressed. In particular, a recently proposed method for separation of transient and steady-state components based on the median filter was investigated. For a better understanding of the processes involved, a modification of the filtering stage of the algorithm was proposed. This modification was evaluated subjectively by listening tests and objectively by an application-based comparison. Also some extensions to the model were presented in conjunction with different possible applications for the transient and steady-state decomposition in the area of audio editing and processing.

Esta tesis trata sobre la separación de señales de audio en componentes transitorios y estacionarios. En particular, se estudia un método reciente para la separación en componentes transitorios y estacionarios basado en la utilización de filtros de mediana. Para una mejor comprensión de los procesos involucrados, se propone una modificación a la etapa de filtrado. La modificación propuesta es evaluada de forma subjetiva por medio de test de escucha y objetivamente mediante la comparación de los resultado de algunas aplicaciones. Además, se presentan extensiones al modelo en conjunto con diferentes aplicaciones en el área de la edición y procesamiento de audio.

Contents

Abstract **iii**

1 Introduction **1**
1.1 Historical context 2
1.2 Motivation 3
1.3 Thesis outline 4

2 Audio signal representation **5**
2.1 Introduction 6
2.2 Time-domain representations 6
2.3 Frequency-domain representation 6
2.4 Time-Frequency representation 7
2.4.1 Uncertainly principle 8
2.4.2 Short-Time Fourier Transform 9
2.4.3 Constant-Q Transform 12
2.4.4 Sinusoidal Modeling 13
2.4.5 Non-Negative Matrix Factorization NMF 18

3 Transient and Steady-State separation **21**
3.1 Model of transient and steady-state components 22
3.2 Separation Methods 22
3.2.1 Median Filter 24
3.2.2 SSE Filter 25
3.3 Extensions 28
3.3.1 Iterative Filtering 28
3.3.2 Relaxed components 29
3.3.3 Sub-Band processing 30
3.4 Reconstruction method 31

4 Tests & Applications **35**
4.1 SSE and Median filter comparison 36
4.1.1 Data set description 36
4.1.2 Tests 37
4.1.3 Beat Tracking 37
4.1.4 Pitch-tracking 40

4.2 Applications in audio editing 44
4.2.1 Removing undesired transients 44
4.2.2 Percussion extraction and remixing 44
4.2.3 Noise Reduction 45
4.2.4 Transient shaping 46
4.2.5 De-reverberation 47
4.2.6 Time/Pitch Modifications 48

5 Conclusions and future work **51**
5.1 Conclusions 52
5.2 Future perspectives 53

Appendix **54**

A Subjective Quality Test Data **55**

B Subjective Quality Test Interface **61**

C Complete Beat-Tracking results **63**

Bibliography **65**

List of Figures **74**

Índice de figuras **76**

Chapter 1

Introduction

This thesis is about sounds and music. Music is one of the most sophisticated forms of communication ever created by humans. It finds the more diverse applications, such as: social control, entertainment, religious rituals, marketing, and aesthetic pleasure, among others. Therefore, its detailed study under all the available human knowledge becomes a worthwhile effort to undertake. The following section attempts to illustrate the relation between music and technology, and the importance of the collaboration among musicians and engineers. This thesis is an effort in this direction.

1.1 Historical context

From the beginning the humans are trying to understand and explain the sounds. The Marin Mersenne observation in [53] is worth quoting:

> "*The string struck and sounded freely makes at least five sounds at the same time, the first of which is the natural sound of the string and serves as the foundation for the rest.*"

Marsenne could not accept that the movement of one string could produce more than one sound. Joseph Sauveur finds the explanation for overtones [70], which he poetically resumes as:

> "*When a point of a calm surface of water is slightly agitated, circular waves are formed and spread around it. When the surface is agitated at another point, new waves are formed and mix with the former; they travel over the surface disturbed by the first wave as they would do over a calm surface, so that they can be perfectly distinguished in the mixture. What the eye perceives with respect to waves, the ear perceives with respect to sounds or aerial vibrations, which travel simultaneously without troubling each other and produce very distinct impressions...*"

In the year of 1807 Joseph Fourier releases his memories "On the Propagation of Heat in Solid Bodies", where he presents the theory of the Fourier Series. Most of the works in audio signal processing are based on this foundational result, and this thesis work is not the exception.[1]

Jumping in time to the early 20th century, the first electronic musical instruments were developed. The Dynamophone, also known as Thelarmonium, was presented before the public in 1906: it weighted 200 tons and could delivery electronic music through the telephone network. Approximately at the same time the musicians perceived the necessity of new instruments to express themselves. In 1913 the Italian composer Luigi Russolo wrote [68]:

[1] This section is based [11], a complete reading of which is highly recommended.

> "*Musical sound is too limited in qualitative variety of timbre. The most complicated orchestra reduces themselves to four or five classes of instruments differing in timbre: instrument played with the bow, plucked instruments, brass-winds, wood-winds and percussion instruments... we must break out of this narrow circle of pure musical sound and conquer the infinite variety of noise sounds,*"

and in 1916 the composer Edgar Varèse stated:

> "*Our musical alphabet must be enriched... We also need new instruments very badly... In my own works I have always felt the need for new mediums of expression.*"

The technological advances after the World War II[2] inspired two different aestethic movements, the musique concrète in Paris and the elektronische Musik in Cologne—the first exploring the creative possibilities of the tape recorder while the second the synthetic generation of sound by the utilization of electronic oscillators. The new mediums of expression that Varèse advocated where created. In brief, as stated in [10]:

> "*Instrument and music can only develop together - and not as they please, but according to a concurrence: potential in the instrument, and need, in the player.*"

The popularization of computers in universities in the sixties and the technological revolution brought by the introduction of the microprocessor in the seventies were the foundation of software-synthesized music. In the following decades, with the massification of personal computers, it became an indispensable tool in every studio. The personal computer and the software that runs on it are the instruments of today.

1.2 Motivation

Audio signal modeling has now decades of development and involves a vast literature. To increment knowledge of the state of the art in this discipline is one of the principal motivations of this thesis. In particular, the transient and steady-state component separation problem is addressed. Along this work, transient components are considered as broad-band, with highly concentrated energy in time, whereas steady-state components are considered as discrete, narrow-band, with smooth temporal behaviour. Such components connect with musical concepts such as beat and pitch, and some of these relations are explored in this work. Also, existing sound processing techniques can benefit from the utilization of this kind of decomposition. For instance, the generation of artifacts can be reduced in noise-reduction applications and transients smearing can be avoided in time-scale

[2] And in particular, the end of the war.

modifications. Finally, the separation makes possible the precise control of each component separately, which enables artistic applications for musicians and composers. To explore applications of the separation techniques proposed is another motivation of the present work.

1.3 Thesis outline

This document follows with an introduction of the principal concepts associated with time, frequency, and their relations in the signal processing context. Time, frequency, and joint time-frequency representations are presented and their limitations discussed in Chapter 2. Some of these representations are the foundations on which the transient and steady-state component separation relies. Next, in Chapter 3 the transient/steady-state models are defined, an algorithm based on median filters is described, and a modification of its nonlinear filtering stage is proposed. The Chapter 4 has two principal sections: first, comparative subjective and objective evaluations between the proposed modification and the original algorithm are conducted and described. In the second part, various applications of the transient/steady-state decomposition are presented and illustrated with examples. Finally, the document ends with some conclusions and ideas for future work in Chapter 5.

Chapter 2

Audio signal representation

2.1 Introduction

This chapter describes some of the principal concepts associated with time, frequency and their relations in the signal processing context. Time, frequency and joint time-frequency representation are presented and their limitations discussed.

2.2 Time-domain representations

Signals have various definitions: in the information theory context signal is defined as an entity that carries information, in electrical engineering a signal is a measurable quantity that varies in time. In particular, sound signals represent the variation of the air pressure along time. Time representation of audio signals is the most intuitive, simple and common.

Although simple, useful information can be derived from the time-domain representation.

Given a signal $s(t)$ its total energy can be defined as:

$$E = \int_{-\infty}^{\infty} |s(t)|^2 dt. \tag{2.1}$$

One can considerate $|s(t)|^2$ as its instantaneous power, or density of energy expressed along time. Thus the average time, that represents the time value around which this density is distributed, can be defined as follows:

$$< t >= \int_{-\infty}^{\infty} t|s(t)|^2 dt. \tag{2.2}$$

In a similar way the standard deviation σ_t, which represents the effective duration of the signal, can be calculated as:

$$T^2 = \sigma_t^2 = \int_{-\infty}^{\infty} (< t > -t)^2 |s(t)|^2 dt. \tag{2.3}$$

In the upper plot of figure 2.1, the time evolution of the amplitude of an audio signal is shown. More precisely it is a male voice singing the English word “so”.

2.3 Frequency-domain representation

The Fourier Analysis is one of the major tools of Mathematics and Physics. It plays a key role in all areas of knowledge in which periodic phenomena take place (Acoustics, Optics, Geophysics, Economy, etc.). Frequency representation often reveals useful information for the comprehension of the underlying processes.

In equation 2.4, the Fourier Transform is applied to a signal $s(t)$, and in 2.5 the inverse Fourier Transform is shown:

$$\mathcal{F}[s(t)](\omega) = \int_{-\infty}^{\infty} s(t) e^{-j\omega t} dt, \tag{2.4}$$

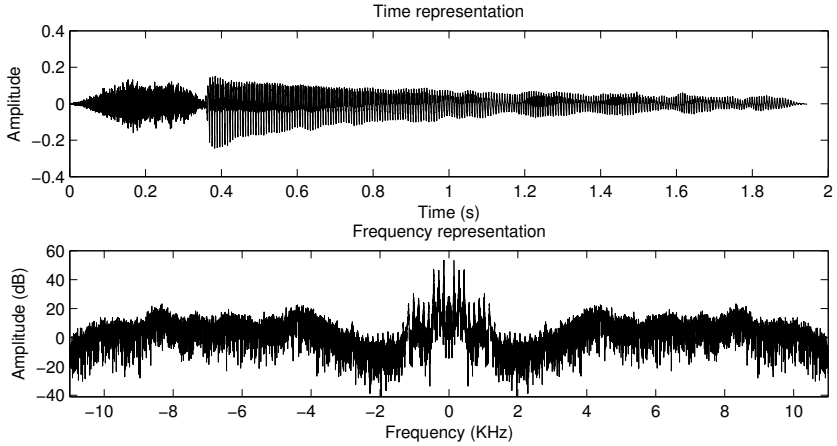


Figure 2.1: Time and frequency representation of an audio signal. A male voice singing "so". Starts with the unvoiced consonant [s] and ends with the voiced vowel [o].

$$s(t) = \bar{\mathcal{F}}[S(\omega)](t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)e^{j\omega t}d\omega. \tag{2.5}$$

The signal $s(t)$ is decomposed in complex exponentials ($e^{j\omega t}$) of infinite duration and frequency ω, each one contributing a relative amount indicated by $S(\omega)$.

One can interpret $|S(\omega)|^2$ as the energy per unit frequency at frequency ω, and then derive via Parseval's theorem the total energy:

$$E = \int_{-\infty}^{\infty} |S(\omega)|^2 d\omega = \int_{-\infty}^{\infty} |s(t)|^2 dt. \tag{2.6}$$

If $|S(\omega)|^2$ represents the density, the averages can be calculated as was done in the time domain. The average frequency represents the frequency around which the energy is distributed, and is defined as:

$$< \omega >= \int_{-\infty}^{\infty} \omega |S(\omega)|^2 d\omega. \tag{2.7}$$

The standard deviation σ_ω, often called the root mean squared bandwidth (denoted B), is calculated as:

$$B^2 = \sigma_\omega^2 = \int_{-\infty}^{\infty} (< \omega > -\omega)^2 |S(\omega)|^2 d\omega. \tag{2.8}$$

The figure 2.1 depicts the module in decibels of the Fourier Transform (lower) and their correspondent time-domain signal (top).

2.4 Time-Frequency representation

Time-Frequency Representations are one of the most important tools in audio signal processing. The historical development of the theory was driven by various disciplines, such as Mathematics, Quantum Mechanics and Signal Processing.

2.4.1 Uncertainly principle

In signal analysis, the Heisenberg-Gabor uncertainly principle refers to the impossibility of signals to be arbitrarily concentrated in time and frequency at the same time. This result is very important because it imposes a theoretical limit to the resolution of time-frequency representations. The original formulation was done in the 1920s by Werner Heisenberg in the context of Quantum Mechanics, and by Dennis Gabor after World War II in the Communication Theory field.

The demonstration assumes a signal with zero mean (both in frequency and time) for simplicity, without loss of generality:

$$s_{\text{new}}(t) = e^{-j<\omega>(t+<t>)} s_{\text{old}}(t+ < t >). \tag{2.9}$$

Equations 2.3 and 2.8 give an expression for the standard deviation of energy in time and in frequency respectively.

$$T^2 = \sigma_t^2 = \int_{-\infty}^{\infty} (< t > -t)^2 |s(t)|^2 dt = \int_{-\infty}^{\infty} t^2 |s(t)|^2 dt, \tag{2.3}$$

$$B^2 = \sigma_\omega^2 = \int_{-\infty}^{\infty} (< \omega > -\omega)^2 |S(\omega)|^2 d\omega = \sigma_\omega^2 = \int_{-\infty}^{\infty} \omega^2 |S(\omega)|^2 d\omega = \int_{-\infty}^{\infty} |s'(t)^2| dt. \tag{2.8}$$

The product of duration (T) and bandwidth (B) is:

$$B^2 T^2 = \sigma_t^2 \sigma_\omega^2 = \int_{-\infty}^{\infty} t^2 |s(t)|^2 dt \int_{-\infty}^{\infty} \omega^2 |S(\omega)|^2 d\omega. \tag{2.10}$$

Applying the Schwarz inequality:

$$B^2 T^2 = \int_{-\infty}^{\infty} |ts(t)|^2 dt \int_{-\infty}^{\infty} |s'(t)|^2 dt \geq \left| \int_{-\infty}^{\infty} ts^*(t) s'(t) dt \right|^2 = |I|^2. \tag{2.11}$$

Integrating by parts, and recalling equation 2.1, the integral I can be expressed as:

$$I = t|s(t)|^2|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} |s(t)|^2 dt - \int_{-\infty}^{\infty} ts(t) s'^*(t) = t|s(t)|^2|_{-\infty}^{\infty} - E - I^*. \tag{2.12}$$

Supposing that $x(t)$ decays fast enough to assure that $t|x(t)|^2$ vanishes at infinity (which is satisfied if $x(t)$ has compact support)

$$|I| \geq Re(I) = \frac{E}{2} \tag{2.13}$$

Considering normalized signals ($E = 1$), and combining equations 2.11 and 2.13 the Heisenberg-Gabor uncertainly principle is

$$TB \geq \frac{1}{2} \tag{2.14}$$

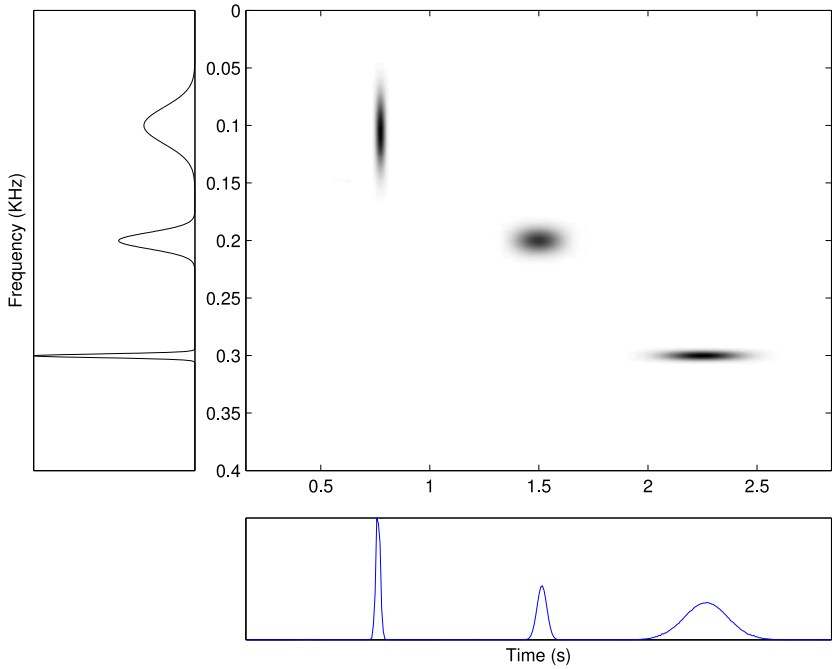


Figure 2.2: Representation of three Gaussian sinusoidal atoms: time, frequency and joint time frequency representations.

In the figure 2.2 the energy distribution in the T-F plane of three different atoms is shown. The Heisenberg-Gabor uncertainly principle is not the only possible approach to mathematically describe the Fourier duality, the Slepian-Pollak-Landau [76] theory addresses the restriction of energy in compact support in time and frequency.

2.4.2 Short-Time Fourier Transform

In the introduction of his fundamental article "Theorie et Applications de la Notion de Signal Analytique" Jean-Andre Ville illustrates the drawbacks of the Fourier analysis in an unbeatable manner:

> "If we consider a fragment containing many measures (which is the least we can demand), and one note, *la* for example, appears once in the fragment, the harmonic analysis will present us with the amplitude and the phase of the corresponding frequency, without locating the *la* in time. Now then, it is obvious that in the course of the fragment there will be instants when the *la* will not be heard. Nevertheless, the representation is mathematically correct, because the phases of the notes near *la* acts to destroy this note by interference when *la* is not heard, and to reinforce it, also by interference, when it is heard; but if there exists in this concept a cleverness which does honor to

> mathematical analysis, there is also a distortion of reality: in fact, when *la* is not heard, the true reason is that the *la* is not emitted."

As seen in section 2.3, the Fourier Transform decompose signals as a combination of sinusoids that last forever in time. Music never presents a strictly periodic behaviour, by contrast its essence is the irregular variation over time. Although irregular, a lot of musical signals observed locally present a pseudo-periodic behaviour.

The Short-Time Fourier Transform (STFT) represents signals in the form of a time-frequency map commonly called spectrogram.

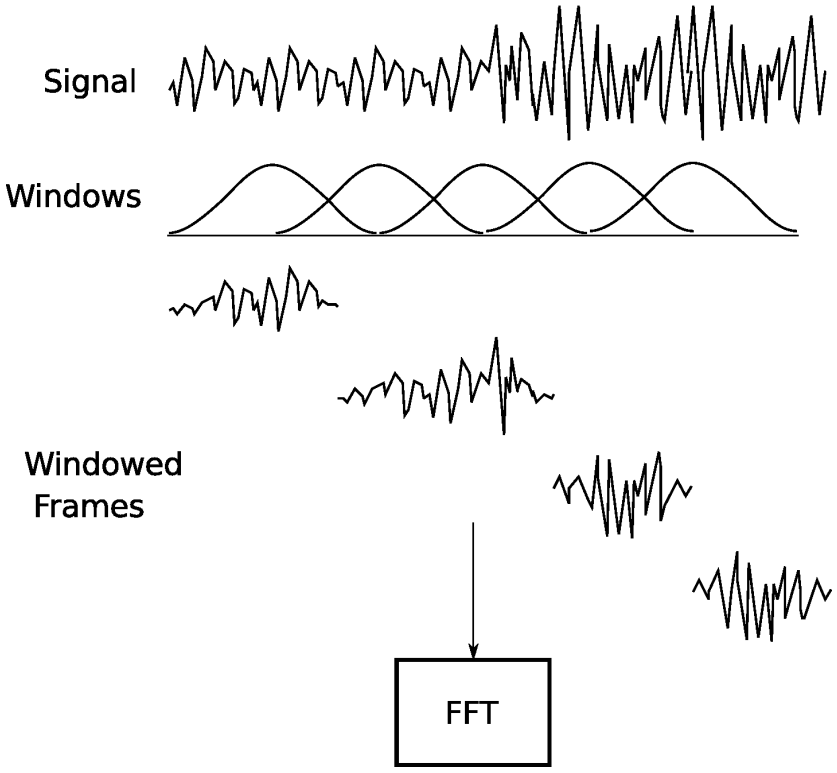


Figure 2.3: This diagram summarises the Short-Time Fourier Transform calculation.

The definition of the continuous-time STFT of a function $x(t) \in L^2(\mathbb{R})$ with a given window $g(t)$ with unit area is:

$$\mathrm{STFT}_x^g(t,\omega) = \int_{-\infty}^{\infty} x(t')g(t'-t)e^{j\omega t'}dt'. \tag{2.15}$$

The STFT of the time-dependant signal $x(t)$ is a linear transformation that depends on the chosen window $g(t)$.

Analysis window

The analysis window $g(t)$ is generally an even function with positive real values concentrated at time zero, and its Fourier Transform also has its maximum at zero

frequency. The analysis window leaves unchanged the signal value at some instant t' whereas attenuates the signal at distant times. One is looking at an excerpt of the entire signal, as is done with a landscape and a real window.

The STFT can be considered as a projection of the signal $x(t)$ into a family of atoms generated by time translations and frequency modulations of a given window $g(t)$:

$$g_{t,\omega}(t') = g(t' - t)e^{-j\omega t'}. \tag{2.16}$$

If $g(t)$ is an even, real-valued function, atom's energy is concentrated in t, ω. Figure 2.2 shows three of such atoms with different time and frequency centres, and energy concentrations in the T-F plane.

Energy Conservation

The energy E_x of the signal can be calculated by integrating the STFT:

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{STFT}^2(t, f) df\, dt. \tag{2.17}$$

This allows one to interpret the STFT as the distribution along the T-F plane of the density of energy. The demonstration of this property can be consulted in [29].

STFT Synthesis

The reconstruction formula given a STFT_x^g of a signal x is:

$$x(t') = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{STFT}_x^g(t, \omega)e^{j\omega t'} d\omega dt. \tag{2.18}$$

Discrete Fourier Transform

In practice, the spectral analysis for audio signals is done in the discrete time domain. In this domain, the STFT is defined as:

$$X_n[e^{j\omega_k}] = \sum_{m=-\infty}^{\infty} w[n - m]x[m]e^{-j\omega_k m}. \tag{2.19}$$

Equation 2.19 has two equivalent interpretations: the overlapp-add (OLA) and the filter bank summation (FBS).

The OLA interpretation consist in considering $X_n[e^{j\omega_k}]$ as a function of n, in such a way that the STFT represents the Fourier transform of the moving signal centered (and windowed) at time n.

In the filter bank summation interpretation, the signal x[m] is first frequency shifted by $e^{-j\omega_k m}$ so that the frequency ω_k is moved to zero, and then low-pass filtered by a filter with impulse response $w[n]$.

2.4.3 Constant-Q Transform

A representation with fixed frequency resolution (such as the STFT) has some limitations when applied to music related signals. For example, if one considers the register of a standard 88 key piano[1], the first semitone is 1.6352-Hz wide, while the last one is 235-Hz wide. To discriminate two notes whose fundamental frequencies are separated by one semitone, a window with more than 27000-sample [2] duration is needed. When using a constant resolution representation its results in oversampling at high frequencies. In 1991 Judith Brown presented in [4] a constant Q spectral transform, where Q is the ratio of the center frequency to the bandwidth of each frequency bin.

The quality factor Q for a quarter-tone resolution is calculated as:

$$Q = \frac{f}{\delta f} = \frac{f}{(\sqrt[24]{2} - 1)f} \approx 34, \tag{2.20}$$

$$N[k] = \frac{S}{\delta f_k} = \frac{S}{f_k} Q. \tag{2.21}$$

The frequency resolution is proportional to the windows size, so a variable resolution representation must have variable windows lengths. In the constant Q transform (CQT), the k-th frequency component is $\frac{2\pi Q}{N[k]}$, and the window length is determined by N[k].

The constant Q transform is defined as:

$$X[k] = \frac{1}{N[k]} \sum_{n=0}^{N[k]-1} W[k,n] x[n] e^{\frac{-j2\pi Q n}{N[k]}}. \tag{2.22}$$

In this case of quarter-tone resolution, the window lengths became $N[k] = \frac{N_{max}}{(2^{1/24})^k}$. Time-frequency plane tiling for the CQT and two STFT with different resolutions are illustrated in Figure 2.4.

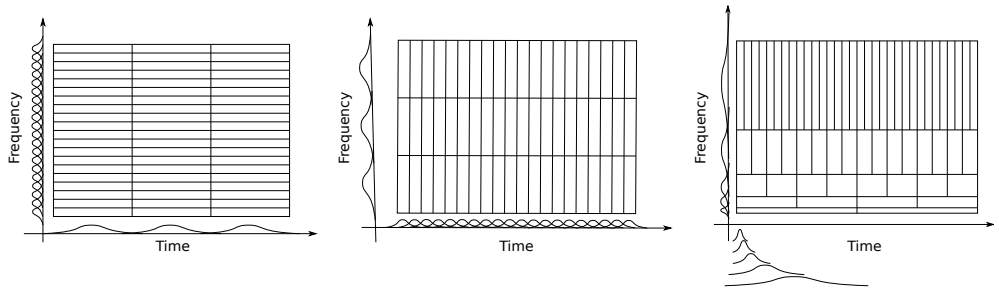


Figure 2.4: Tiling of the time-frequency plane: STFT (left and center) and CQT (right).

Figure 2.5 depicts two spectrograms, one calculated using the STFT and the other with the CQT. The CQT keeps a good compromise between time and frequency resolution along the spectrum: higher resolution at low frequencies at the

[1] The piano register extends from A_0 (27.5 Hz) to C_8 (4186.01 Hz).

[2] Considering a sampling rate of 44100 samples per second.

cost of time resolution, and higher resolution at high frequencies at the cost of frequency resolution.

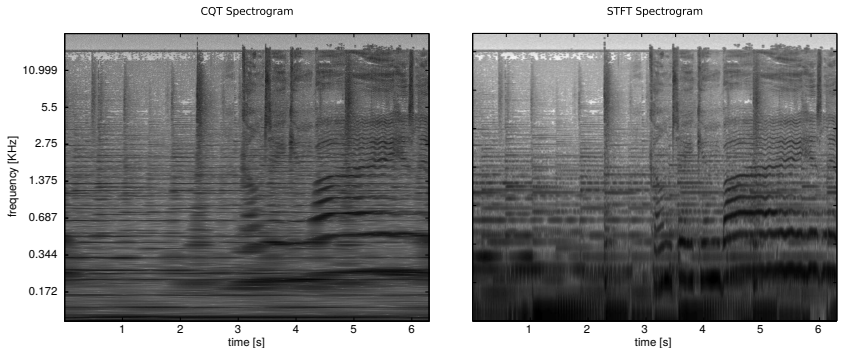


Figure 2.5: CQT and STFT spectrograms; frequency axis in logarithmic scale.

2.4.4 Sinusoidal Modeling

Spectral modeling can be seen as the task of decomposing a signal in its constituent components with some known behavior in time and frequency.

In [52] a sinusoidal model for audio and speech waveforms was presented. This model represents an audio signal $x[n]$ as a sum of time-varying sinusoids:

$$x[n] = \sum_{k=1}^{K} A_k[n] \cos(\omega_k[n].n + \theta_k[n]), \tag{2.23}$$

where $A_k[n]$, $\omega_k[n]$ and $\theta_k[n]$ represent the amplitude, frequency and phase of the k-th partial. The signal is divided and processed in frames, and the model assumes that the sinusoid's parameters remain constant along the frame.

The Sinusoidal Model can be directly related to the STFT, considering only the spectral components with a pseudo steady-state temporal behaviour. This pseudo steady-state tonal behaviour is commonly present in the sound of pitched musical instruments after their attack phase [73].

One method to obtain the model parameters is to search for the peaks of the power spectrum (computed via DFT) of each windowed frame. The frequency ω_k is related with the index k of the DFT sample which corresponds to a peak of the power spectrum: $\omega_k = \frac{2\pi k F_s}{N}$, where F_s is the sampling rate and N the window size.

The complex value of the spectral peak gives an estimate of its amplitude and phase respectively. Figure 2.6 illustrates the steps involved in the algorithm for sinusoidal modeling analysis and synthesis.

The number of peaks changes along frames due to several reasons: audio signals present pitch changes and/or rapidly varying spectrum in addition to side-lobe interactions due to windowing. Aiming at the reduction of spurious peaks a set of heuristic rules may be applied in order to group coherent peaks. In Figure 2.7 some of those rules are illustrated.

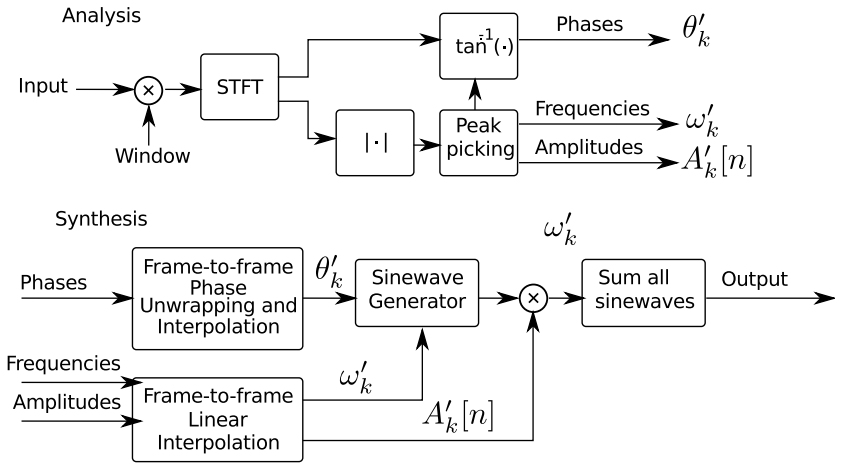


Figure 2.6: Analysis/Synthesis block diagrams of Sinusoidal Modeling (adapted from figure of original article [52]).

There are many methods to estimate the frequency and phase of the spectral peaks in the literature, among which: frequency reassignment [2], interpolation [28] and signal derivative [50]; for a detailed comparison see [30].

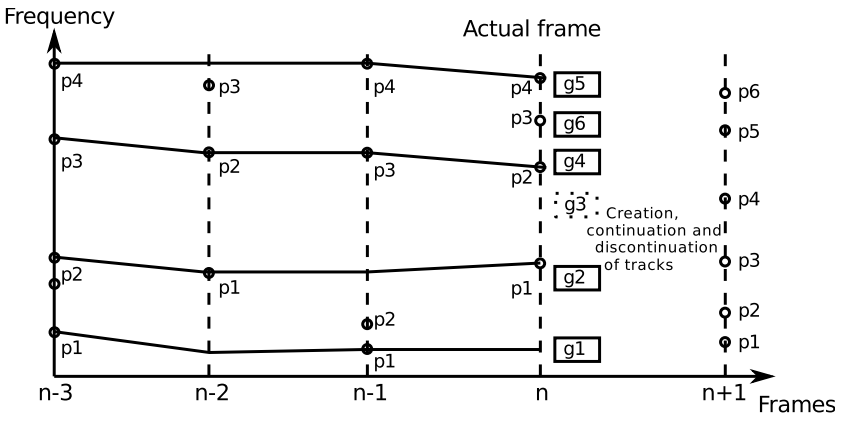


Figure 2.7: Assignment of peaks to tracks is decided by proximity (adapted from [74]). Tracks may be discontinued if not find adequate peaks for several frames; and new tracks may be created if a series of coherent peaks is find for several frames.

Sinusoids-plus-Noise Model

Musical signals are not always properly modeled by slow varying sinusoids. Modeling wideband noise as a sum of sinusoids results in hundreds of short-duration and closely spaced partials.

A sinusoids-plus-noise model is proposed in [72] and [73]. In sinusoids-plus-noise models two components are present: the deterministic part of the signal, formed by the sum of slowly varying partials; and the stochastic part of the signal, $e[n]$:

$$x[n] = \sum_{k=1}^{K} A_k[n] \cos(\omega_k[n].n + \theta_k[n]) + e[n]. \tag{2.24}$$

The stochastic part $e[n]$ can be described as filtered white noise:

$$e[n] = \sum_{k=-\infty}^{\infty} h_n[n-k]u[k], \tag{2.25}$$

where $u[k]$ is white noise and $h_n[k]$ is the impulse response of the time-varying frequency shaping filter at frame n.

For example, wind-driven instruments are properly modeled by this model: the sinusoidal part is associated with the self-sustained oscillation, while the noise-like residual is produced by the turbulent airflow.

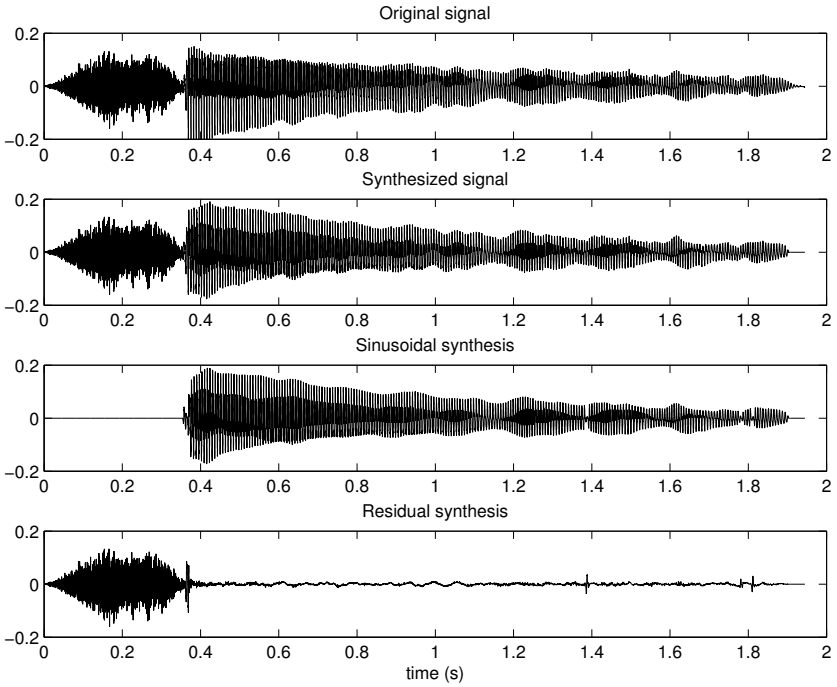


Figure 2.8: Sinusoidal modeling resynthesis.

The analysis procedure first detects partials by studying the time-varying spectral characteristics of the sound and represents them as time-varying sinusoids. These partials are then subtracted from the original sound and the remaining "residual" is represented as a time-varying filtered white noise component. Figure 2.8 depicts the original and resynthesized signals along with its separated sinusoidal and residual components, and Figure 2.9 shows the spectrogram and the sinusoidal tracks for the previously described word "so".

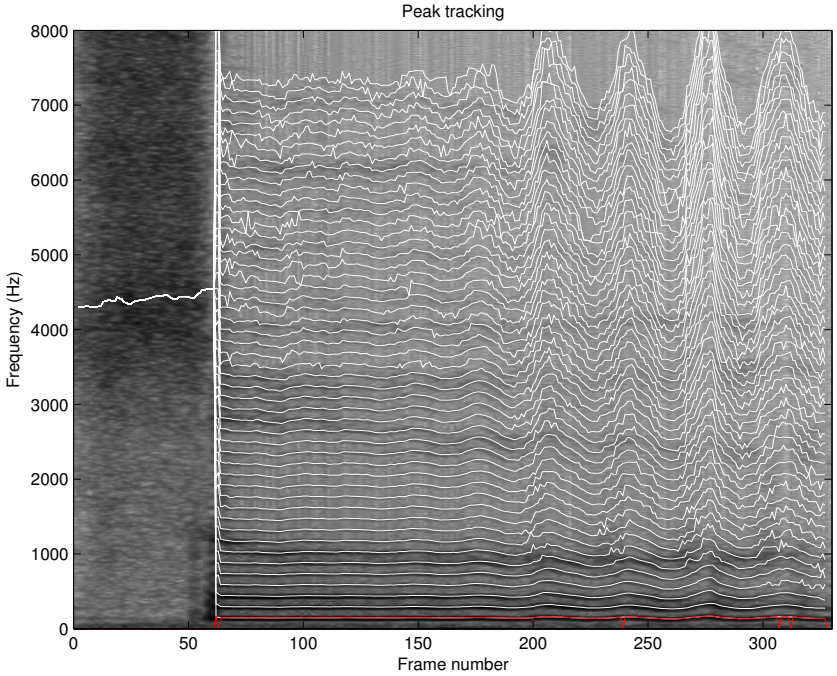


Figure 2.9: Spectrogram with sinusoidal tracks superimposed for word "so". One can observe that the voiced part is well modeled by the sinusoidal tracks.

Sinusoids plus Noise plus Transients Model

The previously presented models implicitly assume that the signal evolves slowly over time, and consequently also the model parameters. This assumption does not always holds; a clear example is the sharp attack of a strumming note. Transients modeled as short-time noise components result in distorted and poorly defined attacks. Moreover, in applications where transients have to be processed in a different manner[3] a model for transient sounds needs to be defined [51, 83, 18].

Transient Modeling Synthesis

The first proposal for a low-order parametric model for transients was presented in [83], the so called Transient-Modeling Synthesis (TMS). This model is inspired in the isomorphic duality between well-developed sines and transients. Transient components (like note attacks, or drum sounds) have an impulsive nature in the time domain; then, due to the time-frequency duality, an oscillatory behaviour can be observed in frequency. Thus, the tracks obtained by performing sinusoidal modeling in the frequency domain, represent the transients in the time domain.

Mapping from the time domain to the desired frequency domain is done using the discrete cosine transform (DCT). For $n, k \ \epsilon \ [0, 1, ..., N-1]$ one possible definition of the DCT is:

[3]Time-frequency modifications and high-quality coding are examples of this.

$$C(k) = \beta(k) \sum_{n=0}^{N-1} x(n) \cos\left[\frac{(2n+1)k\pi}{2N}\right],$$

where $\beta(k) = \sqrt{1/N}$ for $k = 1$, and $\beta(k) = \sqrt{2/k}$ otherwise.

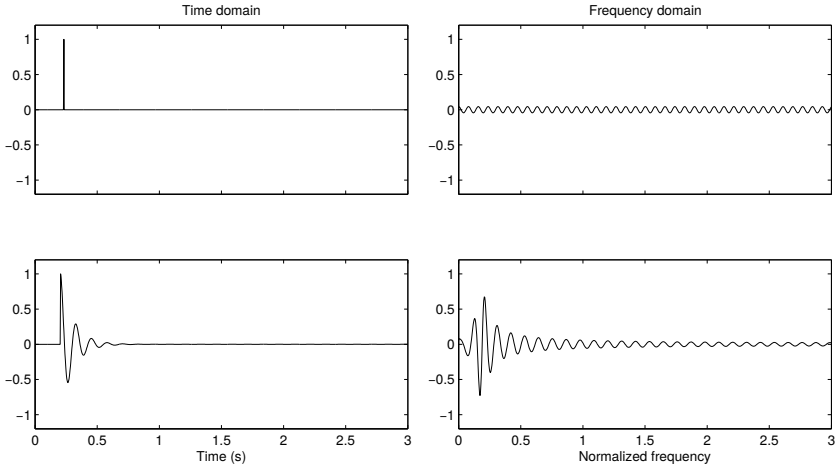


Figure 2.10: Left: Time domain impulse/pulse. Right: Spectra

In Figure 2.10 two synthetic signals and their correspondent spectral contents are displayed. In the upper plot a Kronecker delta in time (left) corresponds to a sinusoid in frequency (right). The bottom plots illustrate a more realistic example, an exponentially damped sinusoidal in time shows a clear sinusoidal nature in frequency.

The analysis algorithm begins by taking non-overlapping blocks of the input signal, (In [83], the authors recommend one second of audio duration per block). The DCT is performed on each block and then sinusoidal modeling is applied. The peak tracking algorithm in the sinusoidal modeling stage can be favoured if one analyzes the DCT frame backwards. The model output is composed by amplitude ($A^k_{l,m}$), frequency ($F^k_{l,m}$) and phase ($\phi^k_{l,m}$), where l is the frame, m the DCT block and k the sinusoid index.

Dictionary-based Matching Pursuit

Orthogonal bases (e.g. the STFT seen in section 2.3) are a class of dictionaries designed to be optimal in the sense of quantity of elements. A natural drawback of these dictionaries is the high number of active elements needed to represent a signal.

Minimizing the number of active elements as design criterion results in over-complete dictionaries. In [49] the Matching Pursuit algorithm is presented as a tool for solving the problem of decomposing a signal into the elements of over-complete dictionaries. It is an iterative algorithm that, at each step, searches in the dictionary for the element that captures most of the signal energy.

A method to separate an audio signal into tonal and transient components is proposed in [75]. The main dictionary is the result of concatenating two dictionaries of Gabor atoms. A time-concentrated atom dictionary captures the energy of the transient, while a frequency-concentrated dictionary captures the tonal information as illustrated in figure 2.11. Figure 2.12 shows the decomposition of a glockenspiel sound using these two dictionaries.

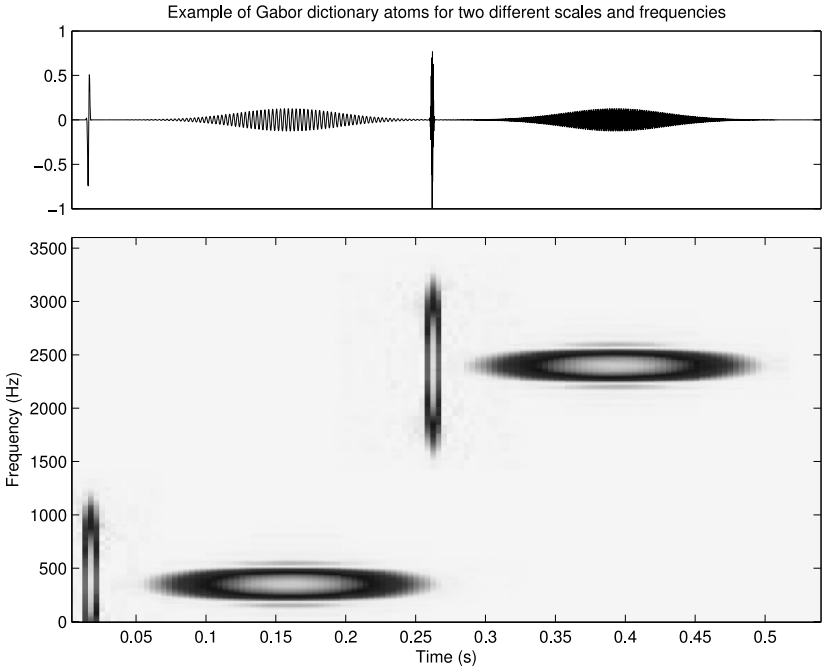


Figure 2.11: Atoms from two Gabor Dictionaries with different time-frequency behavior.

2.4.5 Non-Negative Matrix Factorization NMF

Non-negative Matrix Factorization (NMF) is a linear algebra and signal-processing technique with multiple applications [45]. With the NMF it is possible to obtain a linear representation of reduced dimension with a part-based decomposition. Part-based decompositions plays a fundamental role when the interpretation of physical processes depends on their positiveness.

Given a non-negative matrix V with dimensions $F \times N$, the NMF looks for the best approximate factorization:

$$V \approx WH, \tag{2.26}$$

where W and H are non-negative matrices with dimensions $F \times K$ and $K \times N$ respectively. Usually K is chosen such that $FK + KN << FN$ in order to achieve an effective reduction of the problem's dimensionality.

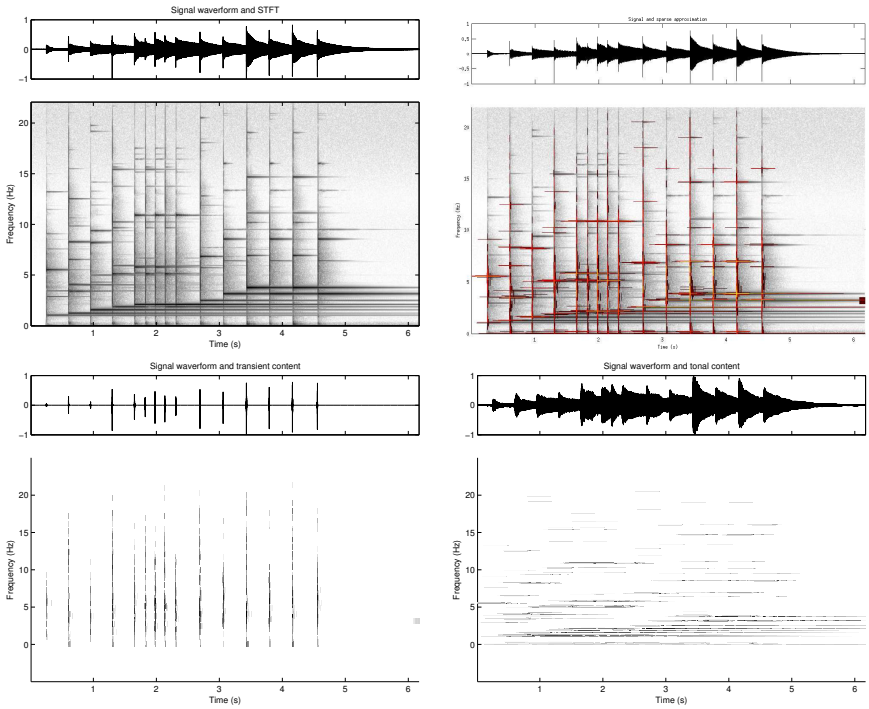


Figure 2.12: Matching Pursuit with different Gabor dictionaries: decomposition of a glockenspiel sound.

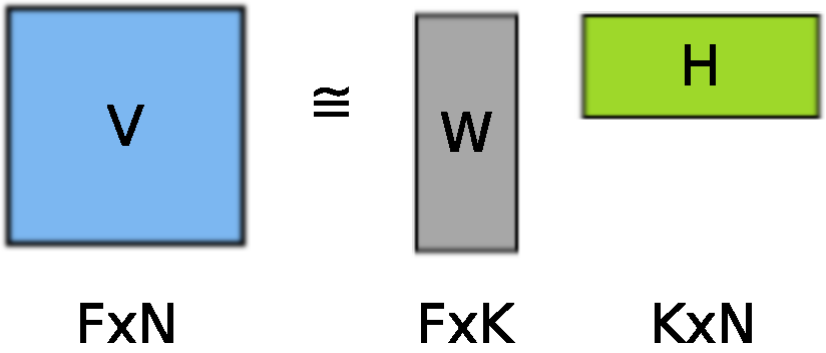


Figure 2.13: Matrix decomposition diagram.

The role of W and H is application dependant. In the context of source separation, matrix W represents a dictionary of atoms, while H represents the mixing matrix.

The NMF factorization problem is equivalent to the minimization with restrictions defined as:

$$\min_{W,H \geq 0} D(V|WH), \tag{2.27}$$

where the cost function $D(V|WH)$ is defined as:

$$D(V|WH) = \sum_{f=1}^{F}\sum_{n=1}^{N} d([V]_{fn}|[WH]_{fn}) \tag{2.28}$$

and function $d(x|y)$ is a scalar divergence.

Multiple divergences with different properties can be utilized with the NMF, and the best choice depends on the application. The Euclidean (Eq. 2.29) and the Kullback-Leibler (Eq. 2.30) are two common divergences.

$$d_{EUC}(x|y) = \frac{1}{2}(x-y)^2 \tag{2.29}$$

$$d_{KL}(x|y) = x\log(\frac{x}{y}) - x + y \tag{2.30}$$

NMF in Audio Applications

In the context of audio applications, the NMF algorithm is commonly applied to the magnitude spectrogram[4] [86].

The spectrogram is then decomposed in positive parts. The column w_k of the dictionary matrix W represent the spectrum of the k-th base element, while the correspondent row h_k of the activation matrix H represents the gain coefficient along time frames. An important property is that the base elements w_k belong to the same space as the signal spectrum.

The product of the k-th column of W by the k-th row of H gives us an approximation of the spectrogram of the k-th source X_k.

$$X_k = w_k.h_k \tag{2.31}$$

To account for musical structure (e.g. time continuity, frequency sparseness), regularization can be added to the model [22], [85].

[4] Any non-negative T-F representation can be utilized with the NMF; the spectrogram is the most common.

Chapter 3

Transient and Steady-State separation

3.1 Model of transient and steady-state components

To precisely and meaningfully discriminate transient and steady-state components is not an easy task. Along this work, transients components are considered as broad-band with highly concentrated energy in time, whereas steady-state components as discrete narrow-band with smooth temporal behaviour.

Several works have addressed this problem. In [81] a feature-based classification of components extracted via Independent Component Analysis is presented. In [32], the authors propose a two-stage processing, involving a non-negative matrix factorization to decompose the spectrogram into components having fixed spectrum with time-varying gain, and a support vector machine to classify them as either pitched or drum components. Recently, various separation methods based exclusively in the anisotropy property were proposed [60, 23].

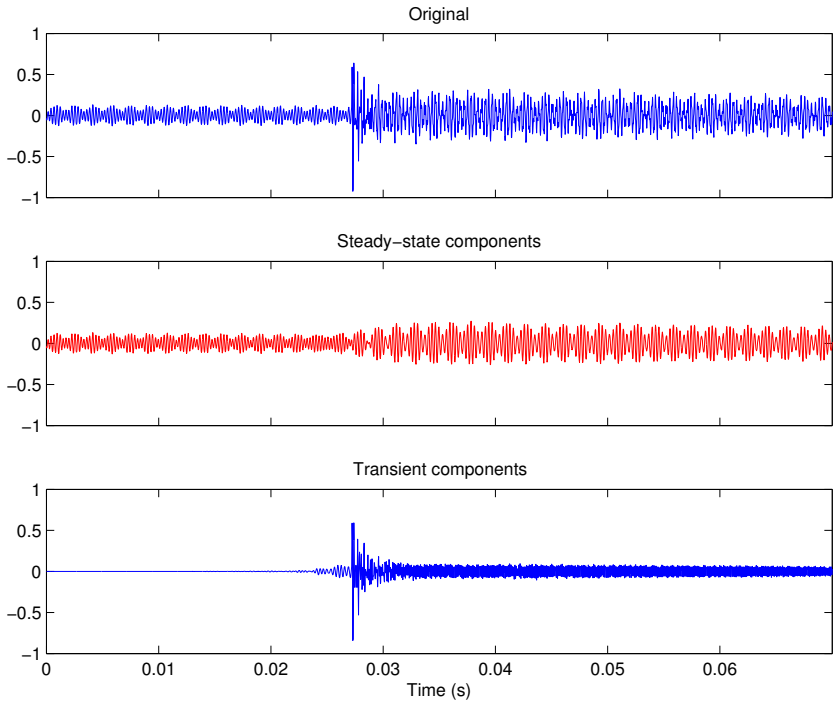


Figure 3.1: Transient and steady-state components of a glockenspiel sound.

3.2 Separation Methods

For an audio signal $s(t)$ with power spectrogram $|S(n,k)|^2$, the transient and steady-state separation problem consists in finding the transient and the steady-state spectrograms $S_t(n,k)$ and $S_{ss}(n,k)$ respectively that satisfy the following properties:

- $|S_{ss}(n,k)|$ sparse in frequency and smooth in time,

- $|S_t(n,k)|$ sparse in time and smooth in frequency,
- $|S_t(n,k)| + |S_{ss}(n,k)| = |S(n,k)|$,

This problem can be formulated as the minimization of a cost function J with constraints (as suggested in [65]) as follows:

$$\begin{aligned} J(|S_t|,|S_{ss}|) = &\sum_{n,k} D\left[S(n,k), S_{ss}(n,k) + S_t(n,k)\right] \\ &+ \frac{1}{2\sigma_{ss}^2} \sum_{n,k} (|S_{ss}(n-1,k)| - |S_{ss}(n,k)|) \\ &+ \frac{1}{2\sigma_t^2} \sum_{n,k} (|S_t(n,k-1)| - |S_t(n,k)|), \end{aligned} \tag{3.1}$$

with the restrictions: $|S_t| \geq 0$ and $|S_{ss}| \geq 0$ and being D a divergence. The first term measures the distance between $|S(t)|$ and $|S_t| + |S_{ss}|$, the second penalizes the temporal discontinuity and the third measures the frequency smoothness. The values σ_t and σ_{ss} , determines the relative weights of the transient and steady-state components in the cost function, respectively.

These components can be seen in the spectrogram as vertical and horizontal ridges, respectively. Figure 3.2 shows a typical spectrogram, computed from a scale played with a glockenspiel. The notes attacks exhibit a transient behaviour; by contrast, the sustained part is clearly steady-state.

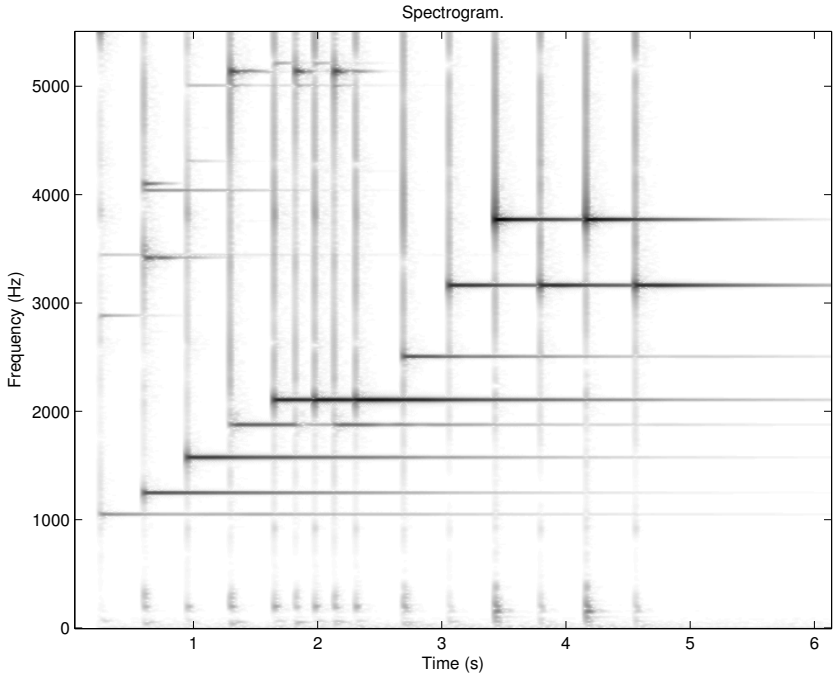


Figure 3.2: Spectrogram of a glockenspiel (N=4096,Hop=256).

The minimization of the previous cost function can be thought as a non-linear filter which smooths the original spectrogram. It filters out the horizontal ridges, corresponding to transients, to obtain the magnitude of the steady-state spectrogram $|S_{ss}|$, and removes the vertical ridges, which corresponds to partials of steady-state components, to obtain the magnitude of the transient spectrogram $|S_t|$.

3.2.1 Median Filter

As seen in figure 3.2, along the time axis the transient components are atypical events, and thus can be considered as outliers, just as steady-state components can be considered outliers along the frequency axis. A common procedure to eliminate outliers is to use of a median filter.

Median filters are commonly used in signal processing for denoising, e.g. removing salt and pepper noise in image filtering [63] or removing noise from digitized vinyl records in audio [39], [37]. The median filter consist in sliding a window of size $2N + 1$ along the signal, replacing the centre value of each window by the median of the samples within the window itself.

In [23] the utilization of median filters in the transient and steady-state component separation problem is proposed.

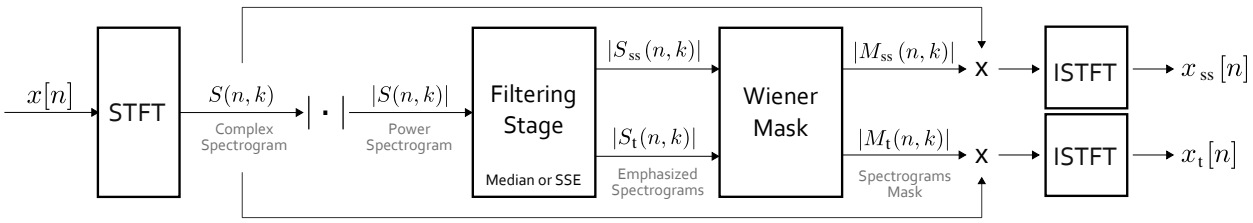


Figure 3.3: Diagram of the entire process.

The general procedure is illustrated in figure 3.3, and can be decomposed in four stages as follows:

1. Obtain a time-frequency representation for the digital audio signal, typically a spectrogram computed via the Short-Term Fourier Transform (STFT):

$$S(n,k) = \sum_i x(i).w(i - nT)e^{-\frac{j2\pi ik}{N}}. \tag{3.2}$$

2. Apply a median filter to the power spectrogram S along the frequency axis to eliminate steady-state components and obtain a "transient emphasized" spectrogram S_{t}, as well as along the time axis to eliminate transient peaks and obtain a "steady-state emphasized" spectrogram S_{ss}:

$$S_{\mathrm{t}}(n,k) = \mathrm{median}(|S(n-l:n+l,k)|), \tag{3.3}$$

$$S_{\mathrm{ss}}(n,k) = \mathrm{median}(|S(n,k-l:k+l)|). \tag{3.4}$$

3. From the emphasized spectrograms, calculate two soft masks based on the Wiener filter, given by:

$$M_{\mathrm{t}} = \frac{S_{\mathrm{t}}^2}{S_{\mathrm{ss}}^2 + S_{\mathrm{t}}^2}, \quad M_{\mathrm{ss}} = \frac{S_{\mathrm{ss}}^2}{S_{\mathrm{ss}}^2 + S_{\mathrm{t}}^2}. \tag{3.5}$$

4. Multiply each mask by the original complex spectrogram, and compute the Inverse Short-Time Fourier Transform [1] (ISTFT) of the results to obtain, respectively, the transient signal x_{t} and the steady-state signal x_{ss}.

Procedure 1 Median Filter Separation

Input: $s(n), N_{ss}, N_t$ [2]
Outputs: $s_{ss}(n)$ and $s_t(n)$
$S(n,k) = STFT[s(n)]$
for $i = 1$ to n_{max} **do**
  for $j = 1$ to k_{max} **do**
    $\hat{S}_{ss}(i,j) = \text{median}[|S(i - N_{ss} : i + N_{ss}, j)|]$
    $\hat{S}_t(i,j) = \text{median}[|S(i, j - N_t : j + N_t)|]$
  end for
end for
$M_{\mathrm{t}} = \frac{S_{\mathrm{t}}^2}{S_{\mathrm{ss}}^2 + S_{\mathrm{t}}^2}$
$M_{\mathrm{ss}} = \frac{S_{\mathrm{ss}}^2}{S_{\mathrm{ss}}^2 + S_{\mathrm{t}}^2}$
$S_t(n,k) = M_t. * S(n,k)$
$S_{ss}(n,k) = M_{ss}. * S(n,k)$
$s_t(t) = ISTFT(S_t(n,k))$
$s_{ss}(t) = ISTFT(S_{ss}(n,k))$

The method allows for perfect reconstruction ($x = x_{\mathrm{ss}} + x_{\mathrm{t}}$). Furthermore, the processes involved are very simple, thus allowing an efficient implementation. Figure 3.4 presents a spectrogram and the signals involved in the filtering stage, the time evolution of a fixed bin and the spectra of a frame. The complete pseudo-code for the algorithm is presented in Procedure 1.

3.2.2 SSE Filter

In [43] a nonlinear filter for stochastic spectrum estimation was presented, and yielded very good results. The main idea is to remove the predominant partials to obtain the stochastic spectrum estimation. We propose using this filter as an alternative to the median filter utilized in the second stage of the procedure described in Section 3.2.1.

[1] The ISTFT is calculated via an Overlap-and-Add procedure.
[2] N_{ss} and N_t are the steady-state and transient median filter lengths respectively.

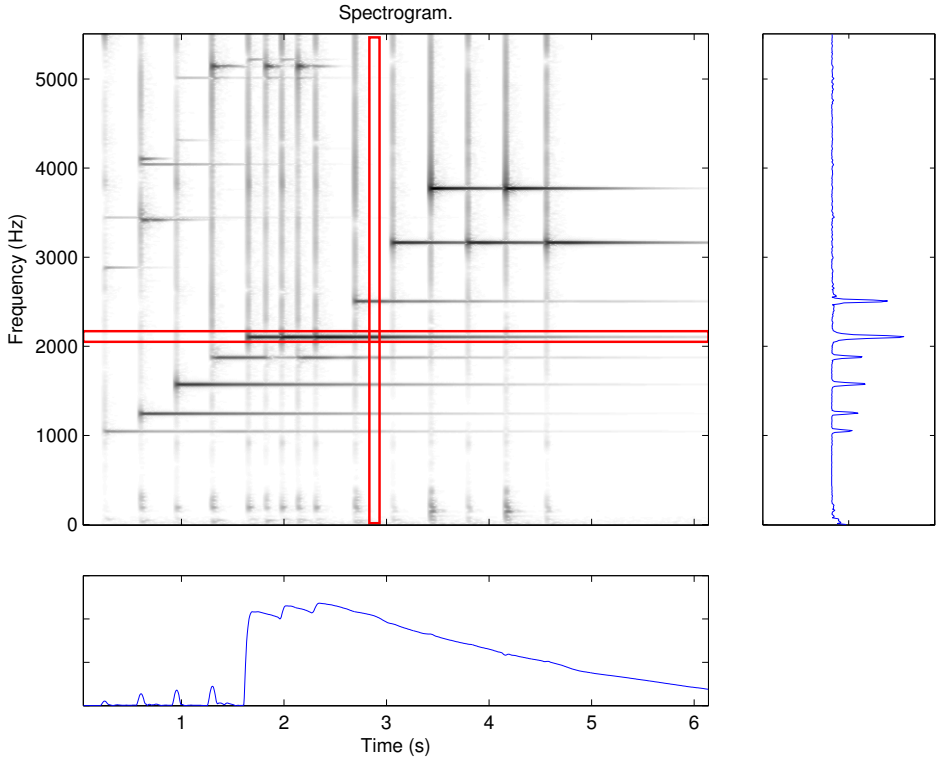


Figure 3.4: Glockenspiel spectrogram (center), time evolution for a fixed FFT bin (bottom) and spectral content of a frame (right).

The SSE algorithm can be resumed as follows:

Firstly, the reciprocal R of each element of the power spectrogram is calculated, turning the peaks of $S(n,k)$ into valleys of $R(n,k)$:

$$R(n,k) = S^{-1}(n,k). \tag{3.6}$$

Then, a moving average (MA) filter is applied along the time axis to filter the transient components, and along frequency bins to eliminate the steady-state components. The MA applied to a valley in R (originally a peak in S) tends to make it disappear. The estimated reciprocals of the desired "transient emphasized" and "steady-state emphasized" spectra are given respectively by

$$\hat{R}_{\mathrm{t}}(n,k) = \frac{1}{M_t+1} \sum_{i=-M_t/2}^{M_t/2} R(n,k+i), \tag{3.7}$$

$$\hat{R}_{\mathrm{ss}}(n,k) = \frac{1}{M_{ss}+1} \sum_{i=-M_{ss}/2}^{Mss/2} R(n+i,k). \tag{3.8}$$

The respective stochastic spectrum estimates (SSE) are then computed as

$$S_{\mathrm{t}}(n,k) = \hat{R}_{\mathrm{t}}^{-1}(n,k), \quad S_{\mathrm{ss}}(n,k) = \hat{R}_{\mathrm{ss}}^{-1}(n,k). \tag{3.9}$$

Procedure 2 SSE Filter

Input: $S(n,k), M_{ss}, M_t$[3]
Outputs: $S_{ss}(n,k)$ and $S_t(n,k)$

1: $R(n,k) = S^{-1}(n,k)$
2: **for** $i = 1$ to n_{max} **do**
3: **for** $j = 1$ to k_{max} **do**
4: $\hat{R}_{ss}(i,j) = \frac{1}{M_{ss}+1} \sum_{m=-M_{ss}/2}^{M_{ss}/2} R(n+i,k)$
5: $\hat{R}_t(i,j) = \frac{1}{M_t+1} \sum_{m=-M_t/2}^{M_t/2} R(n,k+i)$
6: **end for**
7: **end for**
8: $S_t(n,k) = \hat{R}_t^{-1}(n,k)$
9: $S_s s(n,k) = \hat{R}_{ss}^{-1}(n,k)$

The complete pseudo-code for the algorithm is presented in Procedure 2.

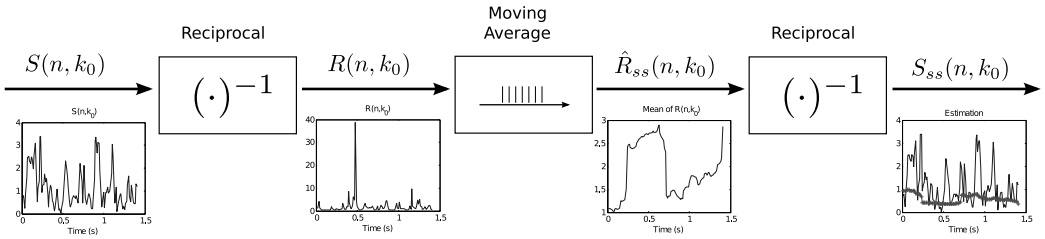


Figure 3.5: Steps of the Stochastic Spectral Estimation applied along the time axis. Based on [43].

Figure 3.5 illustrates the process along time for a fixed frequency bin and in Figure 3.6, a typical spectrogram, computed for an excerpt of a western popular song with piano, drums and vocal, is shown. In the first three seconds, when only the instruments are present, one observes their respective steady-state and transient behaviors. Afterwards, the voice, presenting a deep vibrato, enters.

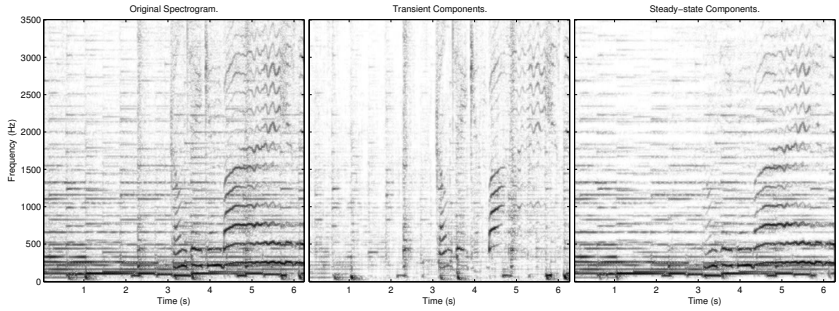


Figure 3.6: Left: Spectrogram of a excerpt from a popular music song. Middle: Spectrogram with transient components. Right: Spectrogram with steady-state components.

[3] M_{ss} and M_t are the steady-state and transient SSE filter lengths respectively.

One can observe in the first three seconds (when only drums and piano are present) that the separation is as expected, meeting the transient and steady-state models as defined, respectively. The deep vibrato voice that enters before second three is outside the defined model, and thus it not completely modeled by transient or steady-state components.

3.3 Extensions

Some extensions can be proposed to overcome limitations of the previously described methods. The following sections present three extensions of the algorithm which allows the model to best fit some particular signals or applications.

3.3.1 Iterative Filtering

The previously described procedure separates the input signal $s(n)$ into two signals $s_t(n)$ and $s_{ss}(n)$ containing the transient and steady-state components respectively. A third kind of component, the residual component s_{res}, can be defined as the non steady-state and non transient each time the output signals $s_{ss}(n)$ and $s_t(n)$ are re-filtered. Grouping the components that do not fit in the transient and steady-state models as the residual, allows the isolation of the steady-state and transient components that fit the models in their respective signals.

Figure 3.7 illustrates the process and figure 3.8 shows the results for different numbers of iterations.

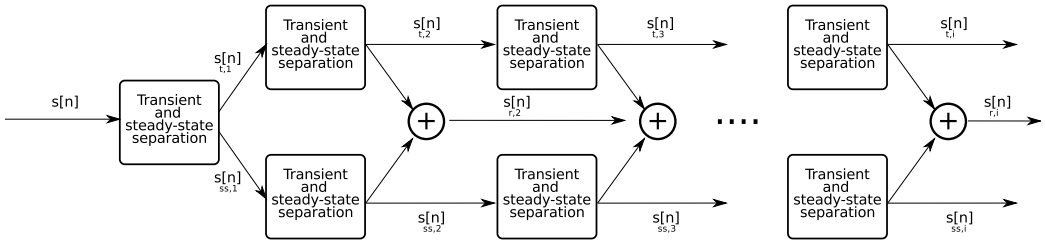


Figure 3.7: Diagram of the iterative process.

As the number of iterations increases, the residual component extracted at each step tends to decrease. One possible way to determine the number of iterations is by using psychoacoustic models to find when the residual increment is perceptually irrelevant [33, 34].

When the model fits very well to the signal to be analysed, the utilization of this iterative processing gives good results in the separation of transient and steady-states components. As in the original method, the addition of the three separated components, s_t, s_{ss} and s_{res} results in the original signal s.

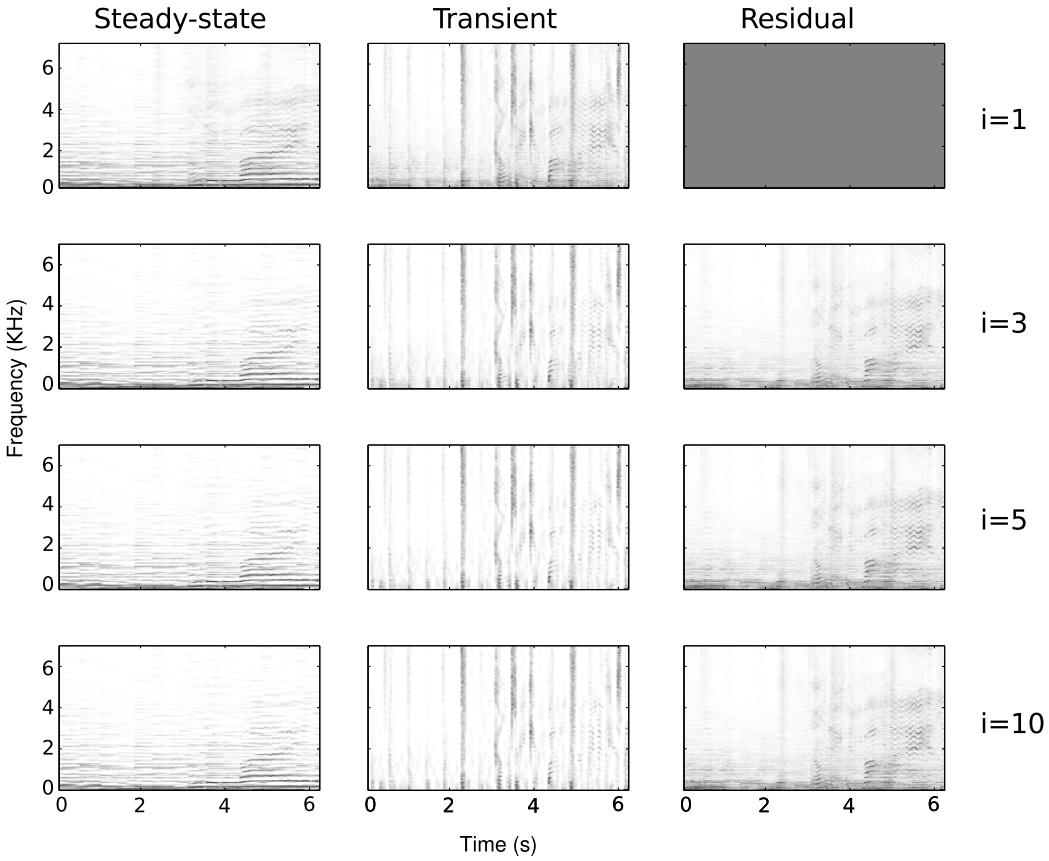


Figure 3.8: Steady-state, transient component and residual extraction for different number of iterations.

3.3.2 Relaxed components

Modeling steady-state components as horizontal ridges in the spectrogram can be too restrictive in some cases. As an example, free intonation instruments, such as the human voice, which can change its pitch in a slow and continuous way over time (glissando, vibrato, etc.) should be included in the steady-state part of the signal. The voice with vibrato effect in figure 3.6 is an example.

An approach to relax the definition of steady-state component to permit slow variations in the evolution of partials, is to define a time-frequency window as depicted in figure 3.9 for each element of the spectrogram. First the maximum along frequency is applied, and then the filtering stage (median or SSE) is applied.

In Figure 3.10, the spectrogram of a synthetic signal is shown together with the respective decompositions. The signal is the superposition of a periodic signal with fundamental frequency of 400 Hz and exhibiting sinusoidal vibrato to two clicks at instants 0.3 s and 0.6 s. The figure shows that the original procedure is unable to follow the vibrato, thus energy from the higher partials is present in the transient component, while in the proposed extension the presence of vibrato in the transient-component signal is just noticeable. In [47] a similar approach is

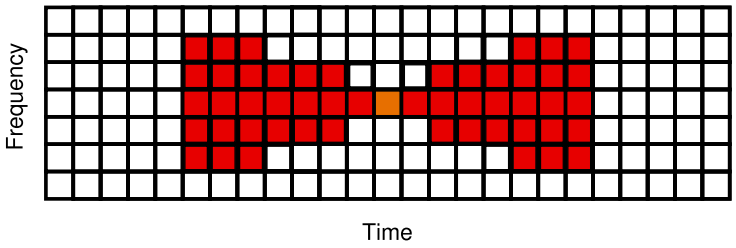


Figure 3.9: Time-frequency kernel.

applied to the source separation task.

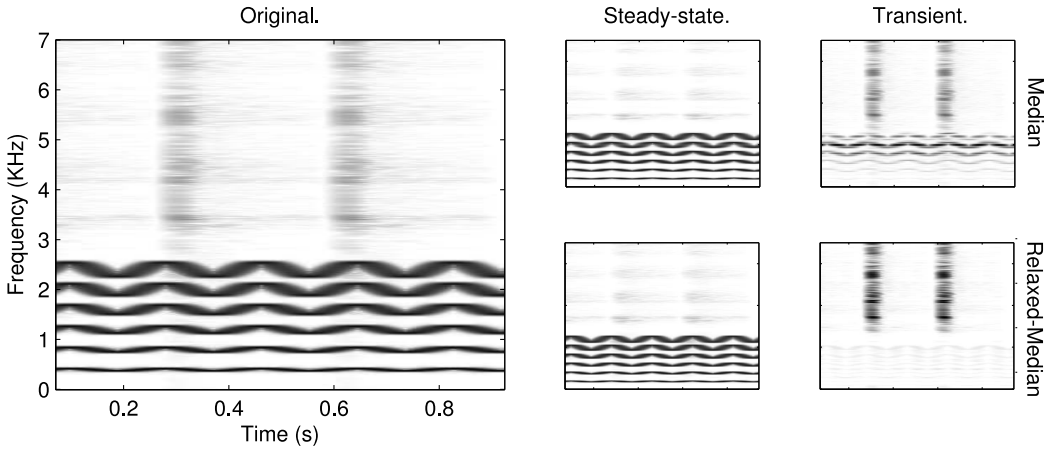


Figure 3.10: left) Spectrogram of a superposition of a periodic signal with vibrato to two clicks; right) Comparison between the original and the modified decompositions.

3.3.3 Sub-Band processing

The sub-band processing strategy is commonly utilized in audio processing and coding. The input signal $x[n]$ is splitted by a bank of filters, each sub-band is properly processed, and then the signal is filtered and recombined to obtain the output signal y[n]. Figure 3.11 illustrates the process.

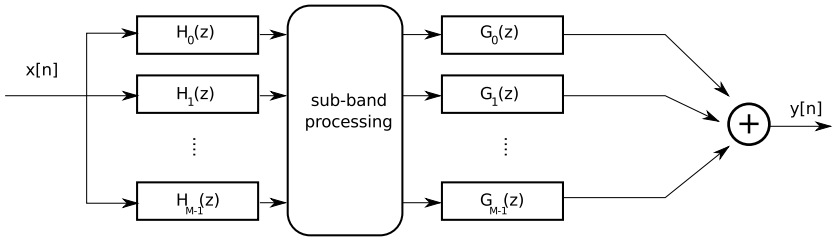


Figure 3.11: Diagram of the sub-band processing schema.

The algorithms presented in the previous section can be favored by sub-band processing. The TF-representation can be selectively adjusted depending on the

frequencies involved; typically window lengths utilized at low frequencies tend to be longer than at high frequencies to allow better discrimination. Another advantage is the flexibility in the choice of the process parameters: the length of the median filter or the moving average in the SSE can be independently adjusted for each band. The obvious disadvantage is the increased complexity of the process.

In figure 3.12 the frequency response of a 4-band filter-bank is shown: the stop-band attenuation of each filter is 60 dB, and the passband ripple is less than 5%.

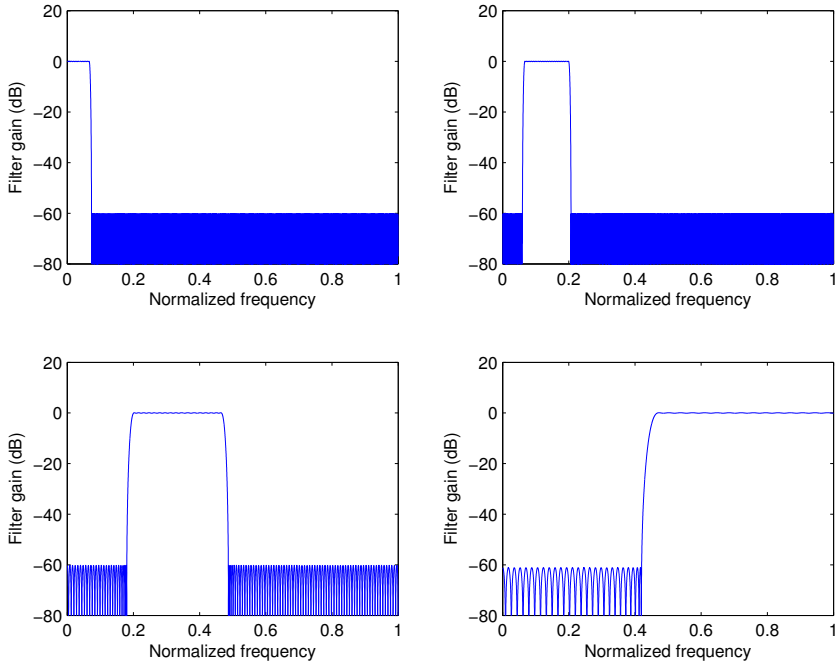


Figure 3.12: Filters response for the Filter-bank analysis.

In figure 3.13 the result employing different window and filter lengths for each band is shown. The window lengths are [4096 2048 1024 1024] samples and the lengths of the moving average filters of the SSE algorithm are $M_t = [15\ 13\ 11\ 11]$ along time, and $M_{ss} = [11\ 13\ 15\ 15]$ along frequency.

The separation using sub-band processing presents some improvements in the high-frequency range compared to the original method. This can be explained by the more detailed time resolution in the time-frequency representation of the upper-band due to the utilization of a smaller window. The overhead produced by the sub-band processing and the limited benefits in the separation make it justifiable under limited circumstances.

3.4 Reconstruction method

Wiener filtering is one of the most widely used methods for source separation. Its objective is to minimize the mean square error, and succeeds when wide-sense

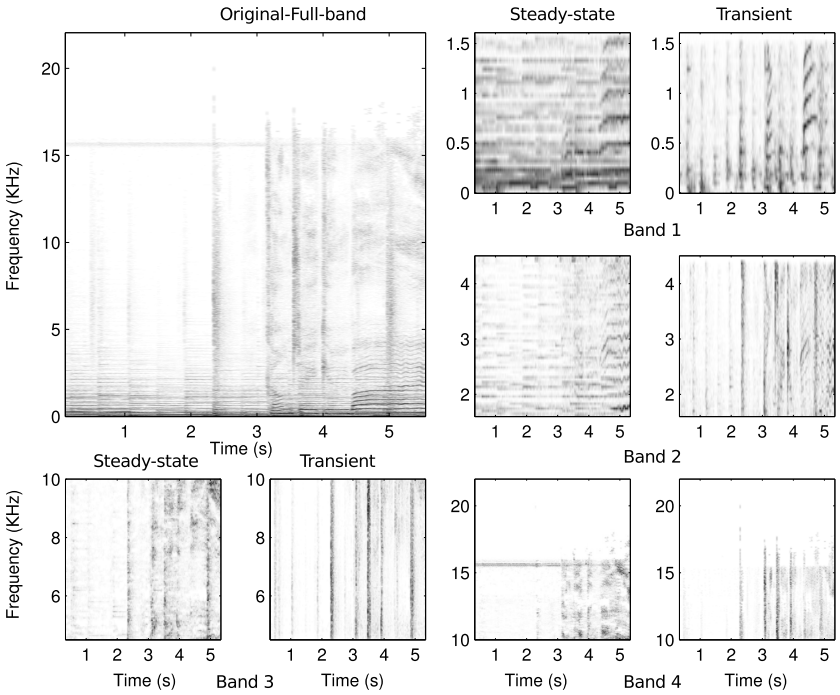


Figure 3.13: Transient and steady-state separation spectrograms for each sub-band. Band 1 is 0 to 1.5 kHz, band 2 is 1.5 to 4.4 kHz, band 3 is 4.4 to 10.3 kHz and band 4 is 10.3 to 22 kHz.

stationary processes are involved [31]. To design a Wiener filter is necessary to state the Wiener-Hopf equations:

$$\sum_{l=-\infty}^{\infty} h[l]r_u[n-l] = r_{du}[n] = h[n] * r_u[n], \forall n, \tag{3.10}$$

where $r_u[n]$ is the autocorrelation of the input process $u[n]$ and $r_{du}[n]$ the cross-correlation between the input and the desired signal $d[n]$. In the frequency domain Eq. (3.10) becomes:

$$H(\omega) = \frac{P_{du}(\omega)}{P_u(\omega)}. \tag{3.11}$$

In our problem, the input is the original signal $s[n, t_0]$ multiplied by a window centered at time t_0 [4]. The desired output signals are the steady-state and transient component signals $s_{ss}[n, t_0]$ and $s_t[n, t_0]$, respectively, as depicted in figure 3.14. Also, the input signal $s[n, t_0]$ is the sum of the steady-state $s_{ss}[n, t_0]$ and transient $s_t[n, t_0]$ signals,

$$s[n, t_0] = s_{ss}[n, t_0] + s_t[n, t_0], \tag{3.12}$$

[4]Audio signals are in general non-stationary processes that locally observed can be approximated as wide-sense stationary.

where s_{ss} and s_t are zero-mean, independent processes renamed as desired signals d_2 and d_1 respectively.

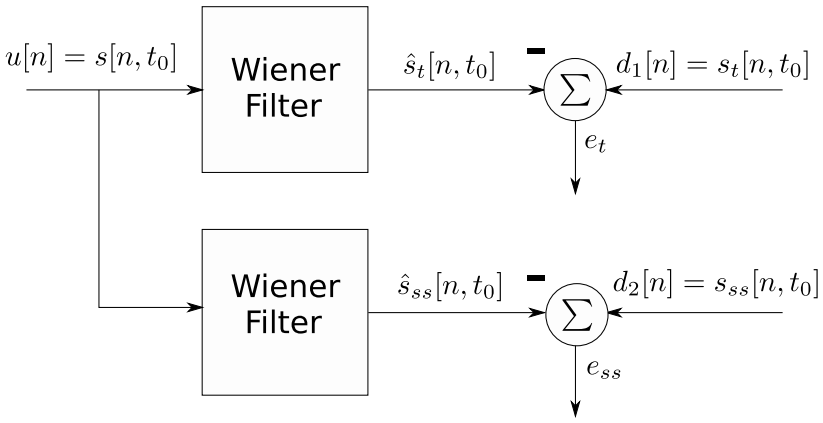


Figure 3.14: Wiener filter configuration for transient and steady-state component separation.

The autocorrelation of the input process can be expressed as:

$$r_u[k] = E[s[n, t_o]s[n-k, t_o]] = r_{d_2u}[k] + r_{d_1u}[k]. \tag{3.13}$$

As s_{ss} and s_t are uncorrelated, the cross-correlation between the input and the desired signal $r_{d_iu}[n]$ can be expressed as:

$$\begin{aligned} r_{d_1u}[k] &= E[s[n, t_o]s_t[n-k, t_o]] = r_{d_1d_1}[k], \\ r_{d_2u}[k] &= E[s[n, t_o]s_{ss}[n-k, t_o]] = r_{d_2d_2}[k]; \end{aligned} \tag{3.14}$$

in the frequency domain Eqs. (3.13), (3.14) become:

$$P_u(\omega, t_0) = |S[\omega, t_o]|^2 = P_{d_1u}(\omega, t_0) + P_{d_2u}(\omega, t_0), \tag{3.15}$$

$$\begin{aligned} P_{d_1u}(\omega, t_0) &= |S_t(\omega, t_o)|^2, \\ P_{d_2u}(\omega, t_0) &= |S_{ss}(\omega, t_o)|^2. \end{aligned} \tag{3.16}$$

Combining Eqs. (3.11), (3.16) and (3.15) one finds that the frequency response of the Wiener filter is:

$$\begin{aligned} H_t(\omega) &= \frac{P_{d_1u}}{P_{d_1u}+P_{d_2u}} = \frac{|S_t(\omega,t_o)|^2}{|S_t(\omega,t_o)|^2+|S_{ss}(\omega,t_o)|^2}, \\ H_{ss}(\omega) &= \frac{P_{d_2u}}{P_{d_1u}+P_{d_2u}} = \frac{|S_{ss}(\omega,t_o)|^2}{|S_t(\omega,t_o)|^2+|S_{ss}(\omega,t_o)|^2}; \end{aligned} \tag{3.17}$$

hence, the steady-state and transient component spectrograms can be calculated by the weighted average:

$$\begin{aligned} &\frac{|S_t(\omega,t_o]|^2}{|S_t(j\omega,t_o)|^2+|S_{ss}(\omega,t_o)|^2} S(\omega, t_0), \\ &\frac{|S_{ss}(\omega,t_o]|^2}{|S_t(j\omega,t_o)|^2+|S_{ss}(\omega,t_o)|^2} S(\omega, t_0). \end{aligned} \tag{3.18}$$

Finally, the transient and steady-state time signals are reconstructed applying the overlap-add procedure to the correspondent spectrogram.

Chapter 4

Tests & Applications

This chapter is divided in two sections. In the first section, a comparison of the SSE and the Median filters as the non-linear filtering stage for the transient / steady-state separation algorithm is performed. The second section describes different applications and examples of audio editing that benefit from the decomposition in transient and steady-state components.

4.1 SSE and Median filter comparison

The performance of the transient and steady-state separation algorithm is evaluated comparing the influence of the filtering stage when using the median and the SSE filter. Two different types of experiment are considered:

1. Systematic listening tests are conducted to compare the original and proposed methods as to their separation performances;

2. An application-based evaluation is carried out considering the beat-tracking and the pitch-tracking problems.

The listening tests and the application to the beat-tracking problem where presented at the "Congreso Internacional de Ciencia y Tecnología Musical CICTeM - 2013" [36]. For each type of experiment a different audio data set is used, both described in the following section.

4.1.1 Data set description

Three data sets were used in these tests, one for the subjective listening test and two others for the application-based evaluation. The audio files are mono and have a sampling rate of 44.1 kHz and 16-bit resolution.

The data set for the subjective listening tests consists of ten-seconds length excerpts of thirteen pieces of North American popular music (rock, folk and blues) extracted from commercial records. It exhibits multiple combinations of transient and steady-state components in the sense of perceptual presence in the mix.

For the application-based evaluation, datasets from the Music Information Retrieval Evaluation eXchange (MIREX) Contest [16, 15] were utilized. The MIREX is a community-based formal evaluation framework within the Music Information Retrieval domain. To measure the performance of a beat-tracking algorithm the Audio Beat Tracking (MIREX 2006) data set was utilized. This data set is composed of twenty excerpts of western popular music of thirty-seconds duration. The beats of each recording have been annotated by 40 different listeners. For the performance of a pitch-tracking algorithm the Melody Extraction Contest (MIREX 2004 and 2005) data sets [16] were utilized. The 2004 data set is composed of twenty excerpts of western popular music of thirty-seconds duration and the 2005 dataset is composed of thirteen excerpts with different durations (from 10 to 15 seconds). Each recording of the pitch-tracking datasets has a reference corresponding melody frequency contour that was manually annotated.

4.1.2 Tests

Subjective test

In order to measure the perceptual difference between utilizing the Median filter and utilizing the SSE filter, a set of formal subjective tests was designed and conducted following the recommendations suggested in [94]. Each participant should listen and compare the separated steady-state and transient components produced by both non-linear filters.

For this purpose, a graphical user interface specifically designed to comply with the requirements of this test was adapted from [71]; a screenshot of the interface is included in the annex BFor each song in the data set, the interface presents the original signal as a reference together with the processed signals to be compared. The order of the compared signals is randomized to assure the blindness of the test. The interface implements audio controls to play/stop any of the signals, and the listener can also define loop points to allow an in-detail listening of certain parts of the audio.

Ten participants answered the next three questions for each transient and steady-state output signal:
* Q1: How much of the desired components has been properly separated?
* Q2: How much of the undesired (residual) components has been left?
* Q3: How would you rate the integrity (in the sense of naturalness) of the separated signal?

The questions were answered by controlling a slide bar that maps the responses to values from 0 to 100. The complete individual results may be found in annex A. The answers presents a wide variation between participants making it difficult to be consistently compared. Then, the raw answers were thresholded to obtain a binary value. This value indicates which algorithm performs better or otherwise if their results can be considered perceptually equivalent.

Table 4.1 summarizes the results of the transient/steady-state separation subjective test. The results show that both methods were considered virtually equivalent regarding perceptual quality.

	$Q1_{\mathrm{t}}$	$Q2_{\mathrm{t}}$	$Q3_{\mathrm{t}}$	$Q1_{\mathrm{ss}}$	$Q2_{\mathrm{ss}}$	$Q3_{\mathrm{ss}}$
Median	23.1	30.8	27.9	24.0	21,2	25.0
SSE	26.0	25	23.1	30.8	25	20.2
Equals	50,9	44.2	49	45,2	54.8	54.8

Table 4.1: Result of subjective test. The subscripts (t) and (ss) indicates transient and steady-state respectively.

4.1.3 Beat Tracking

Most of the beat information present in music is contained in its transient components. Thus, beat-tracking algorithms could potentially be favored by a preprocess-

ing stage after which only transient components are left. Such hypothesis is tested by evaluating the performance of the state-of-the-art beat-tracking algorithm (presented in [20]) on the 2006 MIREX audio beat-tracking test database [16].

Dynamic Programming based beat-tracking algorithm

The output of a beat-tracker is the sequence of time instants derived from the music audio signal that correspond to the instants in which a human listener would tap his foot. For the authors of the algorithm used in this work [20] the beat times need to satisfy two constraints: follow a regular rhythmic pattern, reflecting a locally constant inter-beat-interval; and correspond to a note onset played by one of the instruments.

These constraints are expressed as two functions: the transition cost function and a local onset strength.

The beats are then the set of time instants that minimize those cost functions. The best-scoring set of beats is found by Dynamic Programming, which decomposes the entire problem (find the global optimum within a exponential-sized set) into simpler optimization problems at each step, finding the globally optimal beat sequence.

Performance evaluation

In order to perform an objective comparison of the beat-tracking algorithm with and without the preprocessing stage, the performance was evaluated with the Beat Tracking Evaluation toolbox [14]. A brief description of each of the evaluation methods implemented in the toolbox follows; for a complete survey, see [13, 12].

The **F-measure** is a generic performance measure in the information retrieval context. It is determined by the relation between the correctly estimated beats, the false positives and the false negatives. An estimated beat is considered correct if it falls into a 70-ms wide tolerance window centered at each annotated beat. **Cemgil** et al [8] propose the utilization of a Gaussian error function to take into account the beats estimation accuracy . An error function is build centering a Gaussian at each annotated beat. Then, the performance indicator is calculated as the sum of the values of the error function at the closest estimated beat of each annotation, normalized by the maximum between the number of annotations and the number of estimated beats. In the **PScore**, the beat accuracy is determined by taking the sum of the cross-correlation between two impulse trains, one representing the annotations and the other representing the extracted beats. **Goto** [26] proposes measuring the performance by evaluating the proportion of time in which the beat is correctly tracked. The classification of the beats as correctly tracked or not relies on statistical properties of the difference to the annotated beats . **CMLc**, **CMLt**, **AMLc**, **AMLt** are continuity-based performance indicators computed over the correctly tracked regions. The CMLc and the CMLt are defined as the ratio of the longest continuously tracked region or the total length of correctly tracked regions, respectively, to the total signal length. The AMLc and AMLt are defined in a similar way, but are more permissive in the definition of correctly

tracked beat, allowing off-beat estimation and beats at the double and half of the correct metrical level. To calculate the **Information Gain**, two timing error histograms are constructed, one between the annotated beats and the estimated beats and vice-versa. Then, the Information Gain is the minimum of the Kullback-Leibler divergence between the previously calculated histograms and an uniformly distributed histogram.

In order to avoid unfair comparisons, we searched for the optimum performance of each algorithm via a grid search over its respective parameters.

Results

	Original	Median	SSE
F-Measure (%)	48	48,5	51.4
Cemgil (%)	35,6	35,5	37,5
Goto (%)	6,75	7,25	7,24
PScore (%)	50,8	50,9	53
cmlC (%)	10,2	10,3	11,2
cmlT (%)	18,3	18,4	20,7
amlC (%)	19,3	20,8	22,4
amlT(%)	36,8	39,7	43,1
InfGain (bits)	1,2	1,3	1,4

Table 4.2: Average beat-tracking performance measure.

The average results of the evaluation over the complete data set are presented in Table 4.2, while the corresponding histograms can be found in annex C. It is worth to point-out that the performance with preprocessing improves the beat detection for almost all performance measures. The improvement is small but consistent. In the comparison between separation methods, the proposed SSE filter outperforms the Median filter in all the measures except one of them.

More detailed information can be derived from the results presented in Figure 4.1. It depicts one of the measures for beat-tracking performance evaluation for the complete data set. The transient preprocessing highly increases performance in the case of files train12 and train18 while the SSE filtering yields much better results in the case of files train10 and train13.

The Figure 4.2 illustrates the different outputs of the beat-tracking algorithm for the files train10 and train13 along with the manually annotated beats. It can be seen in Figure 4.2a that when the beat-tracker input is the unprocessed signal or the median filtered transient components signal, the beat-tracking algorithm locks into a metric level four time faster than the manual annotated beat, while when the signal is filtered by the SSE the metric level and phase match the annotated beat.

In the example of Figure 4.2b the metric level detected by the beat-tracking algorithm coincides with the manually annotated beat. But, the beats produced

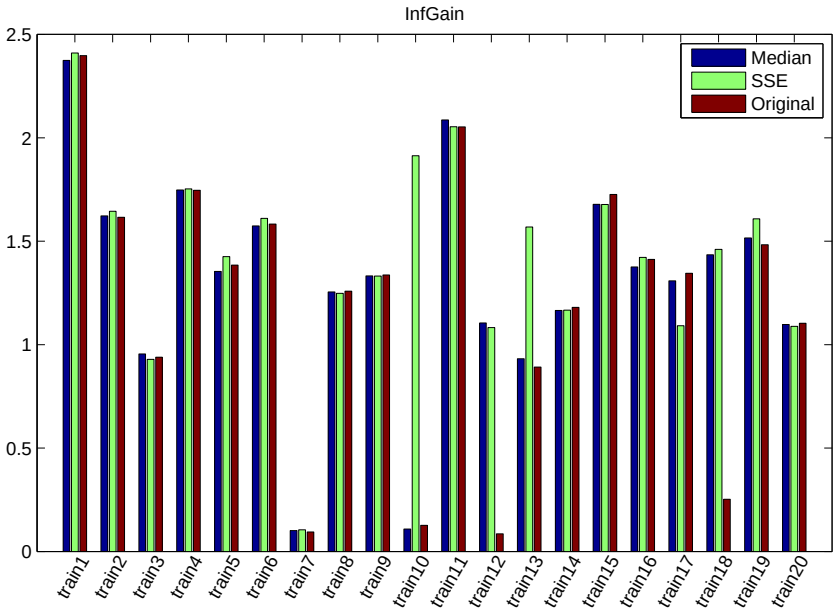
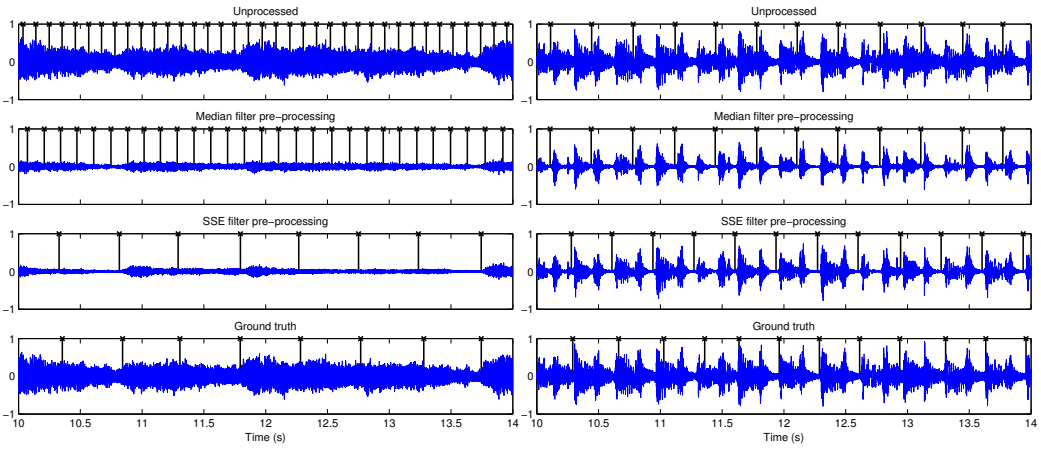


Figure 4.1: One of the performance measurements for beat tracking (Information Gain); comparison for all elements in the data set.



(a) Beat-tracking output for train10. (b) Beat-tracking output for train13.

Figure 4.2: Beat-tracking outputs.

by the algorithm in the original and the median filtered cases are in counter-phase with the manually annotated beats. By contrast, the beat-tracking output for the SSE filtered signal matches both, the metric level and the phase.

4.1.4 Pitch-tracking

Pitch can be defined as the perceptual attribute which allows the ordering of sounds on a frequency-related scale. It can be objectively measured asking a listener to

match the frequency of a sine wave with the tone of the target sound.

In [35] an Harmonic/Percussive separation algorithm (Harmonic Percussive Sound Separation - HPSS) is utilized as the first stage in a singing pitch extraction algorithm, and good results are reported. The HPSS algorithm presented in [59] is also utilized in [77] to estimate the melodic line.

The "tonal" information can be assumed to be carried by the steady-state components of a signal. Analogously the performance improvement of beat-tracking algorithms when tonal information is discarded, a pitch-tracking algorithm can be favoured by the elimination of the transient components. We evaluate the advantage of pre-processing a signal to eliminate the transient components before performing the pitch-tracking, and compare the median filter to the proposed SSE filter.

A state-of-the-art pitch-tracking algorithm presented in [69] was utilized. This algorithm has been developed as a VAMP plug-in for the semantic visualization software Sonic Visualizer [7]. A brief description of the algorithm follows.

Pitch-tracking algorithm

The MELODIA plug-in extracts the principal melody and respective F0 contour of polyphonic audio. The method is comprised of four main blocks:

The Sinusoid Extraction stage begins with an equal loudness filter to mimic the human auditory system sensitivity. It enhances the mid frequencies and attenuates low frequencies. Then, the STFT of the result is computed with window length $M = 2048$, FFT length $N = 8192$ and hop size $H = 128$. Then, the estimation of spectral peak frequencies is improved by a phase-vocoder based technique [40].

The previously calculated peaks are employed to compute the salience function in the range from 55 Hz to 1760 Hz. For a given frequency, the salience function is defined as a weighted sum of its harmonics' energies.

For each frame of the salience function, the peaks are selected as potential F0 candidates. Then the peaks are grouped into continuous pitch contours. Each contour has a limited time span and corresponds to a short phrase. Firstly, the peaks are filtered by hard thresholding. Later, this peaks are grouped into contours using heuristics and auditory analysis cues.

A set of contour characteristics are calculated from the previously generated contours to determine which ones belong to the melody. The characteristics are: pitch mean, pitch deviation, contour mean salience, contour salience deviation, length and vibrato presence. Those contour characteristics are then utilized to filter out the non melodic contours. Finally, the F0 melodic contour is selected from the remaining contours.

Performance measures

The evaluation metrics adopted are the ones used in the MIREX 2005 melody transcription task, which are reviewed and described in [61]. The **overall transcription accuracy (OTA)** combines the pitch transcription and the voicing detection task. It is defined as the proportion of frames correctly labelled with

raw pitch accuracy and voicing detection. The **raw pitch accuracy (RPA)** is defined as the fraction of frames in which the estimated pitch is within a quarter tone from the reference. The **raw chroma accuracy (RCA)** is equal to RPA but allows octave transpositions. The **voicing detection rate (VDR)** is the proportion of estimated voiced frames from the total of voiced frames in the reference. The **voicing detection false alarm rate (VDFAR)** is the proportion of unvoiced frames labeled as voiced by the algorithm.

	Original	SSE	Median
OTA (%)	67	65,9	65.9
RPA (%)	73	71,5	71,3
RCA (%)	73,3	72,2	72,1
VDR (%)	77,6	77,2	76,9
VDFAR (%)	13,7	13,7	14,5

Table 4.3: Average pitch-tracking performance measure.

Results

The Table 4.3 shows the averages of the pitch-tracking performance indicators. The effect of eliminating the transient components decreases the pitch-tracking average performance measures by approximately 1%.

A more detailed analysis shows that the problem arises from the hypothesis that the principal melody corresponds to a steady-state behaviour. In fact, the melody line exhibits pitch variations that are far from being steady-state. Thus, the preprocessing stage eliminates some useful information from the tonal components. As an example, the pitch-tracking OTA for the file opera_male5 are 78.2%, 57.4% and 59.3% for the original, SSE filtered and Median filtered signals respectively. Figure 4.3 depicts the pitch-tracking algorithm output, and also the intermediate stage signals for this signal. The most noticeable difference between the original and the SSE filtered tracking is the octave error between seconds 9 to 13, not present in the tracking with the original sound. Simple approaches can be thought to solve some of this errors. However, the decrease in the RCA for the filtered signals shows that besides the above mentioned, other types of errors are introduced.

The melody in the opera_male5 example presents a deep vibrato, which is not correctly modeled as a steady-state component. The method proposed in section 3.3.2 that accounts for some variations in the frequencies of the steady-state component can improve the separation and also the pitch-tracking performance.

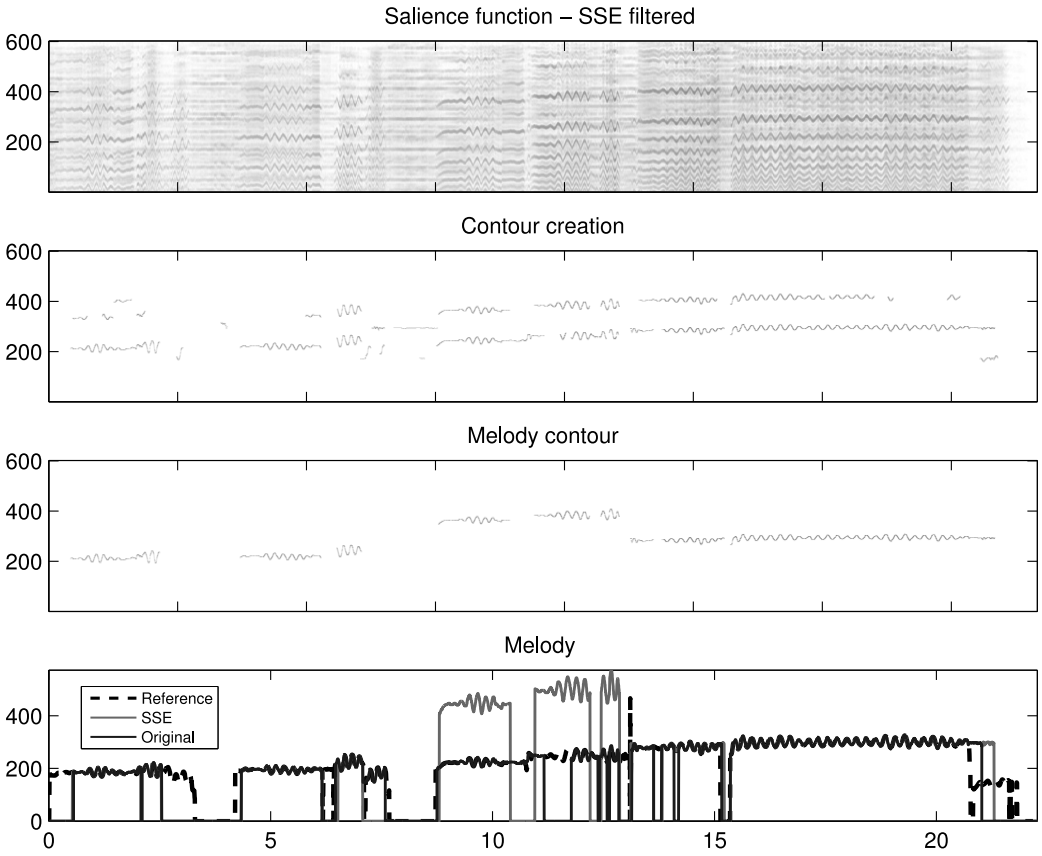


Figure 4.3: Steps of the pitch-tracking MELODIA; involved signals.

4.2 Applications in audio editing

The transient and steady-state decomposition can be helpful in audio editing tasks. In the following sections some audio editing applications are described.

4.2.1 Removing undesired transients

A simple idea is that the separation in transient and steady-state components can simplify a difficult edition task. A real-life example of this is the removing of an unintentionally hit on the guitar body when a sustained note is being played. This undesirable sound is very difficult to edit, and in general when it is possible, a re-recording of this passage is necessary. The difficulty arises from cutting the hit noise while leaving unaltered the sustained guitar note. As depicted in figure 4.4 the undesired sound its just noticeable in the original waveform, but the characteristics of the scene make it clearly audible. The decomposition makes the edition process straightforward; the noise can be properly silenced while leaving untouched the guitar sound.

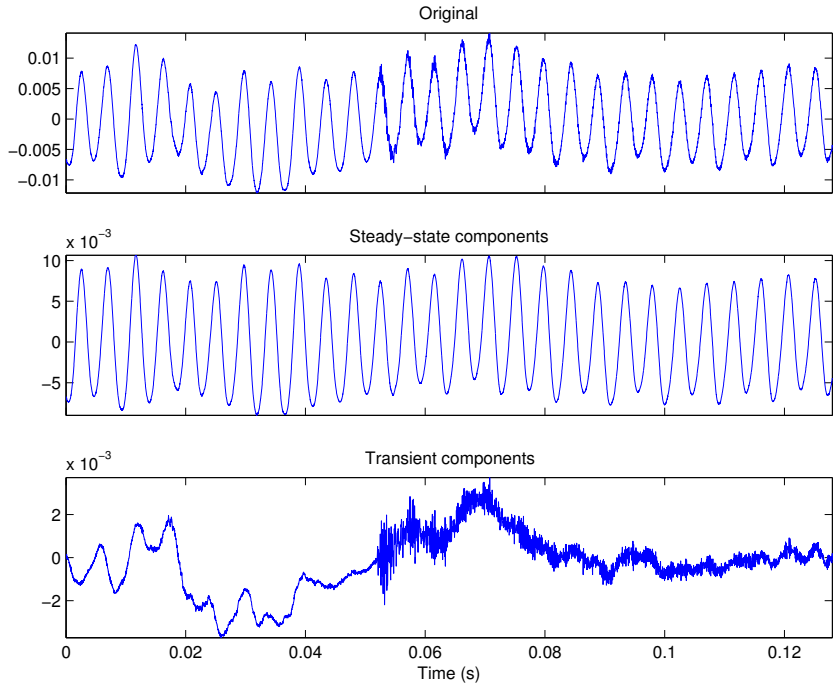
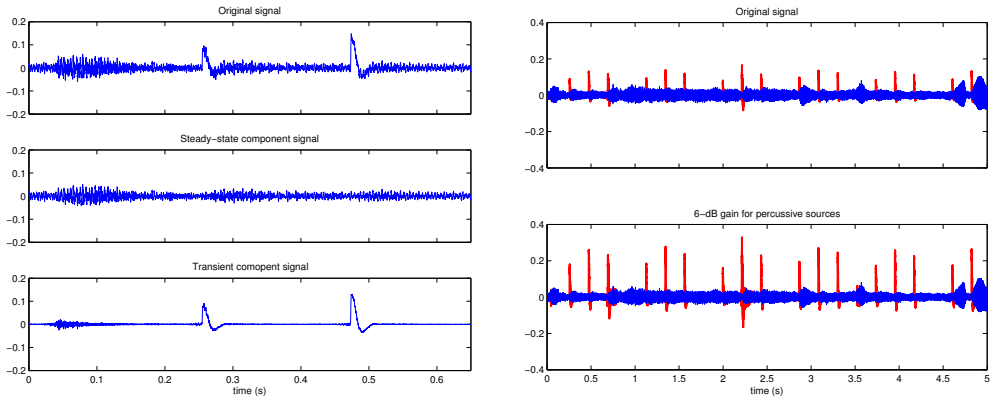


Figure 4.4: Transient/steady-state separation for manual editing.

4.2.2 Percussion extraction and remixing

The remixing process can be defined as the alteration of an edited song to create a new version, that sounds different in some sense, e.g. create stereo mixes from original mono tapes, adapt for radio broadcast, alter the song to reach different



(a) Transient and steady-state separation. (b) Transient component with 6dB gain.

Figure 4.5: Remixing application.

audiences, etc. Ideally, the raw material for remixing are the original multi-track records. Often, this records are non-available or do not even exist, as in the case of the recordings of the first half of the twentieth century. In some other cases, the number of tracks are very limited, for example, the first two Beatles' albums were recorded with two-track machines, since the eight-track recorders were introduced only in 1968 [46].

The original tracks can be approximated by audio source separation algorithms [84, 91],which in general need manual tuning and the previous knowledge of mixed sources, thus becoming not suitable for fully-automated processing. The transient/steady-state decomposition presented in the previous chapter seems to be appropriate for percussion track extraction. Due to the short attack time of the percussive instruments's sound, the spectral content of percussive instruments is broadband [1], thus, correctly modeled by the transient definition utilized along this work.

Figure 4.5a depicts a detail of the decomposition of an audio signal, and figure 4.5b shows a signal decomposition and a remixed version after applying 6dB of gain to the transient components. Along with the percussive sounds, some note attacks of non-percussive instruments are modeled as transients, as depicted in figure 4.5a. Although the separation of percussive instruments is not perfect, it is adequate for remixing. The requirements for separation in the remixing context are less strict than for pure source separation, as the introduced artifacts may often be masked by the mix itself.

4.2.3 Noise Reduction

Noise reduction is an essential tool in different areas as communications, recording and restoration where corrupted or noise-contaminated audio material is involved. Spectral subtraction is the classic technique for stationary-like noise removal [3, 21], meanwhile non-linear filters are used for transient and impulse-like noise removal [38, 41, 82].

The noise reduction process is not transparent; depending on the amount of noise and filter settings, some noticeable artefacts may appear. Undesirable effects such as pre-echo and transition smearing may happen when one applies stationary noise removal techniques. This effects can be mitigated by decomposing the signal and applying noise reduction techniques for stationary-like noise to the steady-state component signal, and impulse noise removal techniques only to the transient components.

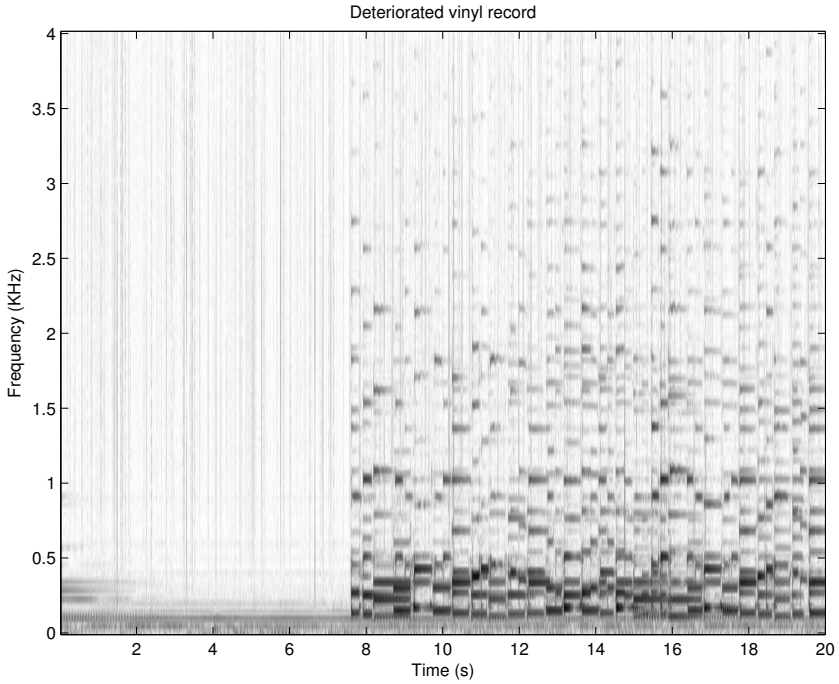


Figure 4.6: Spectrogram of a vinyl record.

Figure 4.6 shows a spectrogram of a digitized vinyl record. The horizontal ridges are clicks probably produced by dust in the grooves while the low frequency stationary noise is due to electrical hum. In Figure 4.7 the spectrogram of the decomposition in transient and steady-state components and their correspondent noise-reduction processed spectrogram are shown. The clicks are present in the steady-state components while the electrical hum is present in the steady-state component, as expected. Applying this work-flow is expected to be more resilient to artifacts; a formal assessment of how much the pre-decomposition can improve the noise-removal methods needs to be done yet.

4.2.4 Transient shaping

Another application of the transient/steady-state decomposition is commonly called in the audio production context by commercial names as Transient Shaper [79], Transient Designer [55]. We denote transient shaper to the process of adjusting the dynamic of a sound, but unlike the dynamic compressors, transient shapers

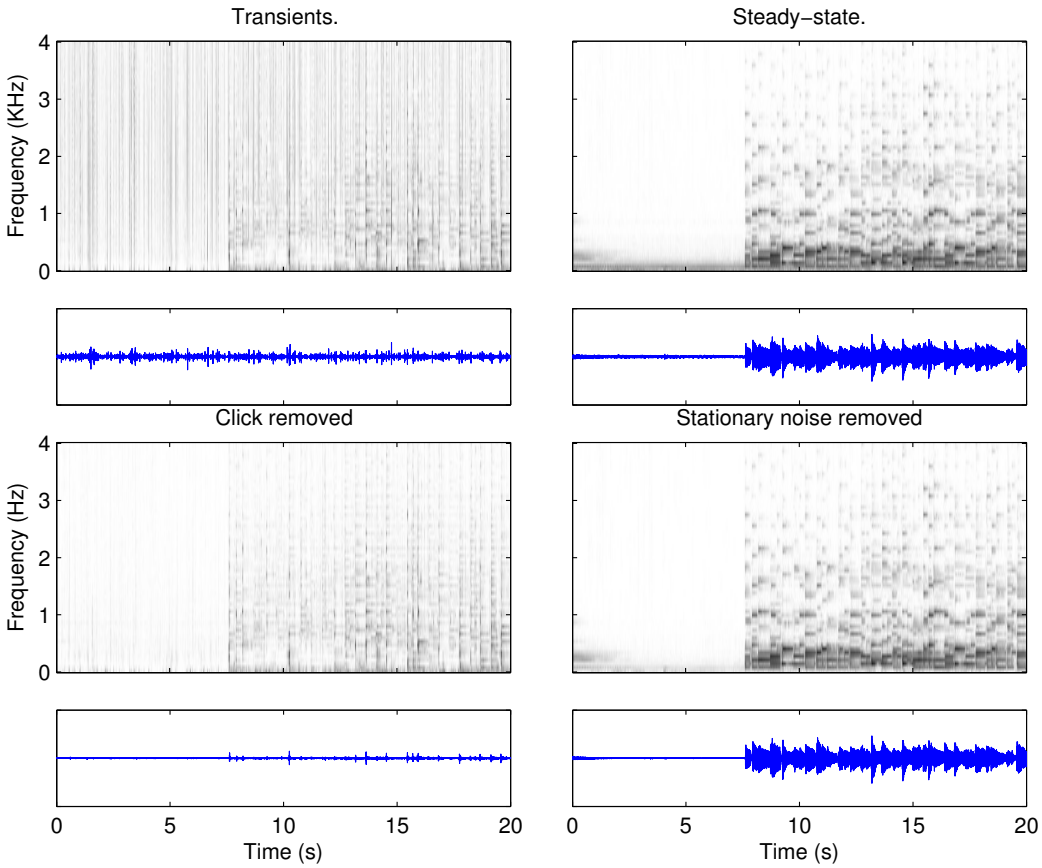


Figure 4.7: Noise reduction.

operates only when a sudden change in energy is present. It has generally two parameters, the transient gain (or attack gain) and the sustain gain (or release gain), and it is often applied in the mixing process to precisely control the amount of attack of drum sounds. The process can be applied to non-percussive instruments such as piano and electric guitar. In the latter case, for example, the transient shaper controls the sound of the pluck.

Transient/steady-state decomposition can be utilized as a transient shaper, as they exactly decompose the signals in that way. Adjusting the transient component level, is what the industry calls attack gain, and adjusting the steady-state gain correspond to the sustain gain. Unlike dynamic compressors, the transient and steady-state decomposition is level-independent.

4.2.5 De-reverberation

The reverberation is the effect of the multitude of echoes arriving from reflections on the surrounding space. Although a certain amount of reverberation is considered pleasant to human perception, and even artificial reverberation is commonly used as an artistic effect in records and live concerts, excessively reverberated

rooms are troublesome. In such cases speech intelligibility is decreased [42], and hinders the accuracy of automatic note onset detection algorithms [89].

The process of reverberation removal is called de-reverberation, and when no information of the acoustic impulse response is available it is called blind de-reverberation. Various methods exist to accomplish de-reverberation as can be seen in [48, 56, 58, 80, 92, 88, 44].

Reverberations of transient components (considered as broad-band components with highly concentrated energy in time) has a steady-state behaviour (in the sense of smooth temporal evolution). Then for example, applying a transient/steady-state decomposition to a reverberated drum tends to segregate the reverberations (wet sound) into the steady-state signal and the direct (dry) sound into the transient signal. In the Figure 4.9 the decomposition of a synthesized drum sound with artificially applied reverb is shown.

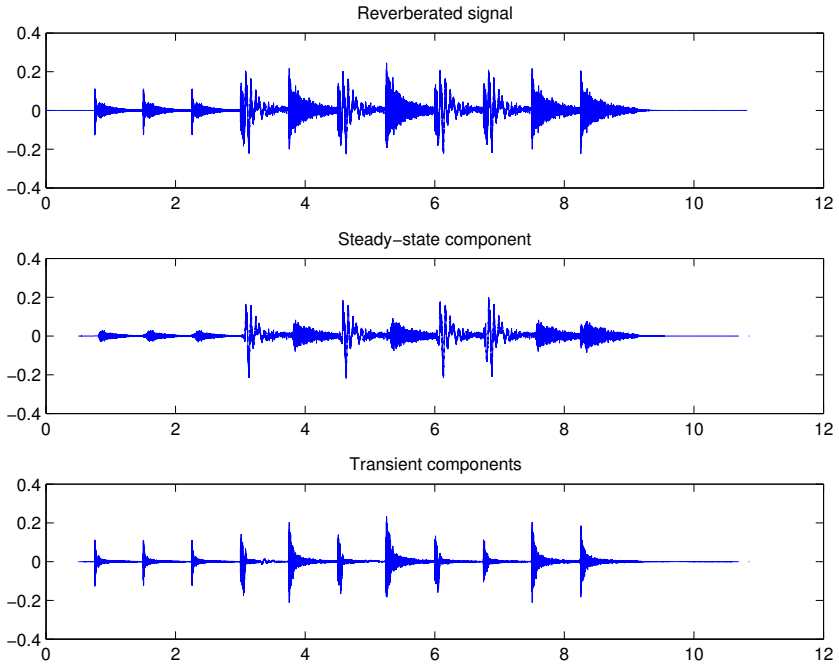


Figure 4.8: De-reverberation of a drum sound.

By controlling the mixing of transient and steady-state components signal, the amount of de-reverberation can be adjusted. The effectiveness of the method depends on the kind of reverberation. The results improves as the decaying time of the reverberations increases.

4.2.6 Time/Pitch Modifications

Playing a record at double speed reduces the playtime to half, and at the same time the sound pitch elevates an octave. Similarly, playing a record at slower rate makes the record duration longer and lowers the sound pitch. In some cases, such as music transcription, audio synchronisation and language translation, it

is desirable to change the speed of a record without altering the pitch. In pitch correction and harmonization for example, the opposite effect of changing the pitch without altering the articulation is desirable. This technique is utilized for artistic purposes in electro-acoustical musical composition [90] and in popular music.

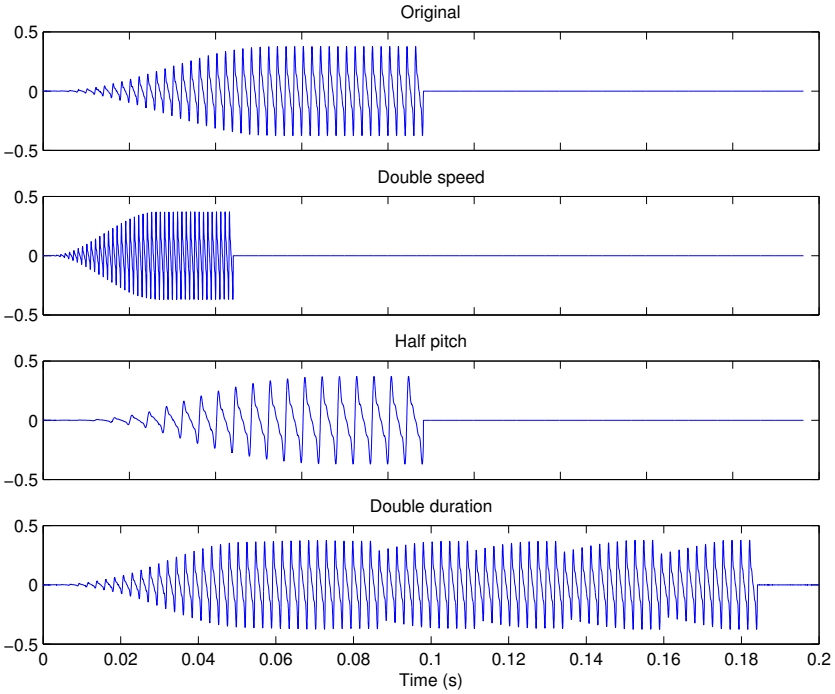


Figure 4.9: Different time-pitch modifications.

Time-domain [67, 9] and frequency- domain algorithms [62, 54, 52] exist to perform those tasks (see [57] for a complete survey). The phase vocoder presents some undesirable effects, in particular a smearing effect appear when processing transients as depicted in Figure 4.10. Various methods were proposed to mitigate this problem [66, 19]. As suggested in [17] the transient/steady-state component decomposition is suitable to avoid transient smearing in the time-scale modifications. The idea is to leave the transients unprocessed while applying a classic Time/Pitch modification algorithm to the steady-state signal.

The results are shown in Figure 4.11,where one can see that the transient components are preserved while the steady-state components are effectively pitch-shifted.

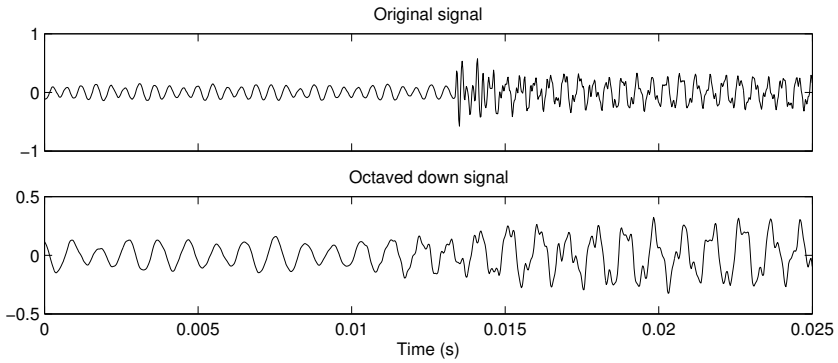


Figure 4.10: Transient smear in an octave-down pitch-shift.

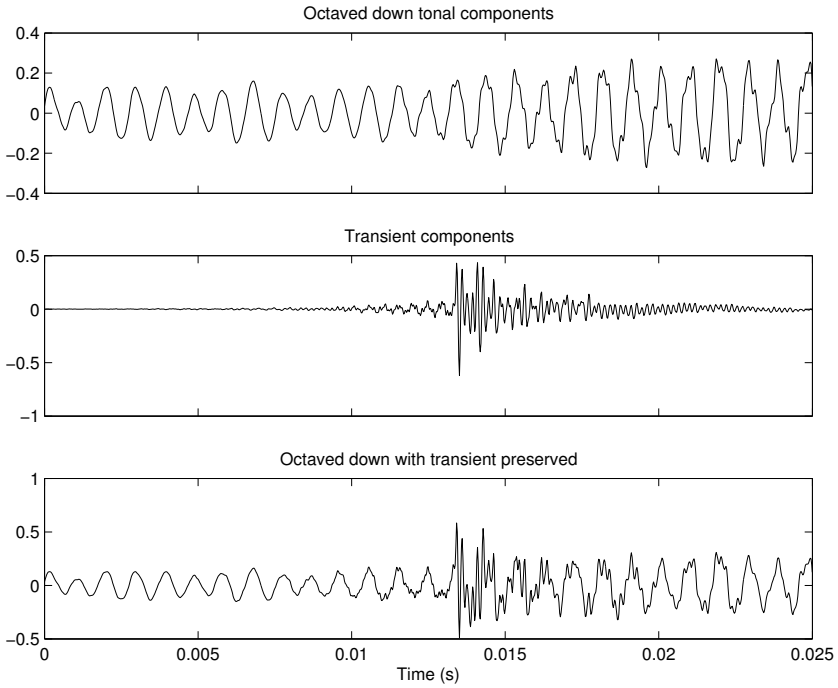


Figure 4.11: Not smeared transient in a octave-down pitch-shift.

Chapter 5

Conclusions and future work

5.1 Conclusions

In the preceding chapters, the transient/steady-state component separation problem was studied. In particular, a recently proposed method for separating transient and steady-state components based on the median filter [23] was investigated. For a best understanding of the processes involved, a modification of the filtering stage of the algorithm was proposed. The modification consisted in utilizing a non-linear filter, originally devised to perform Stochastic Spectrum Estimation (SSE) [43], as the filtering stage instead of the median filter.

To evaluate the perceptual quality of the decomposition, subjective listening tests were designed and conducted. These tests allowed a perceptual comparison between the original and the proposed filtering stage. The results indicated that there are no significant perceptual differences among them, both providing the same quality of separation.

Besides, an application-based comparison was performed by applying the transient and steady-state decomposition as a pre-processing stage for two Music Information Retrieval problems, namely the pitch-tracking and beat-tracking tasks. The hypothesis that the transient components carry the beat information while the steady-state components carry the pitch information was tested. The evaluation showed that the beat-tracking can benefit from the transient components extraction, notably increasing the performance in some cases. On the other hand, the pitch-tracking algorithm did not improved its performance by applying the preprocessing. This seems to indicate that the partials of the melody line are not properly captured as steady-state components, due to its frequency fluctuations. Part of this work was presented in a conference paper [36].

To overcome some of the model limitations that were identified, three extensions were delineated. First, an iterative filtering scheme was proposed, which implicitly defines the residual components as non-steady-state and non-transient components. In other proposed extension, the definition of steady-state components was relaxed to allow for slow variations in partials' evolution. This effect is typical of, but not limited to, free intonation instruments. Finally, to allow more flexibility in the choice of the parameters of the algorithm, a sub-band processing technique was proposed. For each band, the window length, the hop size and the length of the non-linear filters can be adjusted for a fine-tuning of the separation.

Another part of this work was to study possible applications of the decomposition into transient/steady-state components in the area of audio editing and processing. Six applications were surveyed and illustrated with real audio examples. With this technique a very complex editing task such as the removal of undesired transients in presence of steady-state sounds can be done in a simple way. The transient/steady-state separation can also be used as a preprocessing stage to increase the performance of some audio processing techniques. For instance, it allows the application of specific noise-reduction algorithms to each of the obtained components, such as stationary noise-removal techniques on the steady-state signal, and impulse-like noise removal techniques on the transient signal. Since each noise-reduction technique is applied to a signal which mainly

contains the type of component for which it was devised, the number of undesired artifacts is reduced. Analogously, the transient smearing effect that appears when time/pitch modifications are applied can be avoided by modifying only the steady-state components while leaving the transients unchanged. In the other cases, by controlling the amount of transient and steady-state components in a mixture, different audio editing applications can be implemented. This was illustrated by some examples, namely de-reverberation of percussive sound, transient shaping and percussive sound extraction for remixing.

5.2 Future perspectives

All along this work, various topics that deserve further investigation were identified at different levels.

Although the STFT was the TF representation used in this work, it has some drawbacks, such as constant time-frequency resolution. Therefore, other type of representation may be more suitable depending on the type of signal. For example, adaptive representations such as Fan Chirp Transform [6] seem to be useful when harmonic signals with pitch fluctuations are involved.

On the other hand, the original definition of steady-state component proved to be too restrictive in some cases. An extension of the model was proposed in this work to take into account pitch fluctuations commonly present in music signal such as the ones produced by glissando and vibrato. Its validity was illustrated by synthetic examples as a proof of concept, but further investigation is necessary. In future work, a systematic evaluation of the extensions proposed in Section 3.3 will be conducted.

More audio processing applications for the transient/steady-state decomposition could be developed. For example, lossy audio coding of transients tends to generate undesired pre-echo effects that can be minimized by applying different coding schemes to the transient and the steady-state components [93, 87]. Furthermore, it will be interesting to develop a real-time implementation of the algorithm suitable for live performances. The algorithm latency in the current implementation is in the order of 100 ms, which is not suitable for real-time. A low-latency causal version of the algorithm will be developed in future work.

Finally, further developments of the present technique in collaboration with professional musicians will be encouraged. To do that, it is necessary to build user-friendly applications within popular frameworks such as Pure Data [64], Steinberg Virtual Studio Technologies [78] or LADSPA [25]. Musicians can contribute by finely evaluating the quality of the separation and also by proposing novel applications.

Appendix A

Subjective Quality Test Data

Steady-state/transient component separation

Listener 1

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	**Q2**	**Q3**	**Q1**	**Q2**	**Q3**	**Q1**	**Q2**	**Q3**	**Q1**	**Q2**	**Q3**
Fake	61	51	58	65	51	58	32	28	30	34	28	32
1	75	59	70	73	58	68	58	27	25	54	26	26
2	51	56	37	49	55	34	60	38	43	58	39	45
3	73	51	70	73	51	71	62	60	32	73	62	35
4	46	49	42	52	48	42	41	30	38	42	32	38
5	57	47	51	56	46	50	54	37	47	57	38	47
6	59	66	73	58	66	69	49	44	49	50	47	51
7	75	63	84	74	63	85	47	50	67	75	63	74
8	81	79	76	81	78	74	58	45	63	57	43	61
9	82	70	72	82	68	72	69	64	61	69	64	59
10	55	37	52	56	40	52	59	55	76	59	53	75
11	51	67	68	50	69	66	62	44	42	48	35	41
12	50	51	45	50	51	44	54	38	43	55	40	44
13	58	52	55	58	52	55	69	49	37	58	45	39

Appendix A. Subjective Quality Test Data

Listener 2

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	71	65	63	69	29	80	77	88	21	77	71	26
1	90	65	100	81	76	100	72	37	47	59	22	24
2	64	62	70	29	75	29	82	46	41	68	34	55
3	76	79	75	85	78	52	86	81	71	79	71	70
4	91	55	87	82	56	73	67	49	22	81	55	20
5	85	87	100	85	78	87	89	71	73	81	79	60
6	83	80	65	83	85	72	85	69	74	86	84	63
7	80	100	67	79	81	85	92	92	91	79	79	77
8	77	90	100	79	100	100	84	67	58	100	80	50
9	100	86	82	98	92	63	82	79	82	72	56	57
10	84	74	92	84	85	77	84	82	90	100	85	100
11	83	77	75	100	88	67	82	62	55	81	72	30
12	84	82	100	94	83	91	80	72	67	80	83	60
13	88	78	86	90	69	100	69	72	84	67	81	68

Listener 3

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	44	38	45	44	61	50	41	48	32	54	48	40
1	46	50	50	45	49	50	49	46	37	38	49	37
2	58	24	40	46	22	35	50	44	43	57	46	56
3	46	40	43	45	40	43	67	69	66	67	64	67
4	53	44	56	41	44	50	48	37	60	47	37	60
5	50	46	48	48	48	48	46	66	68	47	42	50
6	49	49	58	38	50	49	48	54	52	49	55	50
7	56	49	46	38	48	45	43	39	42	57	63	55
8	55	55	55	34	45	44	47	52	51	48	45	38
9	56	53	56	45	45	48	45	44	48	46	50	54
10	31	28	28	26	21	24	46	48	70	44	47	73
11	41	41	36	42	41	37	44	50	45	46	41	41
12	44	45	46	45	45	46	51	49	48	51	49	50
13	38	50	42	53	35	41	42	38	43	41	56	52

Listener 4

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	66	34	22	79	29	39	33	49	0	33	44	0
1	65	37	47	83	36	69	36	72	20	34	69	21
2	50	30	25	30	30	21	47	35	41	30	50	30
3	50	72	39	51	56	59	61	48	44	65	51	45
4	47	61	34	47	80	18	61	43	31	58	41	29
5	50	68	43	29	80	23	29	63	28	27	64	27
6	86	16	88	86	17	87	66	71	29	56	72	29
7	97	0	43	96	0	41	57	42	40	68	44	40
8	53	65	30	53	35	40	68	35	52	66	52	53
9	60	50	52	72	41	50	51	69	37	51	70	37
10	38	65	12	39	67	12	68	38	56	66	39	54
11	76	53	31	83	50	29	60	62	42	52	59	43
12	82	55	65	90	56	70	65	63	41	59	62	43
13	57	43	50	69	42	53	58	53	54	68	44	57

Listener 5

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	0	44	0	48	0	50	8	0	0	27	24	26
1	49	0	54	0	54	0	30	0	25	0	0	0
2	0	50	51	57	0	0	0	0	26	30	27	0
3	0	50	0	50	0	48	0	32	0	24	0	28
4	0	54	0	50	0	51	31	0	32	0	34	0
5	0	0	0	0	0	0	23	0	30	0	30	0
6	49	0	54	0	55	0	0	31	0	31	0	33
7	0	56	0	49	0	47	0	29	0	29	0	29
8	0	0	0	0	0	0	0	26	0	29	0	25
9	0	47	0	51	0	50	30	0	28	0	30	0
10	0	45	0	42	0	58	0	26	0	28	0	28
11	0	60	0	50	0	47	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	31	0	30	0	35	0

Appendix A. Subjective Quality Test Data

Listener 6

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	100	100	100	63	56	100	51	21	42	52	35	58
1	100	26	77	57	26	79	56	38	50	69	37	76
2	0	0	40	47	40	0	46	23	39	65	22	59
3	40	50	62	45	66	59	72	36	63	58	26	26
4	0	0	69	22	63	70	24	35	34	26	35	72
5	76	75	53	80	83	56	0	71	30	0	54	36
6	67	100	45	53	100	28	50	34	46	37	40	30
7	100	100	43	100	100	24	44	20	0	62	18	18
8	67	18	0	81	30	11	42	39	49	31	36	36
9	57	39	68	46	40	79	47	56	32	47	61	44
10	65	81	47	50	70	40	47	24	45	71	22	69
11	55	75	100	55	74	100	55	17	70	55	17	73
12	100	92	66	100	100	62	47	61	65	70	58	38
13	77	50	76	76	23	100	62	21	70	63	19	71

Listener 7

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	100	100	100	63	56	100	51	21	42	52	35	58
1	80	23	74	50	26	79	56	38	50	69	37	76
2	0	0	40	47	40	0	46	23	39	65	22	59
3	40	50	62	35	66	59	72	36	63	58	26	26
4	0	0	69	21	63	70	24	34	34	26	35	72
5	76	75	52	79	83	56	0	71	30	14	54	36
6	67	100	3	53	100	28	50	34	46	37	40	30
7	100	100	43	100	100	24	44	20	0	62	18	18
8	67	18	0	81	30	11	42	39	49	31	36	36
9	57	39	68	46	40	79	47	56	32	47	61	44
10	65	81	47	50	70	40	47	24	45	71	22	69
11	55	75	100	55	74	100	55	17	70	55	17	73
12	100	92	66	100	100	62	47	61	65	70	58	38
13	77	50	76	76	23	100	62	21	70	63	19	71

Listener 8

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	0	100	4	19	87	0	0	63	73	0	0	0
1	0	77	0	0	95	17	0	37	0	92	0	80
2	0	27	0	24	69	32	0	0	0	0	39	0
3	34	84	31	21	69	19	29	0	67	0	55	0
4	26	92	65	0	28	39	0	58	97	53	0	0
5	0	42	18	52	68	73	0	41	57	0	0	0
6	58	32	34	21	77	37	0	0	0	49	16	0
7	78	87	53	48	48	41	0	38	0	0	0	33
8	0	0	0	8	21	24	0	28	0	0	0	20
9	58	66	66	33	66	50	0	0	0	0	28	21
10	0	0	63	0	0	0	0	37	0	24	0	0
11	0	0	0	34	78	55	0	27	40	0	0	0
12	0	0	0	32	32	32	0	54	51	41	0	0
13	0	0	0	21	21	0	0	0	0	0	0	0

Listener 9

Audio	Steady-state						Transient					
	Median Filter			SSE Filter			Median Filter			SSE Filter		
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Fake	0	0	0	0	0	0	80	19	52	80	34	53
1	0	40	0	47	47	0	83	24	56	82	23	56
2	0	0	0	51	0	0	92	52	64	91	33	62
3	66	0	0	63	0	0	80	50	55	80	51	55
4	64	0	0	0	0	0	82	61	65	82	60	65
5	0	0	0	37	0	0	62	74	79	77	75	79
6	0	0	0	0	0	0	100	90	88	100	92	87
7	0	0	0	0	0	0	82	74	57	82	47	59
8	0	0	0	39	0	0	68	44	74	67	43	72
9	48	0	0	0	0	0	96	82	91	95	58	91
10	50	0	0	0	0	0	71	37	63	70	36	61
11	0	0	0	50	0	0	88	69	76	86	67	74
12	0	0	0	56	0	0	69	53	55	70	59	54
13	0	0	0	0	0	0	0	0	0	0	0	0

Listener 10

Audio	**Steady-state**						**Transient**					
	Median Filter			**SSE Filter**			**Median Filter**			**SSE Filter**		
	Q1	**Q2**	**Q3**	**Q1**	**Q2**	**Q3**	**Q1**	**Q2**	**Q3**	**Q1**	**Q2**	**Q3**
Fake	14	18	41	41	61	54	41	48	34	54	48	90
1	42	51	54	42	59	51	49	46	37	40	49	41
2	58	24	40	46	22	33	50	44	43	52	46	56
3	48	40	43	45	40	43	63	63	63	67	64	67
4	52	44	56	41	44	51	48	37	60	47	37	60
5	51	46	48	48	48	41	46	66	68	47	42	50
6	44	49	58	38	50	49	48	54	52	49	55	50
7	54	49	46	38	48	45	43	39	42	57	63	55
8	55	55	55	34	45	44	47	52	51	48	45	38
9	56	53	56	45	45	48	45	44	48	46	50	54
10	31	28	28	26	21	24	46	48	70	44	47	73
11	41	41	36	42	41	37	44	50	45	46	41	41
12	44	45	46	45	45	46	51	49	48	51	49	50
13	38	50	42	53	35	41	42	38	43	41	56	52

Appendix B

Subjective Quality Test Interface

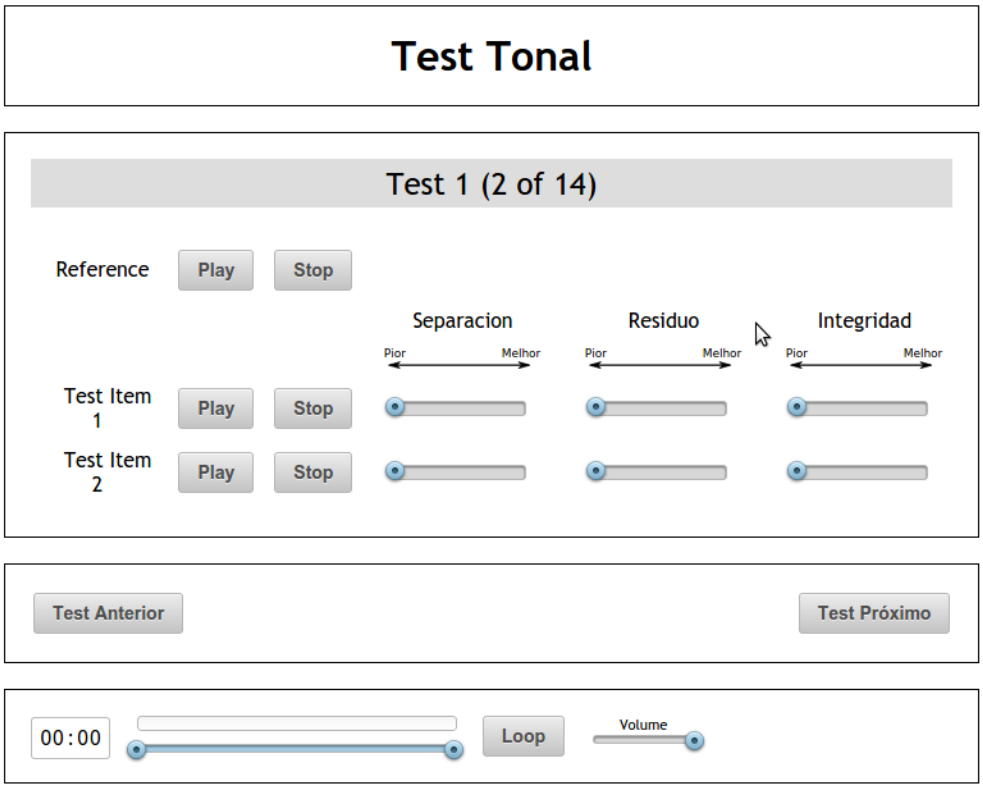
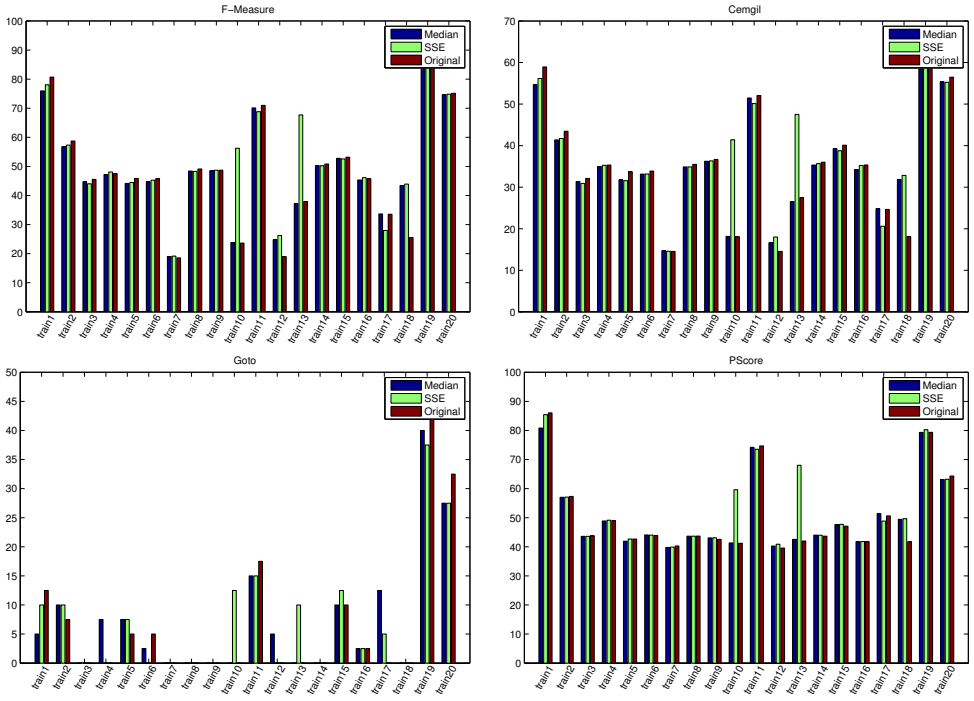


Figure B.1: Subjective listening test web interface.

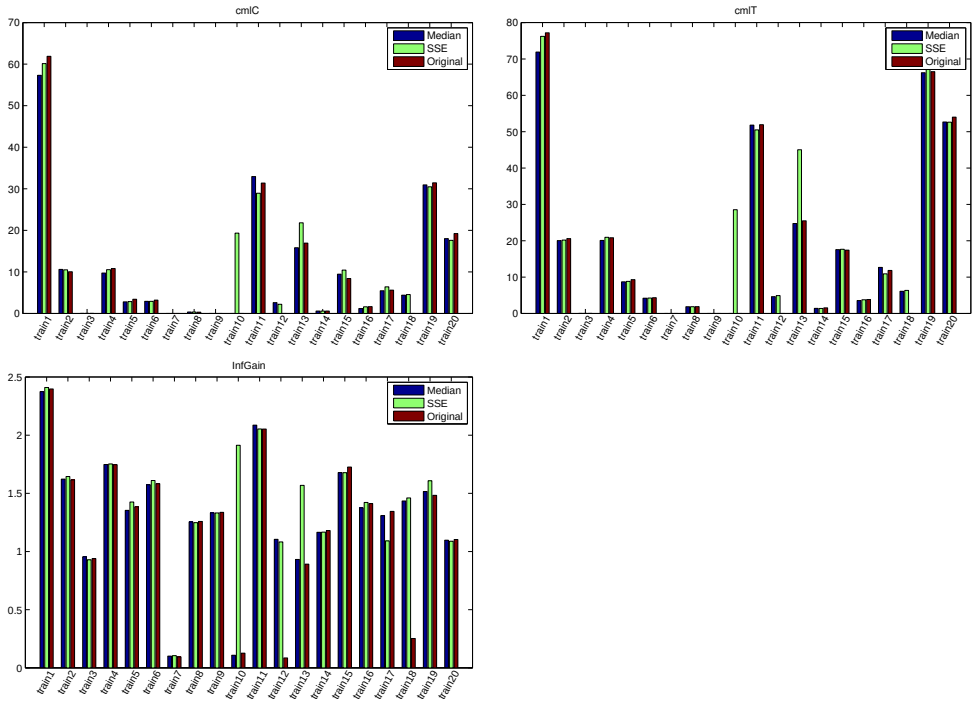
Appendix B. Subjective Quality Test Interface

Appendix C

Complete Beat-Tracking results



Appendix C. Complete Beat-Tracking results



Bibliography

[1] Mitsuko Aramaki et al. "Synthesis and perceptual manipulation of percussive sounds". In: *International Computer Music Conference Proceedings*. ICMA. Barcelona, Spain, Sept. 2005, pp. 335–338.

[2] F. Auger and P. Flandrin. "Improving the readability of time-frequency and time-scale representations by the reassignment method". In: *IEEE Transactions on Signal Processing* 43.5 (May 1995), pp. 1068–1089.

[3] Steven Boll. "Suppression of acoustic noise in speech using spectral subtraction". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 27.2 (Apr. 1979), pp. 113–120.

[4] Judith C. Brown. "Calculation of a constant Q spectral transform". In: *The Journal of the Acoustical Society of America* 89.1 (Jan. 1991), pp. 425–434.

[5] Judith C. Brown and Miller S. Puckette. "An efficient algorithm for the calculation of a constant Q transform". In: *The Journal of the Acoustical Society of America* 92.5 (Nov. 1992), pp. 2698–2701.

[6] Pablo Cancela, Ernesto López, and Martín Rocamora. "Fan chirp transform for music representation". In: *Proceedings of the 13th International Conference on Digital Audio Effects (DAFx-10)*. Graz, Austria, Sept. 2010, pp. 1–8.

[7] Chris Cannam et al. "The sonic visualiser: A visualisation platform for semantic descriptors from musical signals". In: *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*. ISMIR. Victoria, Canada, Oct. 2006, pp. 324–327.

[8] Ali Taylan Cemgil et al. "On tempo tracking: Tempogram representation and Kalman filtering". In: *Journal of New Music Research* 29.4 (2000), pp. 259–273.

[9] F. Charpentier and M. Stella. "Diphone synthesis using an overlap-add technique for speech waveforms concatenation". In: *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'86)*. Vol. 4. IEEE. Tokyo, Japan, Apr. 1986, pp. 2015–2018.

[10] Chris Cutler. "Technology, politics and contemporary music: necessity and choice in musical forms". In: *Popular Music* 4 (Jan. 1984), pp. 279–300.

[11] Olivier Darrigol. "The acoustic origins of harmonic analysis". In: *Archive for History of Exact Sciences* 61.4 (July 2007), pp. 343–424.

[12] M. E. P. Davies and S. Böck. "Evaluating the evaluation measures for beat tracking". In: *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014)*. ISMIR. Taipei, Taiwan, Oct. 2014, pp. 637–642.

[13] M. E. P. Davies, N. Degara, and M. D. Plumbley. *Evaluation Methods for Musical Audio Beat Tracking Algorithms*. Technical Report C4DM-TR-09-06. London, UK: Queen Mary University, Centre for Digital Music, Oct. 2009.

[14] Matthew Davies and Adam Stark. *Beat Tracking Evaluation Toolbox*. 2012. URL: `http://code.soundsoftware.ac.uk/projects/beat-evaluation/`.

[15] J. Stephen Downie, Joe Futrelle, and David K. Tcheng. "The International Music Information Retrieval Systems Evaluation Laboratory: Governance, access and security". In: *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR 2004)*. ISMIR. Barcelona, Spain, Oct. 2004.

[16] J. Stephen Downie et al. "The 2005 music information retrieval evaluation exchange (MIREX 2005): Preliminary overview". In: *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*. ISMIR. London, UK, Sept. 2005, pp. 320–323.

[17] Jonathan Driedger, Meinard Müller, and Sebastian Ewert. "Improving time-scale modification of music signals using harmonic-percussive separation". In: *IEEE Signal Processing Letters* 21.1 (Jan. 2014), pp. 105–109.

[18] Chris Duxbury, Mike Davies, and Mark Sandler. "Separation of transient information in musical audio using multiresolution analysis techniques". In: *Proceedings of the 4th Conference on Digital Audio Effects (DAFx-01)*. COST-G6. Limerick, Ireland, Dec. 2001.

[19] Chris Duxbury, Mike Davies, and Mark B. Sandler. "Improved time-scaling of musical audio using phase locking at transients". In: *112th Audio Engineering Society Convention*. Convention Paper 5530. Audio Engineering Society. 2002.

[20] Daniel P. W. Ellis. "Beat tracking by dynamic programming". In: *Journal of New Music Research* 36.1 (July 2007), pp. 51–60.

[21] Yariv Ephraim and David Malah. "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.6 (Dec. 1984), pp. 1109–1121.

[22] C. Févotte. "Itakura-Saito nonnegative factorizations of the power spectrogram for music signal decomposition". In: *Machine Audition: Principles, Algorithms and Systems*. Ed. by Wenwu Wang. Hershey, USA: IGI Global Press, 2010. Chap. 11, pp. 266–296.

[23] D. Fitzgerald. "Harmonic/percussive separation using median filtering". In: *Proceedings of the 13th International Conference on Diital Audio Effects DAFx-10*. Graz, Austria, Sept. 2010.

[24] Patrick Flandrin. *Time-Frequency/Time-Scale Analysis*. Vol. 10. Wavelet Analysis and Its Applications. San Diego, USA: Academic Press, 1999.

[25] Richard Furse. *Linux audio developer's simple plugin api (LADSPA)*. Apr. 2000. URL: `http://www.ladspa.org`.

[26] Masataka Goto and Yoichi Muraoka. "Issues in evaluating beat tracking systems". In: *Working Notes of the IJCAI-97 Workshop on Issues in AI and Music-Evaluation and Assessment*. Nagoya, Japan, Aug. 1997, pp. 9–16.

[27] Masataka Goto and Yoichi Muraoka. "Real-time beat tracking for drumless audio signals: Chord change detection for musical decisions". In: *Speech Communication* 27.3 (Apr. 1999), pp. 311–335.

[28] Thomas Grandke. "Interpolation algorithms for discrete Fourier transforms of weighted signals". In: *IEEE Transactions on Instrumentation and Measurement* 32.2 (June 1983), pp. 350–355.

[29] Karlheinz Gröchenig. *Foundations of Time-Frequency Analysis*. Applied and Numerical Harmonic Analysis. New York, USA: Springer, 2001.

[30] Stephen Hainsworth and Malcolm Macleod. "On sinusoidal parameter estimation". In: *Proceedings of the 6th International Conference on Digital Audio Effects (DAFx-03)*. London, UK, Sept. 2003.

[31] Monson H. Hayes. *Statistical Digital Signal Processing and Modeling*. Hoboken, USA: John Wiley & Sons, 1996.

[32] Marko Helén and Tuomas Virtanen. "Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine". In: *Proceedings of the 13th European Signal Processing Conference (EUSIPCO 2005)*. EURASIP. Antalya, Turkey, Sept. 2005, pp. 1091–1094.

[33] Richard Heusdens, Renat Vafin, and W. Bastiaan Kleijn. "Sinusoidal modeling using psychoacoustic-adaptive matching pursuits". In: *IEEE Signal Processing Letters* 9.8 (Aug. 2002), pp. 262–265.

[34] Richard Heusdens and Steven Van De Par. "Rate-distortion optimal sinusoidal modeling of audio and speech using psychoacoustical matching pursuits". In: *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'02)*. Vol. 2. IEEE. Orlando, USA, May 2002, pp. 1809–1812.

[35] C. Hsu and J. Jang. "Singing pitch extraction at MIREX 2010". In: *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*. ISMIR. Utrecht, Netherlands, Aug. 2010.

[36] Ignacio Irigaray and Luiz W. P. Biscainho. "Transient and steady-state component extration using nonlinear filtering". In: *Anales del I Congreso Internacional de Ciencia y Tecnología Musical (CICTeM 2013)*. Buenos Aires, Argentina, Sept. 2013. URL: `http://iie.fing.edu.uy/publicaciones/2013/IB13`.

[37] Mark Kahrs and Karlheinz Brandenburg, eds. *Applications of Digital Signal Processing to Audio and Acoustics*. The Kluwer International Series in Engineering and Computer Science. New York, USA: Kluwer, 2002.

[38] T. Kasparis and J. Lane. "Suppression of impulsive disturbances from audio signals". In: *Electronics Letters* 29.22 (Oct. 1993), pp. 1926–1927.

[39] Takis Kasparis and John Lane. "Adaptive scratch noise filtering". In: *IEEE Transactions on Consumer Electronics* 39.4 (Nov. 1993), pp. 917–922.

[40] Florian Keiler, Sylvain Marchand, et al. "Survey on extraction of sinusoids in stationary sounds". In: *Proceedings of the 5th International Conference on Digital Audio Effects (DAFx-02)*. Hamburg, Germany, Sept. 2002, pp. 51–58.

[41] Seong Rag Kim and Adam Efron. "Adaptive robust impulse noise filtering". In: *IEEE Transactions on Signal Processing* 43.8 (Aug. 1995), pp. 1855–1866.

[42] Maria Klatte, Thomas Lachmann, and Markus Meis. "Effects of noise and reverberation on speech perception and listening comprehension of children and adults in a classroom-like setting". In: *Noise and Health* 12.49 (Oct. 2010), p. 270.

[43] N. Laurenti, G. De Poli, and D. Montagner. “A nonlinear method for stochastic spectrum estimation in the modeling of musical sounds”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.2 (Feb. 2007), pp. 531–541.

[44] Katia Lebart, Jean-Marc Boucher, and P. N. Denbigh. “A new method based on spectral subtraction for speech dereverberation”. In: *Acta Acustica united with Acustica* 87.3 (May 2001), pp. 359–366.

[45] Daniel D Lee and H Sebastian Seung. “Learning the parts of objects by non-negative matrix factorization”. In: *Nature* 401.6755 (Oct. 1999), pp. 788–791.

[46] Mark Lewisohn. *The Beatles Recording Sessions*. New York, USA: Harmony Books, 1988.

[47] Antoine Liutkus et al. “Kernel additive models for source separation”. In: *IEEE Transactions on Signal Processing* 62.16 (Aug. 2014), pp. 4298–4310.

[48] Katariina Mahkonen et al. “Music dereverberation by spectral linear predictiomn in live recordings”. In: *Proceedings of the 16th International Conference on Digital Audio Effects (DAFx-13)*. Maynooth, Ireland, Sept. 2013.

[49] Stéphane G Mallat and Zhifeng Zhang. “Matching pursuits with time-frequency dictionaries”. In: *IEEE Transactions on Signal Processing* 41.12 (Dec. 1993), pp. 3397–3415.

[50] Sylvain Marchand. “Improving spectral analysis precision with an enhanced phase vocoder using signal derivatives”. In: *Proceedings of the 1st Workshop on Digital Audio Effects (DAFx-98)*. COST-G6. Barcelona, Spain, Nov. 1998, pp. 114–118.

[51] Paul Masri and Andrew Bateman. “Improved modelling of attack transients in music analysis-resynthesis”. In: *Proceedings of the International Computer Music Conference (ICMC’96)*. ICMA. Hong Kong, China, Aug. 1996, pp. 100–103.

[52] R. McAulay and T. Quatieri. “Speech analysis/synthesis based on a sinusoidal representation”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 34.4 (Aug. 1986), pp. 744–754.

[53] Marin Mersenne. *Harmonie Universelle: Contenant la Théorie et la Pratique de la Musique*. Vol. 2. Paris, France: Éditions du Centre National de la Recherche Scientifique, 1637.

[54] Eric Moulines and Jean Laroche. “Non-parametric techniques for pitch-scale and time-scale modification of speech”. In: *Speech Communication* 16.2 (Feb. 1995), pp. 175–205.

[55] Musiktechnik. *SPL Transient Designer*. Sept. 2014. URL: http://spl.info/de/produkte/analog-coder-plug-ins/transient-designer/video.html.

[56] Patrick A. Naylor et al. "Models, measurement and evaluation". In: *Speech Dereverberation*. Ed. by Patrick A. Naylor and Nikolay D. Gaubitch. Signals and Communication Technology. London, UK: Springer, 2010. Chap. 2, pp. 21–56.

[57] Brett Ninness and Soren J. Henriksen. "Time-scale modification of speech signals". In: *IEEE Transactions on Signal Processing* 56.4 (Apr. 2008), pp. 1479–1488.

[58] Takuma Okamoto, Yukio Iwaya, and Yôiti Suzuki. "Wide-band dereverberation method based on multichannel linear prediction using prewhitening filter". In: *Applied Acoustics* 73.1 (Jan. 2012), pp. 50–55.

[59] Nobutaka Ono et al. "A real-time equalizer of harmonic and percussive components in music signals." In: *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR 2008)*. ISMIR. Philadelhia, USA, Sept. 2008, pp. 139–144.

[60] N. Ono et al. "Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram". In: *Proceedings of the 16th European Signal Processing Conference (EUSIPCO 2008)*. EURASIP. Lausanne, Switzerland, Aug. 2008.

[61] Graham E. Poliner et al. "Melody transcription from music audio: Approaches and evaluation". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.4 (May 2007), pp. 1247–1256.

[62] Michael R. Portnoff. "Time-scale modification of speech based on short-time Fourier analysis". In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 29.3 (June 1981), pp. 374–390.

[63] William K. Pratt. *Digital Image Processing*. 4th ed. Hoboken, USA: John Wiley & Sons, Inc., 2007.

[64] Miller Puckette. "Pure Data: another integrated computer music environment". In: *Proceedings of the International Computer Music Conference (ICMC'96)*. ICMA. Hong Kong, China, Aug. 1996, pp. 37–41.

[65] Zbigniew W. Rás and Alicja Wieczorkowska, eds. *Advances in Music Information Retrieval*. Vol. 274. Studies in Computational Intelligence. Berlin, Germany: Springer, 2010.

[66] Axel Röbel. "Transient detection and preservation in the phase vocoder". In: *Proceedings of the International Computer Music Conference (ICMC'03)*. ICMA. Singapore, 2003, pp. 247–250.

[67] Salim Roucos and Alexander Wilgus. “High quality time-scale modification for speech”. In: *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP’85)*. Vol. 10. IEEE. Tampa, USA, 1985, pp. 493–496.

[68] Luigi Russolo et al. *The Art of Noise: Futurist Manifesto, 1913*. New York, USA: Something Else Press, 1967.

[69] Justin Salamon and Emilia Gómez. “Melody extraction from polyphonic music signals using pitch contour characteristics”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.6 (Aug. 2012), pp. 1759–1770.

[70] Joseph Sauveur. *Principes d’Acoustique et de Musique: ou, Système Général des Intervalles des Sons*. Geneva, Switzerland: Éditions Minkoff, 1701.

[71] Kraft Sebastian. *mushraJS*. Mar. 2013. URL: `https://github.com/seebk/mushraJS`.

[72] X. Serra and J. Smith III. “Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition”. In: *Computer Music Journal* 4.14 (Winter 1990), pp. 12–24.

[73] Xavier Serra. *A System for Sound Analysis/Transformation/Synthesis Based on a Deterministic Plus Stochastic Decomposition*. Technical Report STAN-M-58. Stanford, USA: Stanford University, Center for Computer Research in Music and Acoustics, Oct. 1989.

[74] Xavier Serra. “Musical sound modeling with sinusoids plus noise”. In: *Musical Signal Processing*. Ed. by Curtis Roads et al. Studies on New Music Research. New York, USA: Routledge, 1997, pp. 91–122.

[75] Kai Siedenburg and Monika Dörfler. “Structured sparsity for audio signals”. In: *Proceedings of the 14th International Conference on Digital Audio Effects (DAFx-11)*. Paris, France, Sept. 2011.

[76] David Slepian and Henry O. Pollak. “Prolate spheroidal wave functions, Fourier analysis and uncertainty—I”. In: *The Bell System Technical Journal* 40.1 (Jan. 1961), pp. 43–63.

[77] Hideyuki Tachibana et al. “Melody line estimation in homophonic music audio signals based on temporal-variability of melodic source”. In: *Proceedings of the 35th International Conference on Acoustics, Speech, and Signal Processing (ICASSP’10)*. IEEE. Dallas, USA, Mar. 2010, pp. 425–428.

[78] George Tanev and Adrijan Božinovski. "Virtual Studio Technology inside Music Production". In: *ICT Innovations 2013: ICT Innovations and Education*. Ed. by Vladimir Trajkovik and Anastas Mishev. Vol. 231. Advances in Intelligent Systems and Computing. Cham, Switzerland: Springer, 2014, pp. 231–241.

[79] Schaack Audio Techonolgies. *Transient Shaper*. Sept. 2014. URL: `http://www.schaack-audio.com/transientshaper.html`.

[80] Alexandros Tsilfidis and John Mourjopoulos. "Blind single-channel suppression of late reverberation based on perceptual reverberation modeling". In: *The Journal of the Acoustical Society of America* 129.3 (Mar. 2011), pp. 1439–1451.

[81] Christian Uhle, Christian Dittmar, and Thomas Sporer. "Extraction of drum tracks from polyphonic music using Independent Subspace Analysis". In: *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*. Nara, Japan, Apr. 2003, pp. 843–848.

[82] S. V. Vaseghi and P. J. W. Rayner. "Detection and suppression of impulsive noise in speech communication systems". In: *IEE Proceedings I - Communications, Speech and Vision)* 137.1 (Feb. 1990), pp. 38–46.

[83] T. Verma, S. Levine, and T. Meng. "Transient modeling synthesis: a flexible analysis/synthesis tool for transient signals". In: *Proceedings of the International Computer Music Conference (ICMC'97)*. ICMA. Thessaloniki, Greece, Sept. 1997, pp. 164–167.

[84] Emmanuel Vincent et al. *Blind Audio Source Separation*. Technical Report C4DM-TR-05-01. London, UK: Queen Mary University, Centre for Digital Music, Nov. 2005.

[85] Tuomas Virtanen. "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.3 (Mar. 2007), pp. 1066–1074.

[86] Tuomas Virtanen. "Unsupervised learning methods for source separation in monaural music signals". In: *Signal Processing Methods for Music Transcription*. Ed. by Anssi Klapuri and Manuel Davy. New York, USA: Springer, 2006. Chap. 9, pp. 267–296.

[87] Jing Wang et al. "An adaptive window switching method for ITU-T G.719 transient coding in TDA domain". In: *Proceedings of the 3rd International Conference on Wireless, Mobile and Multimedia Networks (ICWMNN 2010)*. IET. Beijing, China, Sept. 2010, pp. 298–301.

[88] Thomas Wilmering, Mathieu Barthet, and Mark B. Sandler. "Dereverberation of musical instrument recordings for improved note onset detection and instrument recognition". In: *131st Audio Engineering Society Convention*. Convention Paper 8508. Audio Engineering Society. New York, USA, Oct. 2011.

[89] Thomas Wilmering, György Fazekas, and Mark Sandler. "The effects of reverberation on onset detection tasks". In: *128th Audio Engineering Society Convention*. Convention Paper 8114. Audio Engineering Society. London, UK, May 2010.

[90] Trevor Wishart. "The Composition of "Vox-5"". In: *Computer Music Journal* 12.4 (Winter 1988), pp. 21–27.

[91] John F. Woodruff, Bryan Pardo, and Roger B. Dannenberg. "Remixing stereo music with score-informed source separation". In: *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*. ISMIR. Victoria, Canada, Oct. 2006, pp. 314–319.

[92] Naoki Yasuraoka et al. "Music dereverberation using harmonic structure source model and Wiener filter". In: *Proceedings of the 35th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'10)*. IEEE. Dallas, USA, Mar. 2010, pp. 53–56.

[93] Tao Zhang, Wei Wang, and Jialin He. "On the pre-echo control method in transient signal coding of AVS audio". In: *Proceedings of the 2008 International Conference on Audio, Language and Image Processing (ICALIP 2008)*. IEEE. Shanghai, China, July 2008, pp. 242–246.

[94] Slawomir Zielinski and Francis Rumsey. "On some biases encountered in modern audio quality listening tests-A Review". In: *Journal of the Audio Engineering Society* 56 (June 2008), pp. 427–451.

Bibliography

List of Tables

4.1 Result of subjective test. The subscripts (t) and (ss) indicates transient and steady-state respectively. 37
4.2 Average beat-tracking performance measure. 39
4.3 Average pitch-tracking performance measure. 42

List of Tables

List of Figures

2.1 Time and frequency representation of an audio signal. A male voice singing "so". Starts with the unvoiced consonant [s] and ends with the voiced vowel [o]. 7
2.2 Representation of three Gaussian sinusoidal atoms: time, frequency and joint time frequency representations. 9
2.3 This diagram summarises the Short-Time Fourier Transform calculation. 10
2.4 Tiling of the time-frequency plane: STFT (left and center) and CQT (right). 12
2.5 CQT and STFT spectrograms; frequency axis in logarithmic scale. 13
2.6 Analysis/Synthesis block diagrams of Sinusoidal Modeling (adapted from figure of original article [52]). 14
2.7 Assignment of peaks to tracks is decided by proximity (adapted from [74]). Tracks may be discontinued if not find adequate peaks for several frames; and new tracks may be created if a series of coherent peaks is find for several frames. 14
2.8 Sinusoidal modeling resynthesis. 15
2.9 Spectrogram with sinusoidal tracks superimposed for word "so". One can observe that the voiced part is well modeled by the sinusoidal tracks. 16
2.10 Left: Time domain impulse/pulse. Right: Spectra 17
2.11 Atoms from two Gabor Dictionaries with different time-frequency behavior. 18
2.12 Matching Pursuit with different Gabor dictionaries: decomposition of a glockenspiel sound. 19
2.13 Matrix decomposition diagram. 19

3.1 Transient and steady-state components of a glockenspiel sound. . . 22
3.2 Spectrogram of a glockenspiel (N=4096,Hop=256). 23
3.3 Diagram of the entire process. 24
3.4 Glockenspiel spectrogram (center), time evolution for a fixed FFT bin (bottom) and spectral content of a frame (right). 26
3.5 Steps of the Stochastic Spectral Estimation applied along the time axis. Based on [43]. 27

3.6 Left: Spectrogram of a excerpt from a popular music song. Middle: Spectrogram with transient components. Right: Spectrogram with steady-state components. 27
3.7 Diagram of the iterative process. 28
3.8 Steady-state, transient component and residual extraction for different number of iterations. 29
3.9 Time-frequency kernel. 30
3.10 left) Spectrogram of a superposition of a periodic signal with vibrato to two clicks; right) Comparison between the original and the modified decompositions. 30
3.11 Diagram of the sub-band processing schema. 30
3.12 Filters response for the Filter-bank analysis. 31
3.13 Transient and steady-state separation spectrograms for each sub-band. Band 1 is 0 to 1.5 kHz, band 2 is 1.5 to 4.4 kHz, band 3 is 4.4 to 10.3 kHz and band 4 is 10.3 to 22 kHz. 32
3.14 Wiener filter configuration for transient and steady-state component separation. 33

4.1 One of the performance measurements for beat tracking (Information Gain); comparison for all elements in the data set. 40
4.2 Beat-tracking outputs. 40
4.3 Steps of the pitch-tracking MELODIA; involved signals. 43
4.4 Transient/steady-state separation for manual editing. 44
4.5 Remixing application. 45
4.6 Spectrogram of a vinyl record. 46
4.7 Noise reduction. 47
4.8 De-reverberation of a drum sound. 48
4.9 Different time-pitch modifications. 49
4.10 Transient smear in an octave-down pitch-shift. 50
4.11 Not smeared transient in a octave-down pitch-shift. 50

B.1 Subjective listening test web interface. 61

Esta es la última página.
Compilado el Monday 9th February, 2015.
http://iie.fing.edu.uy/