

From Sparse Models to Timbre Learning: New Methods for Musical Source Separation

vorgelegt von

Juan José Burred
aus Valencia, Spanien

Von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr.-Ing. Reinhold Orglmeister
Berichter:	Prof. Dr.-Ing. Thomas Sikora
Berichter:	Prof. Dr. Gaël Richard

Tag der wissenschaftlichen Aussprache: 11.9.2008

Berlin 2009
D 83

Abstract

The goal of source separation is to detect and extract the individual signals present in a mixture. Its application to sound signals and, in particular, to music signals, is of interest for content analysis and retrieval applications arising in the context of online music services. Other applications include unmixing and remixing for post-production, restoration of old recordings, object-based audio compression and upmixing to multichannel setups.

This work addresses the task of source separation from monaural and stereophonic linear musical mixtures. In both cases, the problem is *underdetermined*, meaning that there are more sources to separate than channels in the observed mixture. This requires taking strong statistical assumptions and/or learning a priori information about the sources in order for a solution to be feasible. On the other hand, constraining the analysis to instrumental music signals allows exploiting specific cues such as spectral and temporal smoothness, note-based segmentation and timbre similarity for the detection and extraction of sound events.

The statistical assumptions and, if present, the a priori information, are both captured by a given *source model* that can greatly vary in complexity and extent of application. The approach used here is to consider source models of increasing levels of complexity, and to study both their implications on the separation algorithm, and the type of mixtures they are able to handle.

The starting point is sparsity-based separation, which makes the general assumption that the sources can be represented in a transformed domain with few high-energy coefficients. It will be shown that sparsity, and consequently separation, can both be improved by using nonuniform-resolution time–frequency representations. To that end, several types of frequency-warped filter banks will be used as signal front-ends in conjunction with an unsupervised separation approach aimed at stereo signals.

As a next step, more sophisticated models based on sinusoidal modeling and statistical training will be considered in order to improve separation and to allow the consideration of the maximally underdetermined problem: separation from single-channel signals. An emphasis is given in this work to a detailed but compact approach to train models of the timbre of musical instruments. An important characteristic of the approach is that it aims at a close description of the temporal evolution of the spectral envelope. The proposed method uses a formant-preserving, dimension-reduced representation of the spectral envelope based on spectral interpolation and Principal Component Analysis. It then describes the timbre of a given instrument as a Gaussian Process that can be interpreted either as a *prototype curve* in a timbral space or as a time–frequency template in the spectral domain. Such templates will be used for the grouping and separation of sinusoidal tracks from the mixture.

A monaural separation method based on sinusoidal modeling and on the mentioned timbre modeling approach will be presented. It exploits common-fate and good-continuation cues to extract groups of sinusoidal tracks corresponding to the individual notes. Each group is compared to each one of the timbre templates on the database using a specially-designed measure of timbre similarity, followed by a

Maximum Likelihood decision. Subsequently, overlapping and missing parts of the sinusoidal tracks are retrieved by interpolating the selected timbre template. The method is later extended to stereo mixtures by using a preliminary spatial-based blind separation stage, followed by a set of refinements performed by the above sinusoidal modeling and timbre matching methods and aiming at reducing interferences with the undesired sources.

A notable characteristic of the proposed separation methods is that they do not assume harmonicity, and are thus not based on a previous multipitch estimation stage, nor on the input of detailed pitch-related information. Instead, grouping and separation relies solely on the dynamic behavior of the amplitudes of the partials. This also allows separating highly inharmonic sounds and extracting chords played by a single instrument as individual entities.

The fact that the presented approaches are supervised and based on classification and similarity allows using them (or parts thereof) for other content analysis applications. In particular the use of the timbre models, and the timbre matching stages of the separation systems will be evaluated in the tasks of musical instrument classification and detection of instruments in polyphonic mixtures.

Kurzfassung

Das Ziel der Quellentrennung ist die Erkennung und Extraktion der einzelnen Signale, die in einer Mischung vorhanden sind. Ihre Anwendung auf Audiosignale und im Besonderen auf Musiksignale ist von großem praktischen Interesse im Rahmen der inhaltsbasierten Analyse für neue Online-Musikdienste und Multimediaanwendungen. Quellentrennung findet auch Einsatz in Studio-Nachbearbeitung, Wiederherstellung alter Aufnahmen, objektbasierter Audiocodierung und beim Erstellen neuer Mischungen für mehrkanalige Systeme.

Die vorliegende Dissertation befasst sich mit der Aufgabe, Quellen aus linearen Mono- und Stereomusikmischungen zu extrahieren. In beiden Fällen ist die Aufgabenstellung *unterbestimmt*, d.h., es gibt mehr Quellen zu trennen als Kanäle in der Mischung vorhanden sind. Dies verlangt starke statistische Annahmen, bzw. das A-priori-Erlernen von Information über die Quellen. Andererseits erlaubt die Anwendung auf Musiksignale, spezifische Eigenschaften auszunutzen, wie etwa spektrale und zeitliche Glattheit, notenbasierte Segmentierung und Ähnlichkeit der Klangfarbe, um die einzelnen Klangereignisse zu erkennen und zu trennen.

Sowohl die statistischen Annahmen als auch das eventuelle Vorwissen werden von einem bestimmten *Quellenmodell* erfasst. Ein solches Modell kann stark in Komplexität und Anwendbarkeit variieren. Der verwendete methodische Ansatz bestand daraus, verschiedene Quellenmodelle wachsender Komplexität zu betrachten und ihre jeweiligen Auswirkungen auf die Trennungsalgorithmen und auf den Typ von Mischungen, die sie verarbeiten können, zu studieren.

Der Ausgangspunkt ist die Trennung basierend auf dünnbesetzten (*sparse*) Signalen, in welchem Fall angenommen wird, dass die Quellen in einem bestimmten transformierten Bereich mit wenigen energiereichen Koeffizienten dargestellt werden können. Es wird gezeigt, dass *sparsity*, und folglich Trennung, durch die Verwendung von Zeit-Frequenz Darstellungen nicht-linearer Auflösung verbessert werden. Zu diesem Zweck werden verschiedene Arten von frequenzverzerrten Filterbänken als Front-End im Zusammenhang mit einer unüberwachten Stereo-Trennungsmethode ausgewertet.

Als nächster Schritt werden komplexere Modelle, basierend auf sinusoidaler Modellierung und statistischem Lernen, in Betracht gezogen. Sie erlauben, die maximal unterbestimmte Situation zu behandeln, nämlich die Trennung aus einer einkanaligen (monophonen) Mischung. Ein besonderer Schwerpunkt wird auf das Lernen eines detaillierten, wenngleich kompakten Modells der Klangfarbe von Musikinstrumenten gelegt. Die vorgeschlagene Methode benutzt eine formantenerhaltende, dimensionsreduzierte Darstellung der spektralen Hüllkurve, die auf spektraler Interpolation und auf *Hauptkomponentenanalyse* beruht. Eine wichtige Eigenschaft des Ansatzes ist die detaillierte Beschreibung des zeitlichen Verlaufs der Hüllkurve. Das resultierende Modell beschreibt die Klangfarbe eines Instrumentes entweder in Form einer *Prototypkurve* im Klangfarbenraum oder als eine *Zeit-Frequenz-Schablone* im spektralen Bereich. Solche Schablonen werden für die Gruppierung und Trennung der im Spektrum vorhandenen Partialtöne verwendet.

Im Anschluss wird ein Ansatz für monophone Trennung, die auf solchen Klangfarbenmodellen basiert, vorgestellt. Er gruppiert die Partialtöne anhand von ge-

meinsamen dynamischen Eigenschaften. Jede Gruppe wird mit den erlernten Zeit-Frequenz-Schablonen verglichen, unter Benutzung eines speziell entworfenen Maßes von Klangfarbenähnlichkeit, gefolgt von einer *Maximum-Likelihood*-Entscheidung. Die überlappenden und unvollständigen Anteile werden vom Modell mittels Interpolation gewonnen. Diese Methode wird anschließend für Stereomischungen erweitert. Dafür wird ein Modul für blinde Stereoquellentrennung als Vorverarbeitungsstufe eingesetzt, gefolgt von einer Reihe Verfeinerungen, die durch die erwähnten sinusoidalen Methoden realisiert werden.

Eine besondere Eigenschaft der vorgestellten Trennungsmethoden ist, dass keine Harmonizität angenommen wird. Die Trennung basiert also nicht auf einer vorhandenen Analyse der Grundfrequenzen in der Mischung und verlangt keine Eingabe von Information über die vorhandenen Tonhöhen. Stattdessen beruht die Gruppierung und Trennung der Partialtöne lediglich auf dem dynamischen Verhalten ihrer Amplituden. Dies erlaubt ebenfalls die Trennung disharmonischer Klänge und die einheitliche Extraktion von Akkorden.

Die Tatsache, dass die vorgeschlagenen Methoden überwacht sind und dass sie auf Klassifizierung und Ähnlichkeitsmessungen basieren, erlaubt ihre Verwendung für andere inhaltsbasierte Anwendungen. Somit werden die entwickelten Klangfarbenmodelle in monophonen und polyphonen Klassifizierungsaufgaben ausgewertet.

Resumen

El objetivo de la separación de fuentes es detectar y extraer las distintas señales presentes en una mezcla. Su aplicación a señales de audio y, en particular, a señales musicales, es de elevado interés para aplicaciones del análisis y la recuperación de datos basadas en el contenido, tales como las que han surgido recientemente a raíz de los nuevos servicios de distribución de música por Internet. Otras aplicaciones incluyen, por ejemplo, la separación y remezcla en postproducción, la restauración de grabaciones antiguas, la compresión de audio basada en objetos y la conversión automática a formatos multicanal.

El presente trabajo se centra en la separación de mezclas lineales monoaurales y estereofónicas. En ambos casos, el problema es de tipo *subdeterminado*, lo cual significa que hay más fuentes que canales en la mezcla observada. En este caso, para que la solución sea factible, es necesario asumir ciertas hipótesis estadísticas restrictivas, o bien llevar a cabo un aprendizaje basado en información disponible a priori. Por otro lado, el hecho de restringir el análisis al caso musical permite aprovechar elementos específicos, tales como la uniformidad espectral y temporal, la segmentación a nivel de nota y la similitud tímbrica, para la detección y separación de los eventos sonoros.

Las hipótesis estadísticas y, dado el caso, la información a priori, se ven reflejadas en un *modelo de fuente* cuya complejidad y campo de aplicación puede variar sustancialmente. El enfoque metodológico sobre el que se basa el presente trabajo consiste en ir considerando modelos de complejidad creciente, y en ir estudiando en cada caso las implicaciones sobre el algoritmo de separación y sobre el tipo de mezclas que son capaces de abordar.

El punto de partida es la separación basada en la premisa de escasez (*sparsity*), la cual supone que las fuentes pueden representarse mediante un número reducido de coeficientes no-nulos, al menos en un cierto dominio transformado. Se demuestra que la escasez, y por lo tanto la separación, pueden mejorarse mediante el uso de representaciones tiempo-frecuencia de resolución no uniforme. Para ello, se estudian varios tipos de bancos de filtros no uniformes en la fase de representación de la señal, combinándolos con un algoritmo de separación no supervisada destinado a mezclas estereofónicas.

A continuación se consideran modelos más sofisticados basados en modelos sinusoidales y aprendizaje estadístico, con el fin de mejorar la separación y permitir la toma en consideración de la situación más subdeterminada posible: la separación de mezclas monocanal. Este trabajo otorga especial atención al desarrollo de un modelo detallado pero compacto del timbre de instrumentos musicales, el cual puede ser usado como información a priori en el proceso de separación. El método propuesto usa una representación de baja dimensionalidad de la envolvente espectral que preserva los formantes y se basa en interpolación espectral y *Análisis de Componentes Principales*. Se hace especial énfasis en la descripción detallada de la evolución temporal de la envolvente espectral. De esta forma se obtiene una descripción del timbre de un determinado instrumento en forma de un Proceso Gaussiano que puede interpretarse como una *curva prototipo* en un espacio tímbrico, o bien como un patrón tiempo-frecuencia en el dominio espectral. Dichos patrones se usan

como guía al agrupar y separar las trayectorias sinusoidales presentes en la mezcla.

El primer método de separación supervisada propuesto está destinado a mezclas monocanal y se basa en modelos sinusoidales y en los mencionados modelos tímbricos, previamente entrenados. Analiza los indicios psicoacústicos de *destino común* y *buena continuación* para extraer parcialmente grupos de trayectorias sinusoidales correspondientes a notas individuales. Cada grupo de trayectorias es comparado con cada patrón tímbrico presente en la base de datos mediante el uso de una medida de similitud tímbrica diseñada a tal efecto, a lo cual sigue una decisión de máxima verosimilitud. Los fragmentos ausentes o solapados de las sinusoides se regeneran interpolando el patrón tímbrico seleccionado en el paso anterior. Este método es ampliado a continuación al caso estereofónico mediante la inclusión de una etapa previa de separación ciega basada en la distribución espacial de las fuentes, seguida de una serie de refinamientos llevados a cabo por los anteriores métodos sinusoidales y tímbricos, y destinados a reducir las interferencias con las fuentes no deseadas.

Cabe destacar que ninguno de los métodos propuestos presupone la armonicidad de las fuentes, y por lo tanto no se basan en una etapa previa de transcripción polifónica, ni necesitan información detallada sobre las alturas de las notas. El agrupamiento y separación están basados únicamente en el comportamiento dinámico de las amplitudes de los parciales. Esto implica que es posible separar sonidos altamente inarmónicos, o extraer un acorde tocado por un solo instrumento como una sola entidad.

El hecho de que los procedimientos propuestos sean supervisados y se basen en la clasificación y en medidas de similitud permite su uso en el contexto de otras aplicaciones basadas en el contenido. En concreto, los módulos de comparación y aprendizaje tímbrico serán evaluados en tareas de clasificación y detección de instrumentos musicales en muestras individuales o en mezclas polifónicas.

Acknowledgments

I first wish to express my deepest gratitude to my supervisor, Thomas Sikora, for his constant support and guidance, and for giving me the opportunity to perform this research work at the Communication Systems Group of the Technical University of Berlin. I would also like to thank Gaël Richard for reviewing the thesis and for his valuable advice.

I am extremely grateful to Axel Röbel and Xavier Rodet for hosting me at the Analysis/Synthesis team, IRCAM, during a very enriching 4-month research stay.

I performed the experiments for polyphonic instrument recognition of Sect. 4.8 in collaboration with Luís Gustavo Martins, whom I wish to thank for the fruitful discussions.

Thanks to Carmine Emanuele Cella, Kai Clüver, Martin Haller, Leigh Smith and Jan Weil for reviewing and discussing the manuscripts. Thanks to Jan-Mark Batke for providing the saxophone multitrack recordings used in several experiments.

I thank all my colleagues in Berlin and Paris for the great time working together. I thank all the inspiring people I have met at conferences and meetings throughout these years.

Finally, I am deeply indebted to my parents, Juan José and Pilar, and my brother, Luis Alberto, for their continued confidence and support. This thesis is dedicated to them.

Contents

Abstract	iii
Kurzfassung	v
Resumen	vii
Acknowledgments	ix
List of Abbreviations	xv
1 Introduction	1
1.1 Applications of audio source separation	3
1.2 Motivations and goals	6
1.3 Overview of the thesis	7
2 Audio source separation: overview and principles	9
2.1 Mixing models	13
2.1.1 Instantaneous mixing model	14
2.1.2 Delayed mixing model	15
2.1.3 Convolutional mixing model	15
2.1.4 Noisy mixture models	16
2.2 Stereo recording techniques	16
2.3 Basic signal models	22
2.3.1 Basis decompositions	23
2.3.2 Time–frequency decompositions	25
2.3.3 Sparse decompositions	27
2.3.4 Principal Component Analysis	32
2.4 Analogy between signal decomposition and source separation	36
2.5 Joint and staged source separation	37
2.6 Estimation of the mixing matrix	40
2.6.1 Independent Component Analysis	41
2.6.2 Clustering methods	44
2.6.3 Other methods	45
2.7 Estimation of the sources	45
2.7.1 Heuristic approaches	46
2.7.2 ℓ_1 and ℓ_2 minimization	46
2.7.3 Time–frequency masking	47
2.8 Computational Auditory Scene Analysis	48
2.9 Summary	49
3 Frequency-warped blind stereo separation	51
3.1 Frequency-warped representations	53
3.1.1 Invertibility of the representations	61
3.2 Evaluation of source sparsity	62

3.2.1	Sparsity properties of speech and music signals	62
3.2.2	Sparsity properties of frequency-warped signals	65
3.3	Disjointness and W-Disjoint Orthogonality	65
3.3.1	Disjointness properties of speech and music mixtures	68
3.3.2	Disjointness properties of frequency-warped mixtures	70
3.4	Frequency-warped mixing matrix estimation	71
3.4.1	Kernel-based angular clustering	72
3.4.2	Evaluation with frequency-warped representations	74
3.5	Frequency-warped source estimation	75
3.5.1	Shortest path resynthesis	75
3.5.2	Measurement of separation quality	76
3.5.3	Evaluation with frequency-warped representations	78
3.6	Summary of conclusions	81
4	Source modeling for musical instruments	83
4.1	The spectral envelope	84
4.2	Sinusoidal modeling	89
4.3	Modeling timbre: previous work	93
4.4	Developed model	95
4.5	Representation stage	97
4.5.1	Basis decomposition of spectral envelopes	97
4.5.2	Dealing with variable frequency supports	101
4.5.3	Evaluation of the representation stage	103
4.6	Prototyping stage	110
4.7	Application to musical instrument classification	118
4.7.1	Comparison with MFCC	119
4.8	Application to polyphonic instrument recognition	121
4.9	Conclusions	124
5	Monaural separation based on timbre models	127
5.1	Monaural music separation based on advanced source models	127
5.1.1	Unsupervised methods based on adaptive basis decomposition	128
5.1.2	Unsupervised methods based on sinusoidal modeling	129
5.1.3	Supervised methods	130
5.2	Proposed system	131
5.2.1	Experimental setup	133
5.2.2	Onset detection	134
5.2.3	Track grouping and labeling	136
5.2.4	Timbre matching	138
5.2.5	Track retrieval	143
5.3	Evaluation of separation performance	145
5.3.1	Experiments with individual notes	147
5.3.2	Experiments with note sequences	147
5.3.3	Experiments with chords and clusters	148
5.3.4	Experiments with inharmonic sounds	151

5.4	Conclusions	152
6	Extension to stereo mixtures	155
6.1	Hybrid source separation systems	155
6.2	Stereo separation based on track retrieval	156
6.3	Stereo separation based on sinusoidal subtraction	158
6.3.1	Extraneous track detection	160
6.4	Evaluation of classification performance	163
6.5	Evaluation of separation performance	165
6.5.1	Stereo version of monaural experiments	166
6.5.2	Experiments with simultaneous notes	167
6.6	Conclusions	167
7	Conclusions and outlook	169
7.1	Summary of results and contributions	169
7.2	Outlook	172
A	Related publications	177
	List of Figures	179
	List of Tables	183
	Bibliography	185
	Index	199

List of Abbreviations

Abbreviation	Meaning
ACA	Audio Content Analysis
ADSR	Attack-Decay-Sustain-Release
ASA	Auditory Scene Analysis
BSS	Blind Source Separation
CASA	Computational Auditory Scene Analysis
COLA	Constant Overlap-Add
CQ	Constant Q (quality factor)
CQT	Constant Q Transform
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DR	Detection Rate
DWT	Discrete Wavelet Transform
EI	Envelope Interpolation
ERB	Equal Rectangular Bandwidth
FFT	Fast Fourier Transform
GMM	Gaussian Mixture Model
GP	Gaussian Process
ICA	Independent Component Analysis
i.i.d.	Independent, identically distributed
IID	Inter-channel Intensity Difference
IPD	Inter-channel Phase Difference
ISA	Independent Subspace Analysis
LPC	Linear Prediction Coding
MAP	Maximum A Posteriori
MCA	Music Content Analysis
MIDI	Musical Instrument Digital Interface
MIR	Music Information Retrieval
MFCC	Mel Frequency Cepstral Coefficients
ML	Maximum Likelihood
MSE	Mean Square Error
NMF	Non-negative Matrix Factorization
NSC	Non-negative Sparse Coding
OLA	Overlap-Add
PCA	Principal Component Analysis
PI	Partial Indexing
PRC	Precision
PSR	Preserved Signal Ratio
pdf	Probability Density Function
RCL	Recall
RSE	Relative Spectral Error
SAC	Structured Audio Coding

SAR	Source to Artifacts Ratio
SBSS	Semi-Blind Source Separation
SDR	Source to Distortion Ratio
SER	Signal to Error Ratio
SIR	Source to Interference Ratio
SSER	Spectral Signal to Error Ratio
STFT	Short Time Fourier Transform
WDO	W-Disjoint Orthogonality

1

Introduction

Since the introduction of digital audio technologies more than 35 years ago, computers and signal processing units have been capable of storing, modifying, transmitting and synthesizing sound signals. The later development and refinement of fields such as machine learning, data mining and pattern recognition, together with the increase in computing power, gave rise to a whole new set of audio applications that were able to automatically interpret the content of the sound signals being conveyed, and to handle them accordingly. In a very broad sense, such new *content-based* applications allowed not only the extraction of global semantic information from the signals, but also the detection, analysis and further processing of the individual sound entities constituting an acoustic complex.

Source separation is the task of extracting the individual signals from an observed mixture by computational means. This work focuses on the separation of audio signals, and more specifically of music signals, but source separation is useful applied to many other types of signals, such as image, video, neural, medical, financial or radio signals. Source separation is a challenging problem that began to be addressed in the mid 1980's. It was first formulated within a statistical framework by Héroult and Jutten [72]. With the introduction of *Independent Component Analysis* (ICA) and related techniques in the early 1990's [46], its theoretical study and practical deployment rapidly accelerated.

In the specific case of sound signals, several psychoacoustical studies, and most notably the 1990 work *Auditory Scene Analysis* by Bregman [25], provided the basis for the computational implementation of algorithms mimicking the sound segregation capabilities of the human hearing system. These developments opened two alternative approaches to acoustic separation: biologically-inspired and statistical/mathematical approaches. As it will be seen throughout the present work, more recent developments are based on a combination of both. Another important milestone that helped sound separation was the development of advanced spectral models such as sinusoidal modeling, first presented in 1984 by McAulay and Quatieri [107].

The ability of the human auditory system and associated cognitive processes to concentrate the attention on a specific sound source from within a mixture of sounds has been coined the *cocktail party effect*. First described in 1953 by Cherry [43], the cocktail party effect refers to the fact that a listener can easily follow a conversation with a single talker in a highly noisy environment, such as a party, with many other

interfering sound sources like other talkers, background music, or noises. This is even the case when the energy of the interfering sources, as captured by microphones at the listener's position, is close to the energy of the source on which the attention is focused. A perhaps more appropriate allegory, when applied to computational source separation, refers to the legend that Japanese prince Shōtoku could listen and understand simultaneously the petitions by ten people [118]. Indeed, some systems performing source separation have been called *Prince Shōtoku Computers*, since they usually do not concentrate on a single source, but output a set of separated channels. Note that both allegoric references imply an extra step of semantic understanding of the sources, beyond mere acoustical isolation.

The difficulty of a source separation problem is mainly determined by two factors: the nature of the mixtures and the amount of information about the sources, or about the mixture, available a priori. A detailed discussion of these criteria and their implications will be presented in the next chapter. Here, only the most important concepts and terms are introduced. Source separation is said to be *blind* if there is little or no knowledge available before the observation of the mixture. The term *Blind Source Separation* (BSS) has become the standard label to denote such kind of statistical methods. Strictly speaking, however, there exists no real fully-blind systems, since at least some general probabilistic assumptions must be taken, most often related to *statistical independence* and *sparsity*. It is therefore more appropriate to state that the *blindness* refers to the complexity of the exploited signal models.

There is no generalized consistent assignment between methodological labels and degree of knowledge. In the present work, the following conventions will be used. BSS will refer to problems in which relatively simple statistical assumptions about the sources are made. This includes ICA and sparsity-based methods such as norm-minimization and time–frequency masking methods. *Semi-blind Source Separation* (SBSS) will be applied to methods based on more advanced models of the sources such as sinusoidal models or adaptive basis decompositions. A subgroup of SBSS methods are *supervised separation* methods, in which a set of source models are learnt beforehand from a database of sound examples. Finally, non-blind source separation will refer to systems that need as input, besides the mixture, detailed, high-level information about the sources, such as the musical score or a MIDI sequence.

Another crucial factor is the proportion between number of mixture channels and number of original sources. Separation is easier if the observed mixture has more channels, or the same number of channels, than there are sources to separate. These cases are named, respectively, *over-determined* and *even-determined* (or *determined*) source separation. For this reason, the first practical approaches that appeared on the literature were related to applications involving arrays of microphones, sensors or antennas. Mixtures with less channels than sources are said to be *underdetermined* and pose additional difficulties that must often be addressed by means of stronger assumptions or a larger amount of information. Also, the separation difficulty will depend on the level of reverberation and noise contained in the mixture. As will be seen in detail in Chapter 2, each set of source and mixture characteristics has a corresponding mathematical formulation in the form of a *mixing model*.

1.1 Applications of audio source separation

As mentioned, the sound segregation capabilities of the auditory system have been an important motivation and driving force for research in source separation. It can be argued, however, that actually no real, full separation takes place in the inner ear, nor in the auditory cortex. In fact, we do not really hear separate instruments or voices; sound localization and segregation more appropriately refers to a selective weighting of sound entities in such a way that a differentiated semantical characterization is possible.

In this context, applications of sound source separation can be divided into two broad groups, which Vincent *et al.* [162] call *Audio Quality Oriented* (AQO) applications and *Significance Oriented* (SO) applications. AQO approaches aim at an actual full unmixing of the individual sources with the highest possible quality; in this case, the output signals are intended to be listened to. In contrast, the less demanding SO methods require a separation quality that is just high enough for the extraction of high-level, semantic information from the partially-separated sources. Obviously, separation methods capable of reaching AQO quality will be useful in an SO context as well.

A similar, albeit more general, paradigmatic typology is proposed by Scheirer [135]. He makes the distinction between an *understanding-without-separation* and a *separation-for-understanding* paradigm. In the former, it is the mixture itself that is subjected to feature extraction in order to gain semantical and behavioral information about the constituent sound entities. This is the most common approach in pattern recognition and content analysis applications. The latter corresponds to the above mentioned SO scenario. This taxonomy can be expanded with two further paradigms that correspond to the AQO approach: *separation-without-understanding*, equivalent to BSS, and *understanding-for-separation*, equivalent to SBSS and supervised separation.

In the following subsections, a selection of audio-related applications of source separation, together with their characterization within the above paradigmatic frameworks, will be presented. A final subsection will very briefly mention non-audio applications.

Music Information Retrieval and music transcription

As far as *Audio Content Analysis* (ACA) [28] applications are concerned, source separation is useful under the SO paradigm. In this context, the goal of source separation is to facilitate feature extraction. In most situations, it is easier to analyze partially separated tracks with respect to timbre, pitch, rhythm, structure, etc. than to analyze the mixture itself.

An obvious example is polyphonic transcription [91]. To automatically extract the music score from a digital music signal is an extraordinarily demanding task. There exist robust systems for pitch detection and transcription of single instruments or melodies [74], and some success has been achieved with polyphonic content of two or three voices. However, when it comes to a larger degree of polyphony, the prob-

lem remains unsolved, and it is a matter of debate if it can be achieved in the near future. The problems are common with those of source separation: the overlapping of spectral partials belonging to different voices or instruments, the *tonal fusion* that occurs when several instruments play a voice together, which is then perceived as a single sound entity, and temporal overlaps in rhythmically regular music, which hinder the detection of simultaneous transients or onsets. Most approaches to polyphonic music transcription follow the understanding-without-separation paradigm, and are said to perform *multipitch* estimation [90, 144]. An alternative is to use source separation to obtain the constituent voices, and then perform a more robust monophonic transcription on each of the voices. This is the transcription-related interpretation of the separation-for-understanding paradigm.

The same applies to other *Music Content Analysis* (MCA) and *Music Information Retrieval* (MIR) applications such as musical instrument detection and classification, genre classification, structural analysis, segmentation, query-by-similarity or song identification. It should be noted that, in some cases, using source separation to facilitate these kind of applications involves the derivation of a joint characterization from a set of individual source characterizations. An example of separation-for-understanding system aimed at polyphonic instrument recognition will be the subject of Sect. 4.8.

Viewed from the opposite angle, MCA and MIR techniques applied on the mixture can help (and in some cases, will allow) separation. This corresponds to the understanding-for-separation scenario. For example, detecting the musical instruments present in a mixture can be used to more effectively assign the notes of the mixture to the correct separated sources. All separation systems proposed in Chapters 5 and 6 of this dissertation fall under the understanding-for-separation paradigm.

Unmixing and remixing

Some AQO applications aim at fully unmixing the original mixture into separated sources that are intended to be listened to. This is the most demanding application scenario, and in the musical case is equivalent to generating a multitrack recording from a final mix as contained in the CD release. Ideally, the separated tracks should have a similar quality than they would have had if recorded separately. This can be interesting for archival, educational or creative purposes, but remains largely unfeasible.

A related type of AQO applications concerns the elimination of sources considered undesired. Examples of these include the restoration of old recordings, and denoising for hearing aids or telecommunications. Also belonging to this group is automatic elimination of the singing voice for karaoke systems or of the instrumental soloist for so-called “music-minus-one” recordings aimed at performance practising.

Closely related to the AQO paradigm, and probably more attractive from the practical point of view, is another subset of applications aimed at *remixing* the original mixture. I.e., once a set of separated tracks have been obtained, they are mixed again, with different gains, spatial distributions or effect processings than

the original mixture. This is less demanding than the fully-separated AQO case, since remaining artifacts on the partially separated signals will to a great extent be masked by the other sources present in the remixed version. Examples of remixing applications include the enhancement of relevant sources for hearing aids [125], robot audition [113], upmixing of recordings (e.g., from mono to stereo [94] or from stereo to surround [8]), post-production of recordings when a multitrack recording is not available [180], creative sound transformations for composition, and creation of remixes as cover versions of original songs. A first commercial product using source separation techniques for music post-production (an extension to the Melodyne editor called Direct Note Access) has been announced for release during the first quarter of 2009 by the company Celemony [41]. As of September 2008, no detailed information has been published concerning the capabilities of such a system to separate different instruments (it is primarily intended for the correction or modification of notes within a single-instrument chord), and to what extent it will be able to perform full separation rather than remixing.

Even if not capable of achieving CD-quality separated tracks, all methods aiming at unmixing and remixing can be considered having AQO-separation as an ideal goal, with the lack of quality arising from the limitations of the method. For any given system, the closeness to the AQO scenario will depend on the nature of the mixture with respect to polyphony, reverberation, number of channels, etc.

Audio compression

High-quality lossy audio compression techniques, such as MP3 and AAC, which originated the explosion of online music distribution, exploit psychoacoustical cues to avoid the transmission and storage of masked components of the signals. A new, still experimental, approach to audio coding, named *Structured Audio Coding* (SAC) or *Object-based Audio Coding* [156], has the potential of attaining much lower bitrates for comparable sound qualities. The idea is to extract high-level parameters from the signals, transmit them, and use them at the decoder to resynthesize an approximation to the original signal. Such parameters can be either spectral (such as amplitude and frequency of constituent sinusoids, time and spectral envelopes, spectral shape of noise components), in which case *Spectral Modeling Synthesis* (SMS) [138] or related methods will be used at the decoder, or parameters controlling the physical processes involved in the sound generation, in which case *Physical Modeling Synthesis* (PMS) [145] will be used for reconstruction.

The difficulty of such approaches for the case of complex signals is immediately apparent. Recent, successful research results that report enormous reduction of bitrates, are possible only with simple signals, such as solo passages of single-voiced instruments (see, e.g., the work by Sterling *et al.* [148]). More complex and realistic sound mixtures, much in the same way as for musical transcription, are far more difficult to reduce to a set of spectral or physical parameters. This is again the context in which pre-processing by means of source separation can help in the extraction of such parameters, and thus the extension of the applicability of object-based audio coding to a further level of signal complexity [165].

Non-audio applications

Although not covered by the present work, it is worth mentioning that other applications, unrelated to audio, have arisen in a wide variety of science and engineering fields. For instance, source separation techniques have been applied for image restoration [111], digital communications [47, 129], optical communications [93], electroencephalography (EEG) [45], magnetoencephalography (MEG) [151], analysis of stock markets [9] and astronomical imaging [37].

1.2 Motivations and goals

This dissertation focuses on separation of musical mixtures as an application domain. The main motivation is the use of source separation as a powerful tool in the context of content-analysis applications. Content-based processing lies at the heart of a wide range of new multimedia applications and web services. In the case of audio data, content analysis has traditionally been concentrated on the recognition of single-speaker speech signals. The phenomenal growth of music distribution on the World Wide Web has motivated the extension of Audio Content Analysis to the more challenging field of music signals. Since most music signals are mixtures of several underlying source signals, their semantical characterization poses additional challenges to traditional feature extraction methods. Some authors have reported a “glass ceiling” in performance when extracting traditional speech and music features such as *Mel Frequency Cepstral Coefficients* (MFCC) and using traditional pattern recognition methods such as *Gaussian Mixture Models* (GMM) in applications involving the analysis of complex, mixed signals, such as genre classification or clustering according to musical similarity [7]. Source separation might be the way to break such a barrier.

The fact that most musical mixtures are underdetermined (monaural or stereo, with more than two instruments present) requires taking strong assumptions and/or making simplifications in order for the separation problem to be feasible. On the other hand, constraining the analysis to music allows exploiting several music-specific characteristics, such as spectral envelope smoothness, canonical note-wise temporal envelope shapes (in the form of an Attack-Decay-Sustain-Release envelope), well-defined onsets, rhythmical structure, etc., as cues for the assignment of sound events to separated sources. Both considerations point at the importance of applying an appropriate level of information in the form of *source models* to help facilitate, and in some cases to even make feasible, musical source separation. Source modeling will be the primary guideline of this dissertation.

In this context, the main objective of this work is to contribute new methods for the detection and separation of monaural (single-channel) and stereo (two-channel) linear¹ musical mixtures. The followed approach is to consider source models with increasing level of complexity, and to study both their implications on the separation algorithm and the degree of separation difficulty they are able to cope with.

¹The distinction between linear, delayed and convolutive mixtures will be detailed in Sect. 2.1.

The starting point is sparsity-based separation, which makes a generalist statistical source assumption (not necessarily circumscribed to music). Firstly, the improvement margin that is achievable by optimizing the sparsity of the time–frequency representation of the mixture is investigated. A second level of modeling complexity arises from the combination of a detailed and highly sparse spectral representation (namely, sinusoidal modeling) with novel supervised learning methods aimed at producing a library of models describing the timbre of different musical instruments. As useful by-products of such model-based separation approaches, several MIR applications of the developed methods will be presented: instrument classification of individual notes, polyphonic instrument detection, onset detection and instrument-based segmentation.

1.3 Overview of the thesis

The thematic relationships between the present work’s chapters are schematized in Fig. 1.1. Chapter 2 is a comprehensive overview of source separation principles and methods. It starts by presenting a global framework organized according to the nature of the sources and the mixture to be separated, and to the corresponding various degrees of difficulty. Afterwards, it concentrates on the specific case this work will address: underdetermined separation from linear, noiseless mixtures. Although many of the methods presented in that chapter can be applied to a wide range of signal types, an emphasis is made on audio applications.

Chapter 3 takes an unsupervised (blind) approach and concentrates on evaluating the improvement in separation that is achievable by using nonuniform time and frequency resolutions in the signal representation front-end. In particular, auditory frequency warpings are used as a means of improving the representation sparsity, in combination with a separation system based on stereo spatial diversity.

Chapters 4 and 5 explore a different, complementary conceptual path. They follow the supervised (model-based) scenario, in which some higher-level a priori knowledge about the sources is available. In this work, such knowledge takes the form of a collection of statistical models of the timbre of different musical instruments. Chapter 4 presents and evaluates the novel modeling approaches proposed to that end. A salient characteristic of the modeling technique is its detailed consideration of the temporal evolution of timbre. Although originally intended for source separation applications, the proposed models can be useful to other content analysis applications. In that chapter, they are indeed subjected to evaluation for two non-separation purposes: musical instrument classification and detection of instruments in polyphonic mixtures. The chapter also introduces several important spectral analysis techniques on which the models are based, in particular, sinusoidal modeling and spectral envelope estimation. Chapter 5 exploits all these techniques and the developed models and presents a system aiming at the most demanding separation scenario: separation from a single-channel (monaural) mixture.

In Chapter 6, several ideas from both unsupervised and model-based scenarios are combined to develop *hybrid* systems for the separation of stereo mixtures. More

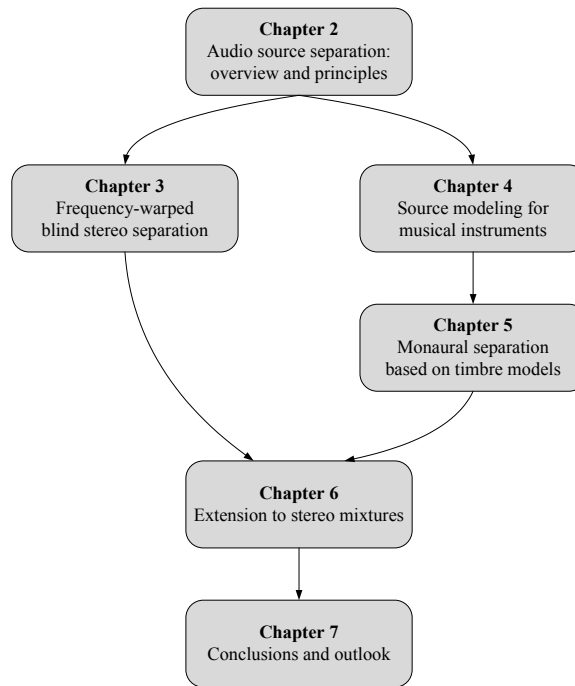


Figure 1.1: Chart of thematic dependencies.

specifically, sparsity-based separation is used to exploit the spatial diversity cues, and sinusoidal and timbre modeling are used to minimize interferences and thus improve separation. Finally, Chapter 7 summarizes the results and contributions, and proposes several directions to further develop the different modeling and separation methods, and to adapt them for other sound analysis or synthesis applications.

Several sound examples resulting from the experiments performed throughout the present work are available online². All algorithms and experiments reported in this work were implemented using MATLAB.

²<http://www.nue.tu-berlin.de/people/burred/phd/>

2


Audio source separation: overview and principles

Source separation from sound mixtures can arise in a wide variety of situations under different environmental, mathematical or practical constraints. The present work addresses a specific problem of audio source separation, namely that of separation from instantaneous musical mixtures, either mono or stereo. It is however useful to consider first a panoramic overview, so that the implications, requirements and utility of the particular problem considered can be put into context. Tables 2.1 and 2.2 show a classification of audio source separation tasks according to the nature of the sources, and to the amount of available a priori knowledge, respectively. The entries in each column are sorted by decreasing separation difficulty.

Obviously, separation is more difficult if sources are allowed to move, which requires an additional source tracking stage. By far, most systems assume that the sources are static. The mixing process can be either instantaneous (the sources add linearly), delayed (the sources are mutually delayed before addition) or echoic (convolutive) with static or changing room impulse response. The last case represents the most natural and general situation. However, under controlled recording conditions in a studio or with appropriate software, the simpler models are applicable. Each mixing situation corresponds to a different mathematical model, all of which will be introduced later on the chapter.

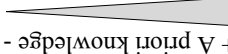
A crucial factor determining the separation difficulty is the number of sources related to the number of available mixtures. Separation is said to be *underdetermined* or *degenerate* if there are more sources than observed mixtures, *overdetermined* if there are more mixtures than sources and *even-determined* or simply *determined* if there are the same number of sources than mixtures. The underdetermined case is the most difficult one since there are more unknowns than observed variables, and the problem is thus ill-posed. Also, noise-free separation will obviously be easier than noisy.

The last two columns in Table 2.1 concern musical mixtures, which have several distinctive features that are decisive in assessing how demanding the separation will be. A crucial factor is musical texture, which refers to the overall sound quality as determined mainly by the mutual rhythmic features between constituent voices. The most difficult musical texture to separate is multiple-voiced *monody* or *monophony*, in which several parallel voices exactly follow the same melody, sometimes separated



Source position	Mixing process	Source/mixture ratio	Noise	Musical texture	Harmony
<ul style="list-style-type: none"> • changing • static 	<ul style="list-style-type: none"> • echoic (changing impulse response) • echoic (static impulse response) • delayed • instantaneous 	<ul style="list-style-type: none"> • underdetermined • overdetermined • even-determined 	<ul style="list-style-type: none"> • noisy • noiseless 	<ul style="list-style-type: none"> • monodic (multiple voices) • heterophonic • homophonic / homorhythmic • polyphonic / contrapuntal • monodic (single voice) 	<ul style="list-style-type: none"> • tonal • atonal

Table 2.1: Classification of audio source separation tasks according to the nature of the mixtures.



Source position	Source model	Number of sources	Type of sources	Onset times	Pitch knowledge
<ul style="list-style-type: none"> • unknown • statistical model • known mixing matrix 	<ul style="list-style-type: none"> • none • statistical independence • sparsity • advanced/trained source models 	<ul style="list-style-type: none"> • unknown • known 	<ul style="list-style-type: none"> • unknown • known 	<ul style="list-style-type: none"> • unknown • known (score/MIDI available) 	<ul style="list-style-type: none"> • none • pitch ranges • score/MIDI available

Table 2.2: Classification of audio source separation tasks according to available a priori information.

by one or more octaves. This situation obviously implies the largest degree of spectral and temporal overlapping. A paradigmatic example of monodic music is Gregorian chant.

The next texture by decreasing degree of overlapping is *heterophony*. Like monody, it basically consists of a single melody. However, different voices or instruments can play that same melody in different ways, for example by adding melodic or rhythmic ornamentation. Heterophonic textures appear, for example, in western medieval music and in the musical tradition of several Asian countries.

The *homophonic* or *homorhythmic* texture denotes a set of parallel voices that move under the same or very similar rhythm, forming a clearly defined progression of chords. Examples of homophonic music include anthems and chorales. Homophony is said to be melody-driven if there is a pre-eminent voice that stands out of the ensemble, with the rest constituting a harmonic accompaniment. Melody-driven homophony is the most usual texture in songs, arias, and in most genres of popular western music, such as pop and rock. In general, homophony is difficult to separate because of the high degree of temporal overlapping and, in the case of tonal music, also because of frequency-domain overlapping.

Polyphonic or *contrapuntal* textures correspond to highly rhythmically independent voices, such as in fugues. In this case, the probability of overlaps will be obviously lower. Finally, for the sake of completeness, single-voiced monody has been included in the table, although it has just a single source and is thus trivial for separation.

The other important musical factor is harmony, which refers to the mutual relationships between simultaneously-sounding notes. In *tonal* music, concurrent pitches are most often in simple numerical relationships, corresponding to the most consonant intervals, such as octaves, fifths, fourths and thirds. Western classical music, from Gregorian chant to the early 20th century, and most popular music are pre-eminently tonal. Such harmonic relationships between pitches makes separation more difficult, since the harmonic components of a note will often overlap with those of other concurrent notes. *Atonal* music, consisting mainly of dissonant intervals, will in contrast be easier to separate.

Like with any other kind of analysis, the more information about the problem is available beforehand, the easier the separation becomes (Table 2.2). The mixing process, which as will be seen is mathematically described by a *mixing matrix*, is assumed to be unknown in almost all separation approaches. In fact, in the very usual even-determined situation, source separation equals to the problem of estimating the mixing matrix, as will be discussed in Sect. 2.7. Sometimes, a statistical model of the mixing matrix is assumed. Note that the mixing process reflects the position of the sources, and thus knowing the mixing process amounts to knowing the source positions. The term “blind” in *Blind Source Separation* (BSS) refers mainly to the fact that the mixing process is unknown.

To improve separation, and sometimes to make it actually possible, statistical features of the temporal or spectral nature of the signals are exploited. Statistical independence is almost always assumed in even-determined separation scenarios and sparsity, a stronger concept, in underdetermined ones. At the cost of being signal-

specific (and thus no longer fully “blind”), many algorithms use more sophisticated signal models that describe more closely the perceptual, rhythmic or timbral features of the signals to be separated. This is especially useful, and in some cases absolutely necessary, in highly underdetermined situations, such as separation from single-channel mixtures. These models can even be trained beforehand on signal example databases. Signal modeling for source separation plays a central role in the present work, and it will be a recurrent topic throughout all the chapters. Finally, it is obvious that knowing the number of sources and their type (e.g., which musical instruments are present) will facilitate separation.

Again, the last two columns of the table concern musical features. Several approaches need a detailed knowledge about either the note onsets (i.e., their temporal starting points), their pitches, or both, to make separation reliable in highly underdetermined and overlapping mixtures. This knowledge takes often the form of a previously available MIDI score.

Within this problem classification, the approaches developed and reported in the present work address the separation from static, instantaneous, underdetermined, noiseless musical mixtures. The source positions, onset times and pitches will always be assumed unknown. Different signal models will be addressed, most importantly sparse and trained source models. Depending on the context and on the particular experiment, the number and type of sources will be known beforehand or assumed unknown.

The present chapter introduces the basic principles of BSS and provides an overview of the approaches most directly related to the context of this work. Sections 2.1 to 2.4 cover a general theoretic framework. In Sect. 2.1 all mixing models that can be encountered in a BSS problem are presented: the linear, delayed and the convolutive mixing models, as well as their reformulation when noise is present. Section 2.2 presents the real-world stereo mixing and recording situations in which the different mixing models can be applied. Section 2.3 reviews signal decompositions and transformations within a common framework, including the *Discrete Fourier Transform* (DFT), the *Short Time Fourier Transform* (STFT), *Principal Component Analysis* (PCA) and sparse representations. The most basic of them are obviously not specifically intended for source separation, but greatly facilitate the process, as will be seen in detail. Indeed, many of the signal models presented in that section are polyvalent, and will be used for different purposes throughout the present work. Section 2.4 focuses on the close relationship between the problems of source separation and signal decomposition and provides a general framework that combines both, and upon which the presented methods are based.

Sections 2.5 to 2.8 constitute a state-of-the-art review of linear, noiseless and underdetermined separation methods. Section 2.5 illustrates the most general approach to it: the joint estimation of the mixing matrix and of the sources, and presents its limitations. Methods following the alternative approach of sequentially estimating the mixing matrix in the first place and the sources in the second place are presented in the next two sections. In particular, Sect. 2.6 covers the most important methods for mixing matrix estimation, including *Independent Component Analysis* (ICA), clustering methods, phase-cancellation methods and methods from

image processing. Approaches to re-estimate the sources once the mixing conditions are known are presented in Sect. 2.7, with special emphasis on norm-minimization algorithms, which are the most important ones and also the ones that will be used throughout this work. Finally, Sect. 2.8 provides a short overview of approaches arising from the fields of psychoacoustics and physiology, referred to globally as *Computational Auditory Scene Analysis* (CASA). Comprehensive overviews of methods for audio source separation can also be found in O’Grady *et al.* [117] and Vincent *et al.* [164].

2.1 Mixing models

When two or more sound waves coincide in a particular point in space and at a particular time instant, the displacement¹ of the resulting mixed wave is given by the sum of the displacements of the concurrent waves, as dictated by the *principle of superposition*. In the most general case, the interfering waves can propagate in different directions, and thus the net displacement must be obtained by vector addition.

When a microphone transduces a sound wave into an electrical oscillation, the information about the propagating directions gets lost and the wave is reduced to the pattern of vibration of the capturing membrane, modeled as a one-dimensional, time domain signal $x(t)$. At most, the direction of impingement will affect the overall amplitude of the transduced signal, according to the directionality pattern of the microphone (see Sect. 2.2). However, once in the electrical domain, signals always interfere unidimensionally, and thus the net displacement of a signal mixture $x(t)$ of N signals $y_n(t)$, $n = 1, \dots, N$ is given by scalar addition of the corresponding instantaneous amplitudes:

$$x(t) = \sum_{n=1}^N y_n(t). \quad (2.1)$$

It should be noted that the signals $y_n(t)$ to which such a universally valid linear mixture formulation refers are the vibration patterns at the point in which the actual mixing takes place, i.e., either the microphone membrane or an (often conceptual) point in the electrical system where signals are artificially added. In source separation, however, the interest lies in retrieving the constituent signals as they were at the point they were produced, i.e., at the sources. The different mixing conditions are thus reflected in the way the source signals $s_n(t)$ are transformed into their *source images* $y_n(t)$, before being added to produce a particular mixture. According to these mixing conditions, three mathematical formulations of the mixing process can be defined: the *linear*, the *delayed* and the *convolutive mixing models*. All three will be introduced in this section, while the next section will illustrate to which real-world situations each of the models can apply. The linear mixing model, being the one the present work is based on, will be covered more in depth.

¹Not to be confused with the amplitude, which is the maximum displacement during a given time interval, usually a cycle of a periodic wave. Displacement can also be called instantaneous amplitude.

All signals considered here are discrete, making the academic distinction between continuous (t) and discrete $[n]$ independent time variables unnecessary. The notation (t) has been chosen for clarity. Source signals will be denoted by $s(t)$ and indexed by $n = 1, \dots, N$. Mixture signals will be denoted by $x(t)$ and indexed by $m = 1, \dots, M$. The term “mixture” will refer to each individual channel $x_m(t)$, in contrast with the audio engineering terminology, where “mix” refers to the collectivity of channels (as in “stereo mix”, “surround mix”).

2.1.1 Instantaneous mixing model

The linear or instantaneous mixing model assumes that the only change on the source signals before being mixed has been an amplitude scaling:

$$x_m(t) = \sum_{n=1}^N a_{mn}s_n(t), \quad m = 1, \dots, M. \quad (2.2)$$

Arranging this as a system of linear equations:

$$\begin{cases} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) + \dots + a_{1N}s_N(t) \\ \vdots & \\ x_M(t) &= a_{M1}s_1(t) + a_{M2}s_2(t) + \dots + a_{MN}s_N(t) \end{cases}, \quad (2.3)$$

a compact matrix formulation can be derived by defining the $M \times 1$ vector of mixtures $\mathbf{x} = (x_1(t), \dots, x_M(t))^T$ and the $N \times 1$ vector of sources $\mathbf{s} = (s_1(t), \dots, s_N(t))^T$:

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_M(t) \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M1} & a_{M2} & \dots & a_{MN} \end{pmatrix} \cdot \begin{pmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_N(t) \end{pmatrix}, \quad (2.4)$$

obtaining

$$\mathbf{x} = \mathbf{A}\mathbf{s}, \quad (2.5)$$

where \mathbf{A} is the $M \times N$ *mixing matrix* whose generic element a_{mn} is the gain factor, or mixing coefficient, from source n to mixture channel m . Alternatively, the signal vectors can be represented as matrices of individual samples of size $M \times T$ or $N \times T$, where T is the length of the signals in samples, resulting in the notation $\mathbf{X} = \mathbf{A}\mathbf{S}$. The notation $\mathbf{x} = \mathbf{A}\mathbf{s}$ will be referred to as *instantaneous notation*, and $\mathbf{X} = \mathbf{A}\mathbf{S}$ will be referred to as *explicit notation*.

Such formulations are called *generative* or *latent variable* models since they express the observations \mathbf{x} as being generated by a set of “hidden”, unknown variables \mathbf{s} . Note that both expressions can also be interpreted as a linear transformation of the signal vector or matrix into the observation vector or matrix, in which \mathbf{A} is the transformation matrix and its columns, denoted by \mathbf{a}_n , are the transformation bases.

The goal of linear source separation is, given the observed set of mixtures \mathbf{x} , to solve such a set of linear equations towards the unknown \mathbf{s} . However, in contrast to

basic linear algebra problems, the system coefficients a_{mn} are also unknown, which makes it a far more difficult problem which, as will be shown, must rely on certain signal assumptions.

In linear algebra, a system with more equations than unknowns is called *overdetermined*, and has often no solution, even if the coefficients are known. A system with less equations than unknowns is called *underdetermined*, and will mostly yield an infinite number of solutions if no further a priori assumptions are met. A system with the same number of equations than unknowns is said to be *determined* or *even-determined* and will most likely have a single solution with known coefficients. In BSS this terminology has been retained for problems where there are more mixtures than sources ($M > N$, overdetermined BSS), less mixtures than sources ($M < N$, underdetermined) and the same number of sources than mixtures ($M = N$, even-determined).

2.1.2 Delayed mixing model

The delayed generative model, sometimes called anechoic, is valid in situations where each source needs a different time to reach each sensor, giving rise to different source-to-sensor delays δ_{mn} :

$$x_m(t) = \sum_{n=1}^N a_{mn} s_n(t - \delta_{mn}), \quad m = 1, \dots, M. \quad (2.6)$$

A matrix formulation can be obtained by defining the mixing matrix as

$$\mathbf{A} = \begin{pmatrix} a_{11}\delta(t - \delta_{11}) & \dots & a_{1N}\delta(t - \delta_{11}) \\ \vdots & \ddots & \vdots \\ a_{M1}\delta(t - \delta_{M1}) & \dots & a_{MN}\delta(t - \delta_{MN}) \end{pmatrix}, \quad (2.7)$$

where a_{mn} are the amplitude coefficients and $\delta(t)$ are Kronecker deltas², and rewriting the model as

$$\mathbf{x} = \mathbf{A} * \mathbf{s}, \quad (2.8)$$

where the operator $*$ denotes element-wise convolution.

2.1.3 Convolutional mixing model

A convolutional generative model applies if there is a filtering process between each source and each sensor. The impulse response that models the filtering between source n and mixture m will be denoted by $h_{mn}(t)$. In order to employ the previous notation of amplitude coefficients and deltas, each filter can be written out as

$$h_{mn} = \sum_{k=1}^{K_{mn}} a_{mnk} \delta(t - \delta_{mnk}), \quad (2.9)$$

²The Kronecker delta is defined as $\delta(t) = \begin{cases} 1 & \text{if } t = 0 \\ 0 & \text{if } t \neq 0 \end{cases}$.

where K_{mn} is the length of that particular impulse response (FIR filters are assumed). Then, the mixture at each sensor is given by

$$x_m(t) = \sum_{n=1}^N h_{mn}(t) * s_n(t) = \sum_{n=1}^N \sum_{k=1}^{K_{mn}} a_{mnk} s_n(t - \delta_{mnk}), \quad m = 1, \dots, M. \quad (2.10)$$

and the mixing matrix is in effect a matrix of FIR filters

$$\mathbf{A} = \begin{pmatrix} h_{11}(t) & \dots & h_{1N}(t) \\ \vdots & \ddots & \vdots \\ h_{M1}(t) & \dots & h_{MN}(t) \end{pmatrix}, \quad (2.11)$$

that can be used again in a convolutive formulation of the form $\mathbf{x} = \mathbf{A} * \mathbf{s}$.

The most typical application of this model is to simulate room acoustics in reverberant environments, the reason for which it is often called reverberant or echoic mixing model. In such a situation, the length of the filters K_{mn} correspond to the number of possible paths the sound can follow between source and sensor, and a_{mnk} and δ_{mnk} to their corresponding attenuations and delays, respectively. Note that the delayed mixing model is a particular case of the convolutive model for which $K_{mn} = 1$ for all m, n .

2.1.4 Noisy mixture models

All the above mixing models can be adapted to the case where additive noise is present by adding a noise vector of the same dimensions as the mixture vector. For instance, in the linear case this will be denoted by

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n} \quad (2.12)$$

or by the explicit notation $\mathbf{X} = \mathbf{A}\mathbf{S} + \mathbf{N}$. The noise is often assumed to be white, Gaussian, and uncorrelated, i.e., having a diagonal covariance matrix of the form $\sigma^2 \mathbf{I}$, where σ^2 is the variance of one of its M components. Furthermore, the noise is assumed to be independent from the sources.

All separation approaches throughout the present work are modeled as noise-free. However, the noisy mixture model will be useful to illustrate the derivation of the general probabilistic framework for BSS in Sect. 2.5.

2.2 Stereo recording techniques

A brief overview of stereo recording and mixing techniques will help to assess the usefulness of each of the models defined in the previous section, and their correspondence and applicability to real-world situations. To date, stereophony is still the most common format for sound recording and reproduction. Although multi-channel³ techniques, most typically 5.1 surround speaker systems for playback and

³ “Multichannel” will be used to denote any system with more than 2 channels. Stereo will not be considered multichannel.

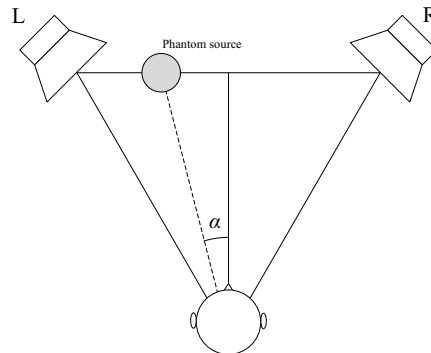


Figure 2.1: Ideal stereo reproduction setup with azimuth angle.

DVDs for storage, are increasingly affordable and widespread, they still have not superseded two-channel systems. The long-standing success of stereo (the first commercial stereo recording, on vinyl disk, was released in 1958) can be explained by its appropriate trade-off between cost and spatial fidelity, and, especially nowadays, by its suitability for headphone listening. The vast majority of CDs, compressed formats such as MP3 or AAC, FM radio broadcasts, as well as many analogue and digital TV broadcasts, are in stereo.

Technically, any recording technique aiming at simulating the spatial conditions in the recording venue can be termed “stereophonic”. The word was derived from a combination of the Greek words “stéreos” (meaning “solid”) and “phōnē” (“sound”), by analogy with stereoscopic or three-dimensional imaging. Although more modern multichannel techniques like surround systems and *Wave Field Synthesis* (WFS) [20] are capable of much more realistic spatial simulations, the word “stereo” has been relegated by common usage only to two-channel systems. The term is also applied to any two-channel synthetic mix, even if not necessarily aimed at resembling natural spatial conditions.

Stereo reproduction is based on the fact that, if a particular source is appropriately scaled and/or delayed between the left and right channels, it will appear to originate from an imaginary position (the so-called *phantom sound source*) along the straight line connecting both loudspeakers (the *loudspeaker basis*). The *azimuth* α is the angle of incidence of the phantom sound source to the listener, and it depends on the relative position of the listener to the loudspeakers (see Fig. 2.1). To perceive the correct direction, the listener must be on the vertex completing an equilateral triangle with the loudspeakers, the so-called *sweet spot*. In this ideal case, the azimuth, measured from the center, can lie on the range $\alpha = [-30^\circ, 30^\circ]$. To make the source position indication independent from the position of the listener, left to right level ratios are used instead, such as denoting a “hard-right” source with 100%R, a middle source with 0% and a “hard-left” source by 100%L. Another possibility uses the polar coordinates convention and assigns 0° degrees or 0 radians to hard-right and 180° or π radians to hard-left. This is the most appropriate approach for source

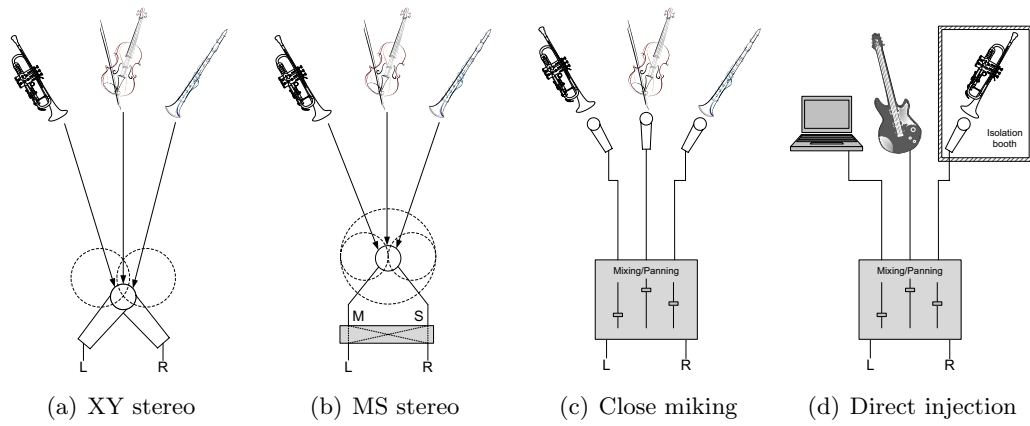


Figure 2.2: Instantaneous stereo recording techniques.

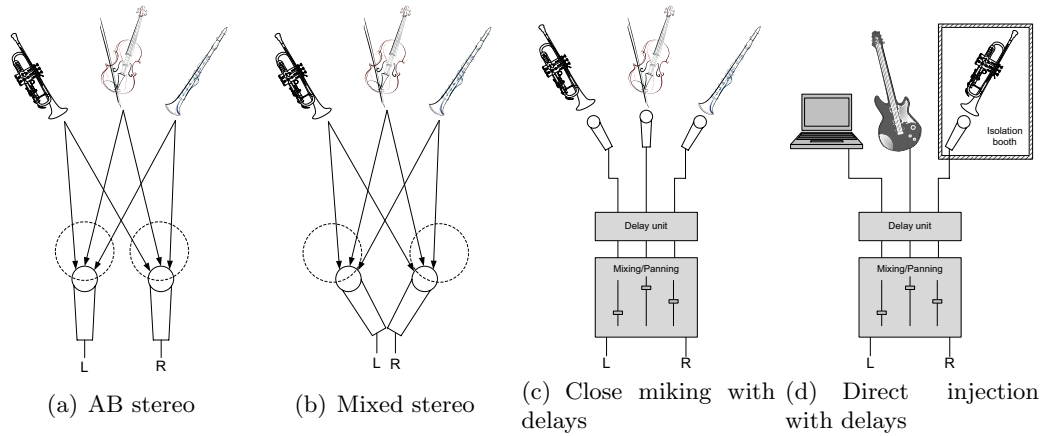


Figure 2.3: Delayed stereo recording techniques.

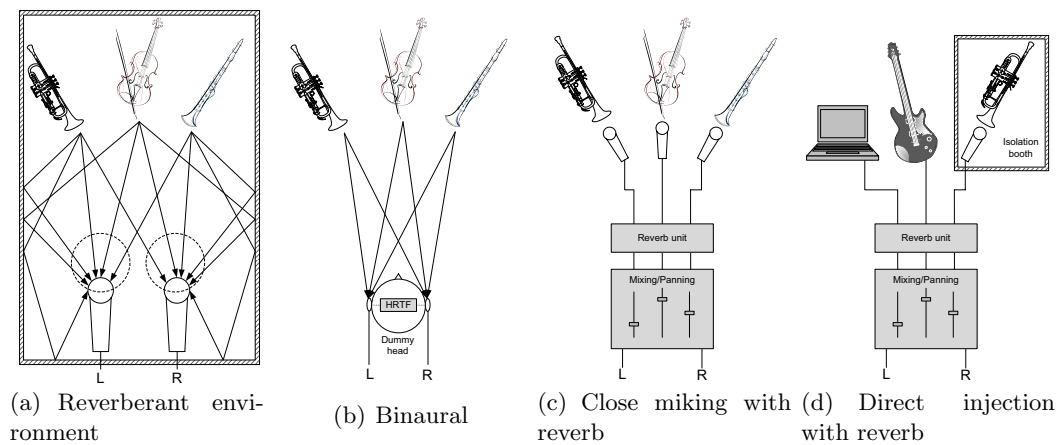


Figure 2.4: Convulsive stereo recording techniques.

separation algorithms (see Sect. 2.6), and is the one that will be used in this work. It should be noted that, although being an angular magnitude, it does not correspond to the perceived angle of incidence, except if the listener is located exactly between both loudspeakers (as in the case of headphone listening). Although technically inaccurate, the term azimuth has been also used to denote stereo source locations independent from the position of the listener, such as in [13].

A general distinction will be made between natural mixtures and synthetic mixtures. Natural mixing refers to recording situations in which the mixing parameters are determined by the relative positions of a set of acoustic sound sources and the microphones. In contrast, synthetic mixing consists of artificially combining a set of perfectly or near-perfectly separated sound sources using a mixing desk or mixing software. Traditionally, natural techniques are preferred for classical music to ensure a truthful reflection of intensity proportions between instruments and of room acoustics, whereas artificial techniques are most common in popular genres, in which studio post-processing effects play a crucial role.

It should be noted that the distinction between natural and synthetic mixtures does not necessarily correspond to the distinction between live and overdub-based⁴ studio recordings. An ensemble can play live in a studio in separated isolation booths, or using directional microphones placed closely to the instruments, or in the case of electrical instruments, directly connected to the mixing desk. On the other hand, overdubs can be made of performers playing at different positions relative to a microphone pair. These are certainly not the most common situations, but are possibilities to be considered. Also, both kinds of methods can obviously be combined in a final musical production. However, for the sake of clarity, it will be supposed here that each mixture is based on a single technique.

Intensity stereophony: XY and MS techniques

As mentioned, a linear stereo ($M = 2$) mixing model (Sect. 2.1.1) applies in the case where the sources are multiplied by a scalar before being added. Thus, stereo localization results solely from the difference in intensity between both channels, which is termed *Interaural* or *Inter-channel Intensity Difference* (IID). Several natural and synthetic scenarios fulfill this. One of them is *intensity stereophony*, which is a natural mixing method involving a pair of microphones whose membranes are located at the same spot. In such an arrangement, the stereo effect is obtained by exploiting the directionality properties⁵ of the microphones. The most common approaches of intensity stereophony are *XY stereophony* and *MS stereophony* [52].

With the XY technique, both microphones are directional and the stereo effect is achieved by mutually rotating them to a certain angle, usually 90° . This setup is represented graphically on Fig. 2.2(a), where the directivity patterns of the micro-

⁴Overdubbing refers to the process of adding a new track to a set of previously recorded tracks.

⁵The directionality or polar pattern of a microphone indicates its sensitivity to sound pressure as a function of the angle of arrival of the waves. If a microphone is most sensitive to a particular direction, it is termed directional. Bidirectional microphones are equally sensitive in two diametrically opposed directions. Omnidirectional ones are equally sensitive to any direction.

phones are denoted by the dashed circles. The sources' direct sound waves, arriving from different directions and distances, will be picked up with different intensities, depending on the angle of impingement.

The MS (for Mid/Side) technique employs one bidirectional and one directional (alternatively omnidirectional) microphone at the same place arranged such that the point of maximum directivity of the directional microphone lies at an angle of 90° from either bidirectional maximum (see Fig. 2.2(b)). In this way, a central channel \mathbf{x}_M and a lateral channel \mathbf{x}_S are obtained, which are then transformed into the left/right channels by

$$\mathbf{x}_L = \frac{1}{\sqrt{2}}(\mathbf{x}_M + \mathbf{x}_S), \quad (2.13)$$

$$\mathbf{x}_R = \frac{1}{\sqrt{2}}(\mathbf{x}_M - \mathbf{x}_S). \quad (2.14)$$

An advantage of the MS system is its total compatibility with mono reproduction: the middle signal directly corresponds to the mono signal, avoiding possible phase cancellations and level imbalances that can appear when adding two separated stereo channels. Assuming ideal anechoic conditions, both XY and MS approaches can be described by the linear mixing model, since direction and distance of the sources result both only in gain differences.

Close miking and direct injection

If several highly directional microphones are located close to the instruments, and again good acoustic absorption of the recording environment is assumed, then the source signals can be considered to be nearly perfectly separated and susceptible of being synthetically mixed (see Fig. 2.2(c)). Obviously, electrical and electroacoustic instruments, as well as any other kind of electronic sound generators such as samplers, synthesizers and computers running synthesis software, can be directly connected to the mixing unit, offering perfect a priori separation⁶ (Fig. 2.2(d)). A perfectly separated source can be also obtained by recording the instrument in an isolation booth, as it is often done with singers.

These two recording methods are most useful for the evaluation of source separation performance, since the perfectly separated original sources are available a priori and can be then used as a baseline for comparison.

Panning

Mixing desks and their software counterparts operate by attenuating and *panning* each channel independently before being added. Panning, a term referring to the panoramic potentiometer that implements it, means to assign an artificial stereo position to a particular channel. This is achieved by sending two differently scaled versions of the input channel to the output left and right channels. By choosing

⁶A notable exception are electrical guitars, which are often recorded by placing a microphone very close to the amplifier in order to capture a richer sound.

the appropriate scaling ratios, a source can be perceived as originating from an imaginary position lying on the line connecting both loudspeakers, thus emulating the conditions of natural recording. An attenuation of around 3 dB should be performed on sources intended to appear near the middle position in order to keep a constant global power level. In effect, panning acts as an additional stage of amplitude scaling, which justifies the applicability of the linear model.

Time-of-arrival stereophony

The delayed mixing model must be used if the sources arrive at the sensors at different times. Thus, not only the IID determine the stereo position, but also the so-called *Interaural* or *Inter-channel Phase Differences* (IPD). In natural recording setups, this happens when the microphones are separated from each other. This is the case of *time-of-arrival stereophony*, whose basic microphone arrangement is the AB technique. In this case, two (usually omnidirectional) microphones are placed in parallel a certain distance apart. The sources will arrive with different amplitudes and with different delays to each one of them (see Fig. 2.3(a)).

The same applies to the so-called *mixed stereophony* techniques, where the separation of the microphones is combined with the orientation at a given angle, thus exploiting principles from both intensity and time-of-arrival methods (see Fig. 2.3(b)). Examples of this approach are the ORTF (Office de Radiodiffusion-Télévision Française), OSS (Optimal Stereo Signal) and NOS (Nederlandse Omroep Stichting) stereo microphone arrangements.

All such delayed stereo techniques allow a more realistic spatialization than intensity-based methods, but have the drawback of adulterating mono downmixes due to phase cancellations. In synthetic mixing environments, the time-of-arrival differences can be simulated by delay units (Figs. 2.3(c) and 2.3(d)).

Recording setups involving convolution

The convolutive mixing model is applicable if the sources get filtered before being mixed. The most common situation is natural recording in a reverberant environment, in which case the filters correspond to the impulse responses of the recording room, evaluated between each possible source and each microphone. All of the previously mentioned natural stereo techniques should be approximated by the convolutive model as long as there is an important effect of room acoustics on the recording (Fig. 2.4(a)).

Another relevant convolutive technique is *binaural recording*⁷, which refers to the use of a “dummy head” that simulates the acoustic transmission characteristics of the human head. It contains two microphones that are inserted at the location of the ears (Fig. 2.4(b)). Binaural recordings offer excellent spatial fidelity as long as they are listened on headphones. The signals arrive at each microphone not only with intensity differences and delays, but also filtered by the head. The

⁷The word “binaural” is often incorrectly used as a synonym for “stereo”, probably because of its analogy with the term “monaural”.

corresponding transfer functions are called *Head Related Transfer Functions* (HRTF) and, if appropriately measured, can be used to simulate binaural recordings via software using a conventional microphone pair.

Finally, any spatial effect introduced in a synthetic mixing process (most typically artificial reverbs) makes the mixture convolutive (Figs. 2.4(c) and 2.4(d)).

2.3 Basic signal models

Nearly all digital signal processing techniques rely on the assumption that the signals can be approximated by a weighted sum of a set of expansion functions. In the time domain, such an *additive expansion* or *decomposition* can be expressed as

$$s(t) = \sum_{k=1}^K c_k b_k(t), \quad (2.15)$$

where K is the number of expansion functions, c_k are the expansion coefficients and $b_k(t)$ are the time-domain expansion functions. The usefulness of such kind of model arises from the superposition property of linear systems, which allows evaluating how a system T transforms a signal by separately computing the outputs of the system to the more simple constituent decomposition functions:

$$T \left\{ \sum_{k=1}^K c_k b_k(t) \right\} = \sum_{k=1}^K c_k T \{ b_k(t) \}. \quad (2.16)$$

The choice of the decomposition functions will depend on the application context. As it will be introduced in detail in Sect. 2.3.3, and often referred throughout this work, crucial to source separation is the criterion of sparsity, which aims at finding a set of decomposition functions in such a way that a reasonable approximation of the signal is possible with most of the expansion coefficients equal or close to zero.

Most well-known signal transformations and analysis methods are specific cases of the discrete additive model of Eq. 2.15. The trivial case is the interpretation of that equation as the *sifting property* of discrete signals [146], by using shifted impulses as the expansion functions: $b_k(t) = \delta(t - k)$. The coefficients c_k correspond then to the sample amplitude values. Basic discrete spectral transforms such as the *Discrete Fourier Transform* (DFT) and the *Discrete Cosine Transform* (DCT) are additive expansions with a finite set of frequency-localized expansion functions fixed beforehand. If the decomposition functions are also localized in time, the result is a time–frequency representation, such as offered by the *Short-Time Fourier Transform* (STFT) and the *Discrete Wavelet Transform* (DWT), as well as any arbitrary decimated filter bank arrangement, such as the ones used for frequency-warped representations.

If the set of expansion functions is not fixed beforehand, and depends on the signal to be analyzed, the expansion is said to be adaptive or data-driven. There are several ways in which such an adaptivity can be implemented. One possibility is to define a fixed collection of basis functions, called a *dictionary*, and select out

of it the bases that best match the observed signal. This is the principle behind overcomplete and sparse decomposition methods, such as *Basis Pursuit* [42] and *Matching Pursuit* [104]. Another possibility is to extract the expansion functions directly from the signal, resulting in adaptive transforms like PCA [82] and ICA [79]. An even more sophisticated approach consists in considering time-varying expansion functions whose parameters are to be extracted from temporal segments of the input signal. This is the case of sinusoidal modeling and its variants, which will be thoroughly reviewed in Chapter 4. An excellent overview of all these types of modeling approaches, considered under a common mathematical framework, can be found in Michael Goodwin’s PhD thesis [65].

A different family of modeling approaches approximate a given signal by prediction, rather than by expansion: they assume that the current output has been generated in some way from the previous outputs. The most basic model of this type, the *autoregressive* (AR) model plays an important role in the estimation of spectral envelopes, and will be introduced within that context in Sect. 4.1.

In this section, basic fixed (STFT) and data-driven (PCA) expansions relevant to the present work will be introduced. More advanced models will be introduced in subsequent chapters: frequency-warped representations in Sect. 3.1, sinusoidal modeling and trained models in Chapter 4, and other source-specific advanced models for musical signals in Sect. 5.1.

2.3.1 Basis decompositions

If the discrete signal to be modeled and the expansion functions are constrained to a finite-length interval $t = 0, \dots, T - 1$ and, using the vector notation $\mathbf{s} = (s(0), \dots, s(T - 1))^T$ for the signal and $\mathbf{c} = (c_1, \dots, c_K)^T$ for the coefficients corresponding to the K expansion functions $b_k(t) = \mathbf{b}_k = (b_k(0), \dots, b_k(T - 1))^T$, it is possible to express Eq. 2.15 in matrix notation:

$$\mathbf{s} = \mathbf{B}\mathbf{c}, \tag{2.17}$$

where \mathbf{B} is a $T \times K$ matrix whose columns are the functions \mathbf{b}_k . This can be interpreted as a linear transformation from the coefficient space to the signal space, with \mathbf{B} as the transformation matrix and \mathbf{b}_k as the transformation bases. Note that such a linear decomposition model is of the same form than the linear mixing model of Eq. 2.5. In fact, there is a strong analogy between source separation and signal decomposition, as will be addressed more in detail in Sect. 2.4.

For multidimensional signals of N dimensions (or equivalently, for sets of N different signals of the same length T), the already introduced explicit notation will be used, with the following convention: variables will be arranged as rows and their realizations (samples) will correspond to the columns. Thus, the formulation of basis decomposition will be of the form $\mathbf{S} = \mathbf{C}\mathbf{B}^T$, with the N signals and coefficient vectors arranged as the rows of matrices \mathbf{S} (size $N \times T$) and \mathbf{C} (size $N \times K$), respectively.

If $T = K$ and the columns of \mathbf{B} are linearly independent, then the set of expansion functions constitutes a *basis* of the signal space, meaning that each signal vector

\mathbf{s} can be represented as a linear combination of the \mathbf{b}_k 's, which are then called *basis functions*. In this case, the basis decomposition is said to be *complete*. If, however, $T < K$, the matrix \mathbf{B} contains linearly dependent vectors and the representation is said to be *overcomplete* [65].

In the complete case, the transformation matrix is invertible, and the expansion coefficients can be readily obtained by

$$\mathbf{c} = \mathbf{B}^{-1}\mathbf{s}. \quad (2.18)$$

In the context of signal transformations, Eq. 2.18 is called the *analysis equation* and Eq. 2.17 is called the *synthesis equation*. By convention, the analysis equation is considered the direct transformation and the synthesis equation is considered the inverse transformation.

A further simplification is possible if the basis is orthogonal⁸, in which case the coefficients are directly given by projecting the signal upon each one of the basis functions:

$$c_k = \langle \mathbf{b}_k, \mathbf{s} \rangle = \mathbf{b}_k^H \mathbf{s}, \quad (2.19)$$

where $\langle \cdot \rangle$ denotes the scalar (or dot) product and H the Hermitian (complex conjugate) transpose, or, in matrix notation, $\mathbf{c} = \mathbf{B}^H \mathbf{s}$. This results in an expansion of the form

$$\mathbf{s} = \sum_{k=1}^K \langle \mathbf{b}_k, \mathbf{s} \rangle \mathbf{b}_k, \quad (2.20)$$

which is called the *orthogonal projection* of \mathbf{s} onto the set of bases \mathbf{b}_k .

Throughout the present work, basis decomposition methods will appear within several different contexts. The DFT is the fundament of the STFT and, as such, of sinusoidal modeling. PCA, an adaptive basis decomposition, will be applied to obtain compact spectral representations in Chapter 4. ICA is closely related to the angular clustering method for estimating the mixing matrix used in Chapters 3 and 6. The DFT and PCA will be briefly introduced in the remainder of the present section, and ICA will be presented as a method for source separation in Sect. 2.6.1.

The Discrete Fourier Transform (DFT)

The DFT is the most popular orthogonal discrete transformation with invariant bases. Its basis functions are complex exponentials of the form $b_k(t) = e^{j\frac{2\pi}{T}kt}$. The analysis equation (Eq. 2.19) then yields the following DFT coefficients (the usual notation for the DFT is $c_k = S(k)$):

$$S(k) = \sum_{t=0}^{T-1} s(t)e^{-j\frac{2\pi}{T}kt}, \quad k = 0, \dots, T-1. \quad (2.21)$$

The quantities $|S(k)|$ and $\angle S(k)$ constitute the magnitude and phase spectrum, respectively, of the signal. Recall that, since the DFT is a complete basis decomposition, $T = K$. The DFT can be efficiently computed by the *Fast Fourier Transform* (FFT) algorithm [121].

⁸Orthogonality implies $\langle \mathbf{b}_i, \mathbf{b}_j \rangle = \delta(i-j)$ or, in matrix form, $\mathbf{B}^{-1} = \mathbf{B}^H$.

2.3.2 Time–frequency decompositions

Spectral transforms like the DFT are *frequency-localized*: their basis functions have a definite position in frequency. The *frequency support* of the DFT, i.e., the set of positions in frequency of the basis functions, is given by $f_k = \frac{k}{T}f_s$, where $k = 1, \dots, K$ and f_s is the sampling rate, or by their normalized counterparts $\omega_k = \frac{2\pi}{T}k$. However, the *time support* $t = 0, \dots, T - 1$ is the same for all basis functions and furthermore equals the time support of the signal to be analyzed. This means that every time-localized event in the original signal (such as an impulse or a quick change in amplitude) will not be appropriately represented in the transformed domain, and its features will appear spread throughout the whole time support. The DFT can thus have unlimited frequency resolution (proportional to $T = K$), but no time resolution inside the considered signal excerpt.

If the analyzed signal is highly non-stationary (which is the case of speech and music signals), a certain time granularity is required to obtain a useful representation. This is especially important to fulfill the sparsity requirements of underdetermined source separation: higher time localization leads to higher time resolution and thus to higher temporal sparsity, since each meaningful temporal component of the signal will be represented by only one or few coefficients. The same applies for frequency localization.

As will be explained later, there is a trade-off relationship between time and frequency resolution. Music and general audio signals are not stationary and are thus both time- and frequency-localized to a certain degree. Thus, choosing an appropriate balance will determine the suitability of a certain signal representation for the separation task. This issue will be thoroughly addressed in Chapter 3.

A general time–frequency decomposition is a generalization of the basic additive model of Eq. 2.15 with a set of expansion functions both localized in time (index r) and frequency (index k):

$$s(t) = \sum_{r=1}^R \sum_{k=1}^K c_{rk} b_{rk}(t). \quad (2.22)$$

Two different notational conventions will be used to denote time–frequency representations, each one being more convenient within its corresponding context. In discussions where keeping the two-dimensional time–frequency meaning is important, such as in time–frequency masking (Sect. 2.7.3) and in spectral basis decompositions (Sect. 4.5.1), a time–frequency transformed signal will be denoted either by an element-wise notation with explicit indexing of the form $S(r, k) \forall r, k$, or by a time–frequency matrix $\mathbf{S}(r, k)$. To avoid confusion with the multi-source matrices of the mixing model $\mathbf{X} = \mathbf{A}\mathbf{S}$ (with time-domain signals as the rows), the time–frequency indices (r, k) will be explicitly indicated in the latter case, even if the whole bin matrix is denoted.

The other convention will be used in those cases where keeping the time–frequency ordering is not necessary, such as in measuring sparsity (Sect. 2.3.3) and in mixing matrix estimation by clustering (Sects. 2.6.2 and 3.4). In that case, all bins of the time–frequency representation corresponding to a single signal will be grouped into

a coefficient vector \mathbf{c} of concatenated representation frames, of size $(RK) \times 1$:

$$\mathbf{c} = (c_{11}, \dots, c_{1K}, c_{21}, \dots, c_{2K}, \dots, c_{R1}, \dots, c_{RK})^T. \quad (2.23)$$

The total number of coefficients will be denoted by $C = RK$. A vector concatenated in this way is said to follow a *lexicographic ordering*. When using multi-signal matrix explicit notation with time–frequency representations, coefficient matrix \mathbf{C} will have as its rows the lexicographically ordered coefficient vectors \mathbf{c}_n corresponding to the transformed sources.

The Short-Time Fourier Transform (STFT) and Gabor expansions

In practice, time resolution can be obtained by dividing the input signal into a sequence of analysis frames and performing a spectral analysis on each one of them. This corresponds to sequentially shifting the input signal $s(t)$ by steps of H samples (called the *hop size*), then multiplying the first chunk of size $L \geq H$ with an analysis window $w(t)$ of the same size, and finally computing a spectral transform of size $T \geq L$ from each frame. When the used spectral transform is the DFT, this procedure results in the *Short-Time Fourier Transform* (STFT) [121], given in frame r and frequency bin k by

$$S(r, k) = \sum_{t=0}^{L-1} s(rH + t)w(t)e^{-j\frac{2\pi}{T}kt}, \quad (2.24)$$

where $r = -\infty, \dots, +\infty$ and $k = 0, \dots, T - 1$ (again, $T = K$). The matrix $|S(r, k)|$ is called the *spectrogram* of the signal, and is the most widely used time–frequency representation.

Viewed from the time domain, as formulated by Eq. 2.22, the STFT amounts to a decomposition of the signal $s(t)$ into a set of expansion functions both localized in time and in frequency and weighted by the STFT coefficients $S(r, k)$ given by the previous equation:

$$s(t) = \sum_{r=-\infty}^{+\infty} \sum_{k=0}^{T-1} S(r, k)w(rH + t)e^{j\frac{2\pi}{T}kt}. \quad (2.25)$$

The expansion functions are now of the form $b_{rk}(t) = w(rH + t)e^{j\frac{2\pi}{T}kt}$. Time localization is provided by the finite-length “hopping window” $w(t)$, and frequency localization by the modulating complex sinusoid $e^{j\frac{2\pi}{T}kt}$. This formulation is also known as *Gabor expansion*, and in this context, the $b_{rk}(t)$ functions are called time–frequency *atoms*.

Uncertainty principle

Inherent to any kind of time–frequency decomposition is a trade-off between time and frequency resolution. Intuitively, long time windows are needed in order to resolve low frequency components, with longer periods. Inversely, short time windows offering better time resolution can only resolve frequency components whose periods

are shorter than the time interval they span. Denoting the time distance between two time frames as Δ_t and the normalized frequency distance between two frequency bins as Δ_ω , there is a lower bound [65] given by

$$\Delta_t \Delta_\omega \geq \sqrt{\frac{\pi}{2}}. \quad (2.26)$$

This is the *uncertainty principle* in signal analysis, and is one of the instances of Heisenberg's uncertainty principle originally formulated between position and momentum in the field of quantum physics. The area delimited by segments Δ_t and Δ_ω is referred to as a *time–frequency tile*.

2.3.3 Sparse decompositions

A signal approximated by the general additive model of Eq. 2.15 or its time–frequency counterpart (Eq. 2.22) is said to be sparse if most of its expansion coefficients c_k or c_{rk} are zero or close to zero. In the following, the coefficients of a given representation will be considered organized as the elements of a lexicographic coefficient vector \mathbf{c} (Eq. 2.23). Notation c_i will be used to denote a generic coefficient of the concatenated vector, and c without index will be used in those cases where it will be more useful to consider the coefficient vector as a random variable.

The reason why high sparsity of the source representations is desired in source separation problems is straightforward: the less coefficients are needed to adequately describe a particular source signal, and the less energetic they are, the less degree of overlapping will occur when mixed with other signals. The only exception will happen if the sources have exactly the same probability distribution, which is rather unlikely in a realistic situation.

An explicit sparsity assumption is not needed in simple BSS problems such as instantaneous and determined BSS. Its paradigmatic approach, ICA, does not rely on sparsity but on statistical independence of the sources, which is a weaker assumption (although closely related, as will be seen). Also, it is a method in the time domain, where general audio signals have a very low sparsity (see Sect. 3.2.1). However, it is crucial in most underdetermined situations. This is the truer the less a priori information is available, and the higher is the ratio between the number of sources and mixtures.

A sparse random variable will have a pronounced peak around the mean in its probability density function (pdf). The higher the sparsity, the sharper will be that peak. In order to characterize sparsity by means of the pdf, consider first the family of *generalized exponential power* distributions, expressed by

$$p(c) = \alpha e^{-\beta|c-\mu|^\nu}, \quad (2.27)$$

where μ is the mean and α is a scaling factor to ensure a pdf with unit area. The parameter ν determines the peakedness of the distribution, and β the width of the peak. For $\nu = 2$, and setting the appropriate α and β for unit area, a Gaussian (or normal) distribution is obtained:

$$p(c) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(c-\mu)^2}{2\sigma^2}}. \quad (2.28)$$

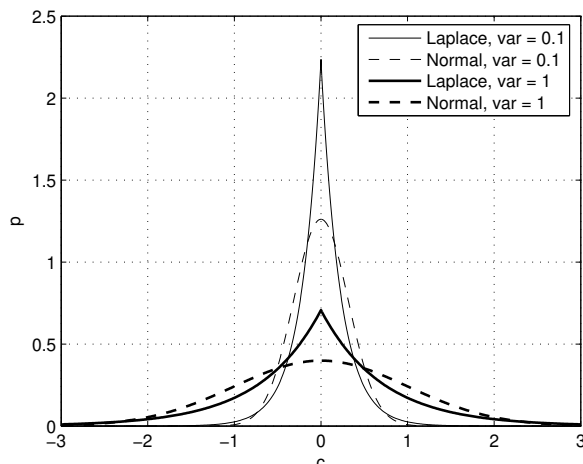


Figure 2.5: Comparison of Laplace and normal probability density functions for two different variances.

For $\nu = 1$, the result is a Laplacian distribution:

$$p(c) = \frac{\lambda}{2} e^{-\lambda|c-\mu|}. \quad (2.29)$$

The variance of the Laplace distribution is $2/\lambda^2$. The Laplacian distribution is especially important as a model for sparsity because of its simplicity. As will be seen, it constitutes the basic sparsity assumption in a wide range of separation algorithms. Figure 2.5 shows a comparison between a Gaussian distribution and a Laplace distribution with zero means and two different values for the variance. Thinner peaks correspond to most values being close to zero, and thus to higher sparsity. It can be seen that, for the same variance, the Laplace density is sparser.

Even higher sparsity is obtained with *impulse-type distributions*, which correspond to $0 < \nu < 1$. In general, exponential power distributions with $0 < \nu < 2$ are called *supergaussian* (sharper than Gaussian), and distributions with $\nu > 2$ are called *subgaussian* (flatter than Gaussian). The extreme subgaussian case is the uniform distribution, which is obtained for $\nu \rightarrow \infty$.

Measures of sparsity

The most common way of measuring the sparsity ξ of a signal representation is by means of the ℓ_p norm of its coefficient vector with the constraint $0 \leq p \leq 1$:

$$\xi = \|\mathbf{c}\|_p = \left(\sum_{i=1}^C |c_i|^p \right)^{1/p}, \quad 0 \leq p \leq 1. \quad (2.30)$$

The ℓ_0 norm gives the number of non-zero elements of \mathbf{c} :

$$\|\mathbf{c}\|_0 = \#\{i, c_i \neq 0\}, \quad (2.31)$$

where $\#\{\cdot\}$ denotes the counter operator. This norm is rarely used since it is highly sensible to noise: a slight addition of noise will make a representation completely nonsparse. Instead, a modified, threshold-based version can be used, called the ℓ_ϵ norm:

$$\|\mathbf{c}\|_\epsilon = \#\{i, |c_i| \geq \epsilon\}. \quad (2.32)$$

However, determining a reasonable noise threshold ϵ for unknown signals is a difficult task [84] and thus this measure also lacks robustness.

The most common norm of this family is the ℓ_1 norm, obtained for $p = 1$:

$$\|\mathbf{c}\|_1 = \sum_{i=1}^C |c_i|. \quad (2.33)$$

It arises naturally in many separation approaches that assume that the sources are Laplacian, as will be seen in Sect. 2.7.2. The ℓ_2 norm $\|\cdot\|$, for which the order index is usually omitted, corresponds to the traditional Euclidean norm, and to the square root of the energy.

In general, it is important to center and normalize according to variance before measuring sparsity. If coefficient vectors of different lengths are to be compared, the ℓ_p norms should also be normalized by the number of coefficients, in which case they will be denoted by $\bar{\ell}_p$. Also, it should be noted that such norms are in reality measures of *non-sparsity*: they are larger the more high-energy coefficients are present. They must be inverted ($-\ell_p$) in order to directly correspond to sparsity.

A more general family of sparsity measures is given by the expectation of non-quadratic functions $g(c)$, which in practical cases is approximated by the empirical average (see [79], p. 374):

$$\xi = E\{g(c)\} = \frac{1}{C} \sum_{i=1}^C g(c_i). \quad (2.34)$$

Choosing $g(x) = |x|$ results in the ℓ_1 norm normalized by the number of coefficients.

Another important particular case thereof (apart from scaling and additive factors) is the kurtosis, given for centered variables by

$$\kappa_4 = E\{c^4\} - 3[E\{c^2\}]^2 \quad (2.35)$$

and the normalized kurtosis

$$\bar{\kappa}_4 = \frac{E\{c^4\}}{[E\{c^2\}]^2} - 3. \quad (2.36)$$

If the coefficients are assumed to be whitened⁹, then $E\{c^2\} = 1$ and both definitions become equivalent:

$$\kappa_4 = E\{c^4\} - 3. \quad (2.37)$$

⁹A random variable is said to be white when it has zero mean (i.e., it is centered) and its covariance matrix is diagonal and of unit variances ($\mathbf{\Sigma} = \mathbf{I}$).

Kurtosis is a well-known measure of the nongaussianity of a distribution. It is positive for supergaussian densities, negative for subgaussian densities and equals zero for Gaussian densities. In this context, supergaussian densities are also called *leptokurtic* and subgaussian ones *platykurtic*. Exponential densities are the more supergaussian the more peaked around the mean, and thus kurtosis will grow with sparsity.

Several other, more sophisticated measures of sparsity have been devised in the literature, including logarithmic and hyperbolic functions, negentropy [79], entropy diversity measures and the Gini index [128], which was originally developed to measure the distribution of wealth in society. Several studies deal with the effect of using different sparsity measures for several optimization purposes [84, 128]. Karvanen and Cichoki show in [84] that in the case of noisy signals with non-symmetric or multimodal densities, different sparsity measures can lead to completely opposite optimization results.

Basic sparse decompositions

Simple signal transformations often result in an enormous gain in representation sparsity. Provided the transformation is linear and perfectly or near-perfectly invertible, the instantaneous mixing model of Eq. 2.5 is applicable, and separation algorithms can be performed in the transformed domain. As a basic example, a 1-second tone of T samples consisting of three static sinusoidal partials¹⁰ will need $T \gg 3$ coefficients for a perfect representation in the time domain, and only three (the amplitudes of the partials) for a perfect representation in the magnitude spectrum.

In the context of underdetermined source separation, the transformation most widely used to increase sparsity is the STFT. Further possibilities include multiresolution techniques such as the DWT and Wavelet Packets. Other signal transformations that are aimed at compact representations, such as dimensionality reduction techniques, lossy audio compression codecs, spectral front-ends like *Mel Frequency Cepstral Coefficients* (MFCC) or any other kind of feature extraction methods, are indeed highly sparse, but inappropriate to source separation since they lose lots of information and cannot be inverted for an accurate signal reconstruction.

Figure 2.6 shows a comparison of basic representations of a sound segment in the time domain, in the frequency domain (DFT magnitude spectrum) and in the time–frequency domain (spectrogram). The examples correspond to a melody fragment played by an alto saxophone. The normalized logarithmic histograms for the corresponding representation coefficients are shown to the right of each representation plot, and the corresponding normalized ℓ_ϵ , ℓ_1 and kurtosis sparsity measures are shown in Table 2.3. The DFT and the spectrogram representation are much sparser than the time-domain signal, as can be observed from the measures and from the peakedness of the respective histograms. As can be seen, the increase in sparsity

¹⁰*Partials* or *overtones* are the predominant sinusoidal components of a sound. It is a more general term than *harmonics*, which denotes the special case in which the frequencies of the partials are integer multiples of the fundamental frequency.

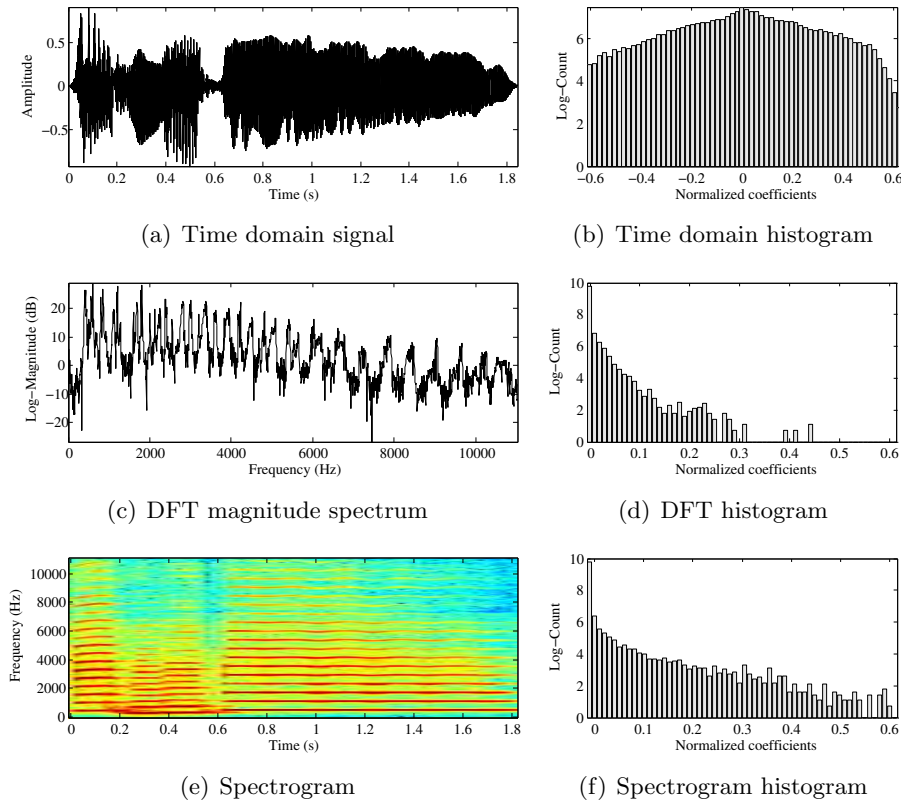


Figure 2.6: Example of sparsity properties of basic signal representations.

greatly varies between the different measures. The DFT was computed by taking the whole clip at a time: i.e., it is a completely frequency-localized representation. Even if in this case it objectively achieves a higher sparsity than the spectrogram time–frequency representation, it is not usable in practical applications, mainly because all methods require a certain time granularity, as provided by the time–frequency representations, and because of the high computational requirements of computing large DFTs. Note that both the DFT and the spectrogram magnitudes are plotted logarithmically, whereas linear magnitudes are used for separation and sparsity measurement.

In the present work, Chapter 3 will be devoted to the evaluation of the use of frequency-warped time–frequency representations as sparse decompositions for underdetermined BSS. A thorough experimental setup will be used to test the sparsity properties of frequency-warped sources and mixtures.

Overcomplete decompositions

As already introduced, a signal decomposition is said to be overcomplete if the dictionary of expansion functions is redundant, i.e., the $T \times K$ matrix \mathbf{B} in the ex-

Measure	$-\bar{\ell}_\epsilon$	$-\bar{\ell}_1$	$\bar{\kappa}_4$
Time	-0.667	-0.207	0.11
DFT	-0.009	-0.009	284.44
Spectrogram	-0.039	-0.025	53.41

Table 2.3: Sparsity measures corresponding to the signals in Fig. 2.6.

pansion model $\mathbf{s} = \mathbf{B}\mathbf{c}$ is unsquare with $T < K$. The representation problem is thus no longer invertible and the general analysis equation 2.18 is no longer applicable. If an overcomplete dictionary consists of time- and frequency-localized functions that represent a wide range of atomic audio events, the resulting representation can achieve high levels of sparsity. Given a dictionary and an input signal, the decomposition algorithm searches the bases according to a sparsity-related objective function. Commonly used dictionaries include Gabor atoms, damped sinusoids and wavelet packets.

Examples of overcomplete decomposition methods are *Basis Pursuit* [42], *Minimum Fuel Neural Networks* (MFNN) [178] and the method of frames [49]. In the context of audio time–frequency processing, the most popular however is the *Matching Pursuit* method, proposed by Mallat and Zhang [104]. Due to its sequential nature it is able to obtain highly refined and sparse results. Basically, it consists of projecting the input signal onto each of the dictionary atoms and to measure their correlation; the most correlated atom gets then subtracted from the signal and the steps are iterated under a stopping condition is met. Endelt and La Cour-Harbo [58] perform a comparison of different overcomplete decomposition methods for the purpose of musical signal representation. Matching Pursuit obtained the best overall performance in terms of sparsity measured by the ℓ_1 norm.

2.3.4 Principal Component Analysis

Principal Component Analysis (PCA) [82], also called *Karhunen-Loève Transform* (KLT), is a linear, orthogonal adaptive transform whose goal is to decorrelate a set of input random vectors. Geometrically, it finds the orthogonal directions of maximum variance in the data scatter plot. No assumptions about the probability distributions, or of generative models of the vectors are needed; it is enough to estimate the first and second order statistics from the input samples. The main application of PCA is dimensionality reduction. Since it plays an important role in the scope of the present work (as a method for compaction of the timbre models introduced in Chapter 4), it will be introduced here in detail.

The following explicit notation for the direct PCA transformation (analysis equation) will be used:

$$\mathbf{Y} = \mathbf{P}^T \mathbf{X} \quad (2.38)$$

and for the inverse transformation (reconstruction or synthesis equation):

$$\mathbf{X} = \mathbf{P}\mathbf{Y}. \quad (2.39)$$

\mathbf{X} and \mathbf{Y} are the $N \times L$ input and output data matrices, respectively, consisting of L vectors from an N -dimensional space. This can be interpreted as a set of L realizations (i.e., observed samples) of a set of N random variables, or of an N -dimensional random vector. For instance, in a time-domain context, N would be the number of signals and L the number of time samples. \mathbf{P} is the $N \times N$ orthogonal ($\mathbf{P}^{-1} = \mathbf{P}^T$) PCA matrix¹¹. Its columns (denoted by \mathbf{p}_i) are the PCA bases. The rows of \mathbf{Y} are the *principal components* of the data.

The motivation of PCA is to find a transformation that minimizes the reconstruction error when using less basis vectors than N . In this case, the result is an orthogonal projection from $M < N$ basis vectors (columns) out of the full $N \times N$ matrix, or, in matrix notation, a reduced $M \times N$ transformation matrix \mathbf{P}_ρ^T . For a single vector \mathbf{x} (one column of \mathbf{X}), and using instantaneous notation and Eq. 2.20, the following reconstructed vector is obtained¹²:

$$\hat{\mathbf{x}} = \sum_{i=1}^M \langle \mathbf{p}_i, \mathbf{x} \rangle \mathbf{p}_i = \sum_{i=1}^M (\mathbf{p}_i^T \mathbf{x}) \mathbf{p}_i \quad (2.40)$$

and using matrix notation for all L vectors:

$$\hat{\mathbf{X}} = \mathbf{P}_\rho \mathbf{Y}_\rho = \mathbf{P}_\rho \mathbf{P}_\rho^T \mathbf{X}. \quad (2.41)$$

In other words, we seek a reduced-dimension representation \mathbf{Y}_ρ (in a space spanned by a subset of M bases \mathbf{p}_i) of the input data \mathbf{X} so that the most information is retained. Note that the $M \times N$ matrix \mathbf{P}_ρ^T is no longer orthogonal, and thus $\mathbf{P}_\rho \mathbf{P}_\rho^T \neq \mathbf{I}$.

Thus, the reconstruction error or *residual* is to be minimized, which is given by

$$\boldsymbol{\epsilon} = \mathbf{X} - \hat{\mathbf{X}}. \quad (2.42)$$

Equivalently (and, as will be seen, more conveniently), the mean square magnitude thereof, which yields the *Mean Square Error* (MSE) criterion, can be subjected to minimization:

$$J_{MSE} = E\{\|\mathbf{x} - \hat{\mathbf{x}}\|^2\} = E\{(\mathbf{x} - \hat{\mathbf{x}})^T (\mathbf{x} - \hat{\mathbf{x}})\}. \quad (2.43)$$

Note that the expectation is performed along all L sample vectors.

Using the previous equations, assuming that the bases are orthonormal ($\mathbf{p}_i^T \mathbf{p}_i = 1$ and $\mathbf{p}_i^T \mathbf{p}_j = 0$ if $i \neq j$), and taking into account the linearity of the expectation operator, the following is obtained:

¹¹Note that, since only real coefficients are considered here, $\mathbf{P}^T = \mathbf{P}^H$.

¹²The hat notation ($\hat{}$) will be used in this work to denote *reconstruction* or *estimation*.

$$J_{MSE} = E \left\{ \left\| \sum_{i=1}^N (\mathbf{p}_i^T \mathbf{x}) \mathbf{p}_i - \sum_{i=1}^M (\mathbf{p}_i^T \mathbf{x}) \mathbf{p}_i \right\|^2 \right\} \quad (2.44)$$

$$= E \left\{ \left\| \sum_{i=M+1}^N (\mathbf{p}_i^T \mathbf{x}) \mathbf{p}_i \right\|^2 \right\} \quad (2.45)$$

$$= \sum_{i=M+1}^N E \{ (\mathbf{p}_i^T \mathbf{x})(\mathbf{x}^T \mathbf{p}_i) \} \quad (2.46)$$

$$= \sum_{i=M+1}^N \mathbf{p}_i^T E \{ \mathbf{x} \mathbf{x}^T \} \mathbf{p}_i = \sum_{i=M+1}^N \mathbf{p}_i^T \mathbf{R}_x \mathbf{p}_i \quad (2.47)$$

where $\mathbf{R}_x = E\{\mathbf{x}\mathbf{x}^T\} = \frac{1}{L-1}\mathbf{X}\mathbf{X}^T$ is the $N \times N$ correlation matrix of the $N \times L$ input data matrix¹³.

This expression can be minimized using the Lagrange method with an orthonormality constraint, which yields the Lagrangian function

$$\mathcal{L}(\mathbf{p}_1, \dots, \mathbf{p}_N, \lambda_1, \dots, \lambda_N) = \sum_{i=M+1}^N \mathbf{p}_i^T \mathbf{R}_x \mathbf{p}_i + \sum_{i=M+1}^N \lambda_i (1 - \mathbf{p}_i^T \mathbf{p}_i). \quad (2.48)$$

Setting the derivative to zero finally yields the relationship

$$\mathbf{R}_x \mathbf{p}_i = \lambda_i \mathbf{p}_i, \quad (2.49)$$

which shows that the searched bases \mathbf{p}_i are the eigenvectors of the correlation matrix, giving a closed solution for PCA. Furthermore, if this expression is replaced into the criterion definition, the result is

$$J_{MSE} = \sum_{i=M+1}^N \mathbf{p}_i^T \lambda_i \mathbf{p}_i = \sum_{i=M+1}^N \lambda_i. \quad (2.50)$$

That is, in order to minimize the error, the $N - M$ smallest eigenvalues λ_i must be left out. In other words, the M eigenvectors \mathbf{p}_i corresponding to the M largest eigenvalues must be retained in the final, reduced transformation matrix.

The eigenvector equation 2.49 can be rewritten as

$$\mathbf{R}_x \mathbf{P} = \mathbf{P} \mathbf{\Lambda}, \quad (2.51)$$

where $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues: $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N)$. In this case, \mathbf{P} is invertible and $\mathbf{P}^{-1} = \mathbf{P}^T$, so it is possible to write

$$\mathbf{R}_x = \mathbf{P} \mathbf{\Lambda} \mathbf{P}^T. \quad (2.52)$$

¹³ $E\{\mathbf{x}\mathbf{x}^T\} = \frac{1}{L-1}\mathbf{X}\mathbf{X}^T$ is the unbiased estimator of the correlation matrix. The biased estimator is $E\{\mathbf{x}\mathbf{x}^T\} = \frac{1}{L}\mathbf{X}\mathbf{X}^T$. In this work, unbiased estimators will be used.

This is the standard formulation of the eigenvalue problem for square matrices, called *Eigenvalue Decomposition* (EVD). It is solved using numerical algorithms, one of the most popular being the *QR algorithm*. Note that since \mathbf{R}_x is a symmetric matrix, its eigenvectors are mutually orthogonal, which is coherent with the orthonormality constraint imposed to the Lagrange solution to PCA.

PCA can alternatively be computed by means of *Singular Value Decomposition* (SVD). It is easy to show that an EVD of the correlation matrix corresponds to performing an SVD on the input data \mathbf{X} and taking the left singular vectors and the square of the singular values as the eigenvectors and eigenvalues, respectively. The advantages of using SVD instead of EVD is that it is a more reliable and precise algorithm, and that it avoids computing the correlation matrix of the data.

There exists a slightly different definition of PCA, which approximates a projection of the form

$$\hat{\mathbf{x}} = \sum_{i=1}^M \langle \mathbf{p}_i, \mathbf{x} \rangle \mathbf{p}_i + \sum_{i=M+1}^N c_i \mathbf{p}_i. \quad (2.53)$$

That is, instead of the truncated projection of Eq. 2.40, where the last $N - M$ basis vectors are ignored, they are fixed with some constant scalars c_i . An analogous derivation leads to the PCA bases being given by the unit-length eigenvectors of the covariance matrix

$$\mathbf{\Sigma}_x = E\{(\mathbf{x} - E\{\mathbf{x}\})(\mathbf{x} - E\{\mathbf{x}\})^T\}, \quad (2.54)$$

instead of the correlation matrix \mathbf{R}_x . However, both definitions are equivalent if the input data has zero mean. To avoid confusion regarding which definition is used, the input data is usually first centered as a previous step before PCA:

$$\mathbf{X}_c = \mathbf{X} - E\{\mathbf{X}\}. \quad (2.55)$$

In the following, it will be always assumed that the input data has been centered.

The derivation using the MSE criterion is just one possibility to obtain the PCA bases. The same result is obtained if the criterion is to retain the largest variance in the kept dimensions. In this context, it can be shown that the variance of the i -th principal component equals the eigenvalue λ_i corresponding to the (normalized) eigenvector \mathbf{p}_i :

$$\sigma_{Y_i}^2 = E\{y_i^2\} = \lambda_i. \quad (2.56)$$

Also, from Eq. 2.52,

$$\mathbf{R}_y = \mathbf{P}^T \mathbf{R}_x \mathbf{P} = \mathbf{\Lambda}. \quad (2.57)$$

Thus, the data yielded by PCA is fully uncorrelated ($E\{y_i y_j\} = 0, i \neq j$).

Whitening

The purpose of *whitening* is to decorrelate a centered random vector \mathbf{x} and to make the variances of their elements equal to unity, by transforming it into a random vector \mathbf{z} that fulfills $\mathbf{\Sigma}_z = \mathbf{I}$. Whitening can be performed using PCA as the decorrelation stage, followed by eigenvalue scaling to normalize the variances:

$$\mathbf{Z} = \mathbf{\Lambda}^{-1/2} \mathbf{P}^T \mathbf{X} = \mathbf{V} \mathbf{X}, \quad (2.58)$$

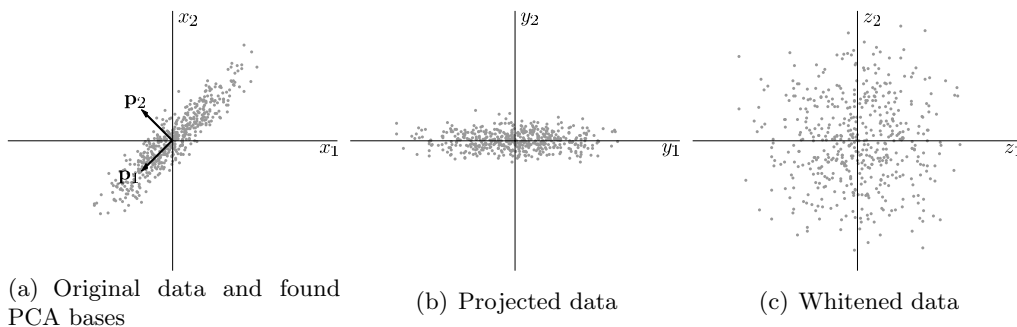


Figure 2.7: Example of PCA and whitening of a bivariate normal distribution.

where $\mathbf{V} = \mathbf{\Lambda}^{-1/2}\mathbf{P}^T$ is the whitening matrix. Note that the square root is in this case an element-wise operation.

Figure 2.7 shows the effect of applying the successive stages of decorrelation by PCA and whitening to a two-dimensional, normally distributed random vector. The figure depicts the *scatter plots* of the joint distribution after each transformation step. Note that PCA, as an orthogonal transformation, corresponds geometrically to a rotation of the axes. Whitening, however, is not an orthogonal operation.

2.4 Analogy between signal decomposition and source separation

As introduced in the previous sections, both source separation from linear mixtures and linear signal decompositions can be formulated as linear transformations. The notation $\mathbf{X} = \mathbf{A}\mathbf{S}$ has been used for linear source separation, $\mathbf{S} = \mathbf{C}\mathbf{B}^T$ for general basis expansions and $\mathbf{X} = \mathbf{P}\mathbf{Y}$ for PCA. In fact, separation of a linear mixture can be viewed as a basis decomposition problem in which the columns of the mixing matrix \mathbf{a}_n are the bases and the sources \mathbf{s} are the transformed coefficients. An underdetermined mixing matrix \mathbf{A} of size $M \times N$ ($M < N$) corresponds thus to an overcomplete expansion matrix \mathbf{B} of size $T \times K$ ($T < K$).

There are many examples in the literature that offer such a dual point of view. For instance, ICA has been both used as a BSS method [79] and as a basis decomposition method aimed at feature extraction (see [16] and Chapter 21 in [79]). Also, it has been shown that, under a sparsity criterion, underdetermined source separation and adaptive basis decomposition are equivalent problems [98]. As an example, the results in [58] show that the bases obtained by adaptive decomposition methods such as Basis Pursuit closely correspond to the separated sources building the sound mixture.

There are however some conceptual differences that should be taken into account. In the instantaneous mixing model, as well as in the PCA formulation, vector \mathbf{x} denotes a single observation of an M -dimensional random vector. In contrast, vector \mathbf{s} in the decomposition model denotes T observations of a single random variable.

In the time-domain case, T is the number of time samples, which will most certainly fulfill $T \gg M$, $T \gg N$ and $T \leq K$. As a consequence, time will be conveyed in the decomposition basis matrix \mathbf{B} , giving rise to time-domain expansion functions as its columns. In contrast, the bases in \mathbf{A} and \mathbf{P} do not represent time-domain functions, and are best interpreted as the directions of the new coordinates in the transformed space.

It is also possible to combine both signal decomposition and source separation as a sequence of processes into a common framework [182]. This gives rise to separation systems that work in the transformed domain. In such methods, a set of signals is first transformed into a sparse representation, for which the multi-signal notation $\mathbf{S} = \mathbf{C}\mathbf{B}^T$ must be used. Using the linear model $\mathbf{X} = \mathbf{A}\mathbf{S}$, the separation problem turns into a joint separation/decomposition problem formulated by

$$\mathbf{X} = \mathbf{A}\mathbf{C}\mathbf{B}^T. \quad (2.59)$$

If the mixture is subjected to linear decomposition using the same basis as the sources:

$$\mathbf{X} = \mathbf{Y}\mathbf{B}^T, \quad (2.60)$$

then it is possible to ignore the basis altogether and perform separation only in the transformed domain, by solving the transformed linear mixing model

$$\mathbf{Y} = \mathbf{A}\mathbf{C}. \quad (2.61)$$

Under these conditions, any equation involving original-domain matrices \mathbf{X} and \mathbf{S} will be equally valid replacing them with their transformed-domain counterparts \mathbf{Y} and \mathbf{C} . All approaches considered in the present work are of this type, and separation in the transformed domain will be the central topic throughout Chapter 3.

2.5 Joint and staged source separation

In Sect. 2.1.1, instantaneous source separation was formulated as a solution to the system of linear equations $\mathbf{X} = \mathbf{A}\mathbf{S}$, where both the variables (sources) and the coefficients (mixing matrix) are unknown. This and the next two sections introduce how this can be solved. It is worth emphasizing that all following derivations are equally valid in the transformed domain by substituting \mathbf{X} and \mathbf{S} by \mathbf{Y} and \mathbf{C} , respectively.

The most general way to approach a solution is to define a minimization problem based on some cost function of the error $\mathbf{X} - \mathbf{A}\mathbf{S}$. For example, using the MSE and instantaneous notation, this yields the optimization problem

$$\min_{\mathbf{A}, \mathbf{s}} E\{\|\mathbf{x} - \mathbf{A}\mathbf{s}\|^2\}. \quad (2.62)$$

The equivalent using explicit notation is the minimization of the Frobenius norm of the error matrix:

$$\min_{\mathbf{A}, \mathbf{S}} \|\mathbf{X} - \mathbf{A}\mathbf{S}\|_F^2. \quad (2.63)$$

The Frobenius norm is given by

$$\|\mathbf{X}\|_F = \sqrt{\sum_{i,j} x_{ij}^2}. \quad (2.64)$$

Approaches formulated in this way are termed *joint source separation* methods, since they estimate both unknown quantities, \mathbf{S} and \mathbf{A} , at the same time.

The general formulation of Eq. 2.63 has infinitely many solutions given by matrices $\mathbf{U}\mathbf{S}$ and $\mathbf{A}\mathbf{U}^{-1}$ for any invertible matrix \mathbf{U} . Thus, the problem needs to be further constrained. One possible way to do it is to assume certain probability distributions for the variables involved, and to tackle the problem from a probabilistic point of view [182]. In a Bayesian context a *Maximum A Posteriori* (MAP) formulation can be applied, aimed at maximizing the posterior probability $P(\mathbf{A}, \mathbf{S}|\mathbf{X})$. According to Bayes' theorem, and assuming that \mathbf{A} and \mathbf{S} are statistically independent ($P(\mathbf{A}, \mathbf{S}) = P(\mathbf{A})P(\mathbf{S})$) this posterior is given by

$$P(\mathbf{A}, \mathbf{S}|\mathbf{X}) = \frac{P(\mathbf{X}|\mathbf{A}, \mathbf{S})P(\mathbf{A})P(\mathbf{S})}{P(\mathbf{X})} \propto P(\mathbf{X}|\mathbf{A}, \mathbf{S})P(\mathbf{A})P(\mathbf{S}). \quad (2.65)$$

If \mathbf{A} is assumed to be uniformly distributed (i.e., all mixing weights are equally probable), then $P(\mathbf{A})$ will not have an influence on the optimization, and thus the problem reduces to

$$\max_{\mathbf{A}, \mathbf{S}} P(\mathbf{A}, \mathbf{S}|\mathbf{X}) \propto \max_{\mathbf{A}, \mathbf{S}} P(\mathbf{X}|\mathbf{A}, \mathbf{S})P(\mathbf{S}). \quad (2.66)$$

It will be now assumed that the sources and their samples are statistically independent, and thus the joint prior $P(\mathbf{S})$ is factorial:

$$P(\mathbf{S}) = \prod_{n,t} p_n(s_n(t)), \quad (2.67)$$

where p_n is the pdf of source $s_n(t)$. To compute the remaining likelihood factor $P(\mathbf{X}|\mathbf{A}, \mathbf{S})$ it is convenient to consider the noisy linear model (Eq. 2.12) with white Gaussian noise of covariance $\sigma^2\mathbf{I}$. In that case, since \mathbf{A} and \mathbf{S} are considered fixed for the likelihood, the only source of uncertainty is the noise. The probability is then given by the Gaussian distribution of the noise matrix $\mathbf{N} = \mathbf{X} - \mathbf{A}\mathbf{S}$. Assuming again statistical independence, the following is obtained:

$$P(\mathbf{X}|\mathbf{A}, \mathbf{S}) \propto \prod_{m,t} \exp\left(-\frac{(x_m(t) - (\mathbf{A}\mathbf{S})_{mt})^2}{2\sigma^2}\right), \quad (2.68)$$

where $(\mathbf{A}\mathbf{S})_{mt} = \sum_{n=1}^N a_{mn}s_n(t)$. Substituting Eqs. 2.68 and 2.67 into 2.66, taking the logarithm, and inverting the sign, the following MAP cost function is finally obtained:

$$\min_{\mathbf{A}, \mathbf{S}} \left\{ \frac{1}{2\sigma^2} \|\mathbf{X} - \mathbf{A}\mathbf{S}\|_F^2 - \sum_{n,t} l_n(s_n(t)) \right\}, \quad (2.69)$$

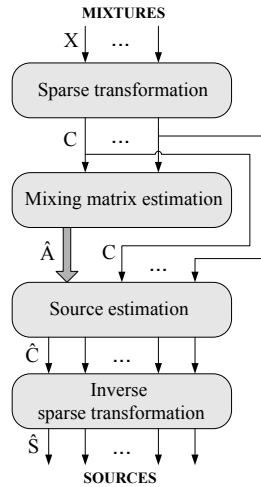


Figure 2.8: Diagram of a general staged Blind Source Separation system.

where l_n are the log-densities (logarithm of the probability densities) of the sources. The first term on the cost function is the reconstruction error. The second term is a penalty for sparsity.

The advantage of such a joint MAP formulation to BSS is its generality: it is valid for any number of sources and mixtures, and thus it has been used in underdetermined separation [182] or overcomplete decomposition scenarios [119]. However, it is extremely computationally demanding and unstable with respect to convergence [182]. For this reason, rather than a joint optimization, most separation methods follow a staged approach in which the mixing matrix and the sources are estimated successively in two separate steps. Such approaches will be referred to as *staged source separation* methods. As will be seen, they offer great algorithmic simplification, and flexibility in the system design, since the approaches for mixing matrix and source estimation can be freely combined. The staged approach has been described under a formalized framework by Theis and Lang [153], where the mixing matrix estimation stage was called *Blind Mixing Model Recovery* (BMMR) and source estimation was called *Blind Source Recovery* (BSR).

Figure 2.8 shows a block diagram of a general staged source separation system, with additional transformation blocks so that the separation is performed in a sparse domain, and indicating the relevant variables. The following two sections will provide an overview of methods that have been proposed to solve the separate tasks of mixing matrix estimation (Sect. 2.6) and source estimation or resynthesis (Sect. 2.7). All methods reviewed in the following sections and chapters follow the linear mixing model, unless otherwise noted. It is out of the scope of the present work to consider delayed and convolutive methods. The interested reader is referred to [79, 117, 164] for overviews covering that type of approaches. Also, note that all methods presented here are fully blind and solely relying on spatial information. Semi-blind methods or approaches based on a higher degree of a priori knowledge about the sources, such

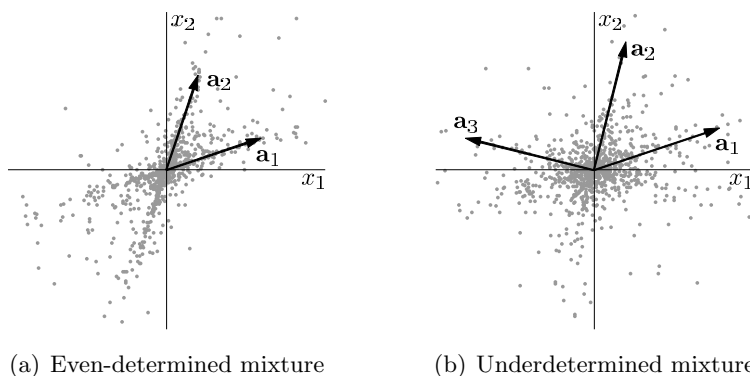


Figure 2.9: Scatter plot in two-channel mixture space for statistically independent and sparse sources.

as methods using advanced signal models or previous training, will be introduced in Sect. 5.1 in the context of monaural separation and in Sect. 6.1 in the context of stereo and multichannel separation.

2.6 Estimation of the mixing matrix

The first step in a staged BSS method is to estimate the mixing matrix \mathbf{A} from the mixture matrix \mathbf{X} . Under certain circumstances, it is possible to graphically illustrate the general ideas and the goals of the process. The M signals in the mixture matrix (its rows) can be considered the dimensions of an M -dimensional random vector. Their empirical joint distribution can be thus represented by means of a scatter plot, as has been done in Sect. 2.3.4 to illustrate PCA. Each point on the scatter plot lies on the position defined by the value proportion of that particular signal sample or coefficient between the mixture channels.

In the context of source separation, the scatter plot of \mathbf{X} corresponds to the *mixture space*, which in general will contain complex elements. Denoting the columns of the mixing matrix by \mathbf{a}_n , it is possible to rewrite the linear mixing model $\mathbf{x} = \mathbf{A}\mathbf{s}$ as

$$\mathbf{x} = \sum_{n=1}^N \mathbf{a}_n s_n. \quad (2.70)$$

This equation is valid for each sample $s_n(t)$ of the time-domain signals or for each coefficient c_i of the transformed signals. It becomes apparent that, if each mixture sample or coefficient is contributed only by one source (i.e., $s_n \neq 0$ and $s_j = 0$ for all $j \neq n$), the point \mathbf{x} will lie on the direction defined by vector \mathbf{a}_n in the complex mixture space. In a more realistic scenario, each mixture channel contains contributions from all sources, and thus the points corresponding to each source will deviate from the direction \mathbf{a}_n . However, if the sources are sufficiently sparse, and assumed to be statistically independent, the scatter plot will show the points corresponding to

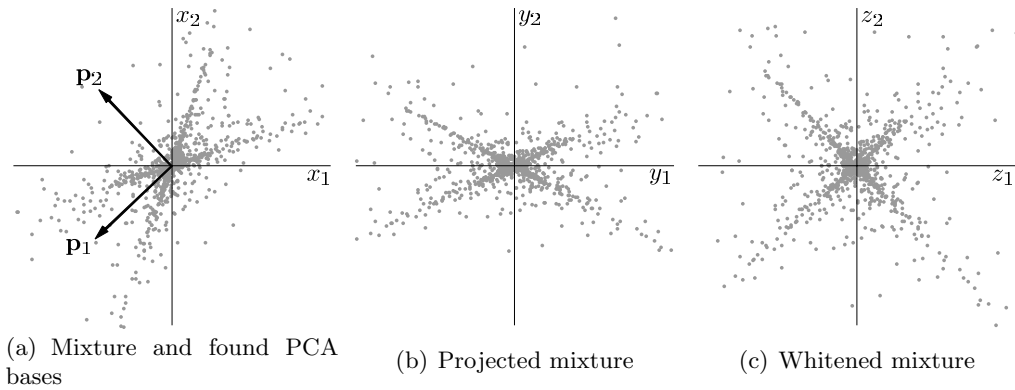


Figure 2.10: Application of PCA to an even-determined mixture of two statistically independent and sparse sources.

a particular source concentrating around its direction. Thus, the mixing directions correspond to the columns of the mixing matrix. This direction clustering phenomenon will be the more clear the higher is the sparsity of the sources. Although many algorithms do not strictly require the clustering to be visually perceptible, it can help to understand how they work for simple mixing setups. For convenience, unit-length mixing directions will always be assumed: $\|\mathbf{a}_n\| = 1$.

As an example, Fig. 2.9 shows two scatter plots in mixture space corresponding to an even-determined and an underdetermined mixture of independent and identically distributed (i.i.d.) sources generated from impulse-type distributions ($\nu = 0.6$ in Eq. 2.27). The vectors corresponding to the columns of the mixing matrix are superimposed on the scatter plots. It can be seen that, with the same sparsity, increasing the number of sources decreases the clustering effect, and thus mixing matrix estimation becomes more difficult.

As a consequence, from a geometrical point of view, the goal of a mixing matrix estimation algorithm is to find the mixing directions \mathbf{a}_n from the mixture scatter plot. Can this be achieved by PCA or whitening? A simple example will serve as illustration. Consider again the two-channel mixture of Fig. 2.9(a). When subjected to PCA, its directions of maximum variance are found (Fig. 2.10(a)). However, it can be seen in the figure that they do not correspond to the mixing directions. This would only be the case if the mixing matrix happened to be orthogonal. Thus, uncorrelation alone is not enough for source separation. This demonstrates that not only covariance-related (second order) statistics, but higher-order statistics need to be exploited, and that assumptions stronger than uncorrelation need to be adopted. These are the motivations underlying *Independent Component Analysis* (ICA).

2.6.1 Independent Component Analysis

Returning to Fig. 2.10, it is possible to observe an interesting effect that whitening had on the scatter plot: it has turned the mixing directions orthogonal (Fig. 2.10(c)).

Analytically, the application of PCA and whitening (Eq. 2.58) to a mixture gives

$$\mathbf{Z} = \mathbf{V}\mathbf{X} = \mathbf{V}\mathbf{A}\mathbf{S}. \quad (2.71)$$

Since the sources are i.i.d., they are also uncorrelated and white, and thus $E\{\mathbf{s}\mathbf{s}^T\} = \frac{1}{T-1}\mathbf{S}\mathbf{S}^T = \mathbf{I}$. Therefore,

$$E\{\mathbf{z}\mathbf{z}^T\} = \mathbf{V}\mathbf{A}E\{\mathbf{s}\mathbf{s}^T\}(\mathbf{V}\mathbf{A})^T = \mathbf{V}\mathbf{A}(\mathbf{V}\mathbf{A})^T = \mathbf{I}, \quad (2.72)$$

which shows that the “modified mixing matrix” $\mathbf{V}\mathbf{A}$ is orthogonal. Thus, to find the mixing directions, a further orthogonal transformation (a rotation) can be applied to the whitened data. The goal of ICA [79] is to find that additional rotation. Graphically, statistical independence corresponds to the signal data clouds being aligned with the scatter plot axes. A successful application of ICA to Fig. 2.9(a) will give the depicted mixing vectors \mathbf{a}_n . Figure 2.9(b) shows however that whitening and rotation will not be helpful in finding the 3 sources, and thus it is possible to anticipate that ICA will not be applicable to underdetermined mixtures. Although not mandatory for all ICA algorithms, pre-processing the data by PCA and whitening greatly simplifies the analysis due to the created orthogonality.

It is important to emphasize at this point that independence is a stronger requirement than uncorrelation. Two uncorrelated random variables x and y fulfill $E\{xy\} = E\{x\}E\{y\}$, whereas two independent variables fulfill

$$E\{g(x)h(y)\} = E\{g(x)\}E\{h(y)\} \quad (2.73)$$

for any absolutely integrable functions $g(\cdot)$ and $h(\cdot)$. Uncorrelation is thus a special case of independence for variables transformed by linear functions. Independence holds for any kind of linear or nonlinear functions. The property of Eq. 2.73 allows regarding independence as *nonlinear uncorrelation*.

The mentioned rotation to align the axes can thus be expressed as an orthogonal transformation

$$\mathbf{Y} = \mathbf{W}\mathbf{Z} \quad (2.74)$$

that maximizes the statistical independence of the variables contained in \mathbf{Y} , which are then called *independent components*. ICA employs numerical optimization algorithms to search for a matrix \mathbf{W} that maximizes a criterion function objectively measuring the degree of independence. Then, $\mathbf{Y} \approx \mathbf{S}$ and since $\mathbf{Y} = \mathbf{W}\mathbf{V}\mathbf{A}\mathbf{S}$, the estimated mixing matrix will be given by $\hat{\mathbf{A}} = (\mathbf{W}\mathbf{V})^{-1}$. If the mixture is assumed to be whitened beforehand, then $\mathbf{V} = \mathbf{I}$ and $\hat{\mathbf{A}} = \mathbf{W}^{-1} = \mathbf{W}^T$.

An important requirement in order for ICA to work is that the sources must be *nongaussian*, i.e., either supergaussian or subgaussian, as are any of the sparse distributions introduced in Sect. 2.3.3. The graphical explanation for this is that a whitened (and thus orthogonal) mixture of Gaussian sources would correspond to a hypersphere in mixture space (such as in Fig. 2.7(c)). Then, no definite mixture directions would be identifiable, and an optimization would fail to converge.

This can also be shown analytically. The multivariate Gaussian density is given by

$$p_{\mathbf{s}}(\mathbf{s}) = \frac{1}{(2\pi)^{n/2}(\det \mathbf{\Sigma})^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{s} - \boldsymbol{\mu})^T \mathbf{\Sigma}^{-1}(\mathbf{s} - \boldsymbol{\mu})\right), \quad (2.75)$$

where $\mathbf{\Sigma}$ is the covariance matrix, $\boldsymbol{\mu}$ is the mean vector and n is the dimensionality. Assume now that the sources are centered ($\boldsymbol{\mu} = 0$) and i.i.d. ($\mathbf{\Sigma} = \mathbf{I}$), reducing the above to

$$p_{\mathbf{s}}(\mathbf{s}) = \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{\|\mathbf{s}\|^2}{2}\right). \quad (2.76)$$

In general, the density of a random variable transformed by the linear and invertible transform matrix \mathbf{A} is given by

$$p_{\mathbf{x}}(\mathbf{x}) = \frac{1}{|\det \mathbf{A}|} p_{\mathbf{s}}(\mathbf{A}^{-1}\mathbf{x}). \quad (2.77)$$

Applying this to the previous equation, and noting that due to the orthogonality $\mathbf{A}^{-1} = \mathbf{A}^T$, $|\det \mathbf{A}| = 1$ and $\|\mathbf{A}^T \mathbf{x}\|^2 = \|\mathbf{x}\|^2$, the following joint distribution for the mixture is obtained:

$$p_{\mathbf{x}}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} |\det \mathbf{A}|} \exp\left(-\frac{\|\mathbf{A}^T \mathbf{x}\|^2}{2}\right) = \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2}\right). \quad (2.78)$$

It can be seen that the distributions of the sources and of the mixture are identical. In other words, no information about the mixing is conveyed by the mixture, and thus its identification will be impossible. Consistent with this is the fact that uncorrelated Gaussian variables are also independent, which implies that they can already be fully described by second-order statistics.

ICA methods are characterized by the strategy they employ to objectively define statistical independence, and by the used optimization algorithm. One important family of algorithms relies on the *central limit theorem*, which states that the sum of i.i.d. random variables tends towards a Gaussian distribution. This can be applied to each row of the ICA model of Eq. 2.74 because it is a linear combination of the sources of the form $y_n = \mathbf{b}^T \mathbf{A} \mathbf{s}$. Therefore, a component y_n will be more Gaussian than any of the sources s_n , unless it equals one of them. In other words, maximizing nongaussianity allows obtaining $\mathbf{Y} \approx \mathbf{S}$. As has been introduced in Sect. 2.3.3, kurtosis is an appropriate measure of nongaussianity. Sometimes, negentropy is used instead of kurtosis as a nongaussianity measure, because of the sensitivity of the latter to outliers. Both of them can be used as an objective function measured on \mathbf{Y} in combination with an optimization algorithm such as a gradient descent method, to perform ICA. A popular algorithm of this class is FastICA [78], which uses an optimized fixed-point search based on either kurtosis or negentropy. Note that the source separation/sparse decomposition analogy arises again in this context: ICA separates maximizing independence as measured by nongaussianity which, in turn, can be used as a measure of sparsity. In fact, ICA can be used for sparse coding applications (see [79], p. 396).

Other methods were motivated by principles of information theory and consider mutual information as a measure of statistical dependence. It has been shown that minimizing the mutual information of the y_n amounts to maximizing the sum of their nongaussianities, and thus leads to exactly the same criteria and algorithms as above. Information-theoretic optimization was one of the first ICA methods proposed [46].

Another possible formulation of ICA arises in a probabilistic context by considering the probability distribution of the mixture given in Eq. 2.77 as a likelihood function $p(\mathbf{X}|\mathbf{W})$ with $\mathbf{W} = \mathbf{A}^{-1}$ as the parameters, and performing a *Maximum Likelihood* (ML) estimation. Proposed ML methods include the *Bell-Sejnowski* (BS) algorithm [15] and the *natural gradient* algorithm [3]. FastICA can be also adapted to a ML context. ICA by ML is equivalent to maximizing the output entropy of a neural network, which is referred to as the Infomax principle [15].

Worth mentioning is finally the group of tensorial methods, which are formulated as a generalization of decorrelation for higher-order statistics. Cumulant tensors can be considered as high-order generalizations of the covariance matrix. High-order decorrelation can be obtained by making the cumulants zero, in much the same way as forcing zero covariances leads to whitening. A basic method of this kind is *Fourth-Order Blind Identification* (FOBI) [35]. A generalization thereof, called *Joint Approximate Diagonalization of Eigenmatrices* (JADE) [36] is probably the most widely used tensorial ICA method.

2.6.2 Clustering methods

Clustering methods rely on a direct geometrical analysis of the scatter plot to detect the mixing directions. Thus, in contrast to ICA, a strong sparsity is required, the stronger the higher is the sources/mixtures ratio (see Fig. 2.9). Thus, it is crucial in order for most of these methods to be applicable, that the mixture has been transformed into a sparse domain beforehand, usually using Fourier or Wavelet transforms.

A possible way to apply clustering on the data points in mixture space is to project them onto a hypersphere and use a standard clustering algorithm on the projection. The centers of the found clusters will then correspond to the mixing directions. Only half of the hypersphere must be considered, since each mixing direction actually corresponds to two clusters, one at each side of the mixture space origin. This approach was used by Zibulevsky *et al.* [182] based on *fuzzy C-means* clustering.

Rather than projecting onto a (hyper-)sphere, the Hard-LOST (*Line Orientation Separation Technique*) algorithm presented by O’Grady and Pearlmutter [115] searches for the directions by means of a modified *k-means* clustering algorithm in which cluster centres and distances to cluster centres have been replaced by line orientations and distances to lines. The term “hard” on the algorithm’s name refers to the hard assignment of data points to each cluster, meaning that each point is univocally associated to one cluster, such as in traditional k-means methods. In contrast, “soft” assignment refers to describing the degree of cluster membership of each point in a fractional way, such that each point has a list of probabilities of belonging to each one of the clusters. Within this context, a Soft-LOST version of the above mentioned method, based on *Expectation-Maximization* (EM), was introduced in [116].

Other methods employ density-based clustering, which consists in estimating the underlying probability density in mixture space and locating the directions at the

peaks of the density function. They are usually based on *kernel density estimation*, also called *Parzen window estimation*, which allows obtaining an estimated density without making any a priori probabilistic assumptions. Kernel estimation defines the estimated density as a sum of local kernel functions assigned to each one of the C data points x_i :

$$\hat{p}(x) = \frac{1}{hC} \sum_{i=1}^C K\left(\frac{x - x_i}{h}\right), \quad (2.79)$$

where $K(\cdot)$ is some *kernel* or *potential function* and h is a smoothing parameter. Parzen windowing has been applied to mixing matrix estimation by Erdoğmuş *et al.* [59]. Another method of this type is the one presented by Bofill and Zibulevsky in [23] and [24], which uses a weighted triangular function as the kernel. This is the method of choice here for estimating the mixing matrix in the experiments involving stereo mixtures. Thus, in Sect. 3.4 it will be addressed in more detail.

A related approach by van Hulle [155] applies density-based clustering not directly on the data points, but on a *topographic map* trained from the data via a *kernel-based Maximum Entropy learning Rule* (kMER). Gaussian kernels are then centered at the weights of the map, and added to estimate the density. A particularity of this approach is that, for speech signals, it is capable of working in the time domain, due to its considerable degree of time sparsity (see also Sect. 3.2.1).

2.6.3 Other methods

The ADReSS (Azimuth Discrimination and Resynthesis) system proposed by Barry *et al.* [13] is intended for linear stereo mixtures and exploits the fact that the phase of the sources remains unchanged when scaled for stereo location applying IID. Assuming a source has the same level on both channels, its contribution will disappear from the difference between the channels due to phase cancellation. A linear range of scaling factors is applied to each DFT frame of one channel before being subtracted from the other channel until a cancelling out of amplitudes reveals the correct scaling factor, and thus the corresponding source position. An efficient implementation of this algorithm was presented in [48].

The application of image processing methods to the scatter plot has been proposed by Lin *et al.* [99]. In that work, the data bins are first subjected to edge detection by selecting the regions with the highest density of data points, according to a given threshold. This will ideally yield a set of lines crossing at the origin and approximately corresponding to the mixing directions. The edge image is then subjected to a Hough transform, which is a classical image feature extraction technique used to detect straight lines. The peaks of the Hough transform correspond to the most data-populated line directions, and thus to the columns of the mixing matrix.

2.7 Estimation of the sources

In the even-determined case, and if each source is at a different position, the mixing matrix is square and invertible. Thus, once the mixing matrix \mathbf{A} has been estimated

as $\hat{\mathbf{A}}$, the estimated sources can be readily obtained by

$$\hat{\mathbf{S}} = \hat{\mathbf{A}}^{-1} \mathbf{X}. \quad (2.80)$$

This means that no real source estimation stage is needed, and that in this case, source separation amounts to identification of the mixing matrix. This is for instance the case of ICA, which is a mixing-matrix-estimation-only approach to BSS.

In over- and underdetermined situations, the mixing matrix is rectangular of size $M \times N$ and thus not invertible. An overdetermined situation ($M > N$, more mixtures/equations than sources/unknowns) can be reduced to an even-determined one just by ignoring some mixtures or by applying dimensionality reduction techniques such as PCA [153]. The underdetermined case ($M < N$) is however not trivial since the equation system is ill-posed and infinite solutions are possible, which calls for the usage of some other search method.

2.7.1 Heuristic approaches

Vielva *et al.* [158] present some heuristic approaches to invert the underdetermined problem. The simplest one, which is called 1-D in that work, consists in selecting the estimated mixing direction $\hat{\mathbf{a}}_n$ closer to a given data point \mathbf{x} , and then projecting the data upon that direction to get its contribution to the given mixture: $s_n = \hat{\mathbf{a}}_n^T \mathbf{x}$. This hard-assignment method assumes that, at each time or time-frequency point, each mixture is contributed by a single source, which is only approximately valid for highly sparse signals (see Sect. 3.3). Hard assignment of mixing directions was also used for separation by Lin *et al.* [99].

The second approach, M-D ($M > 1$), selects M mixing directions for each \mathbf{x} according to a given criterion, and then inverts the problem by means of a $M \times M$ square reduced mixing matrix $\hat{\mathbf{A}}_\rho$ with the selected vectors as its columns: $s_n = \hat{\mathbf{A}}_\rho^{-1} \mathbf{x}$. If the criterion is to select the columns that minimize the ℓ_1 or ℓ_2 norms of the projection, the M-D approach is equivalent to the methods presented in the next section.

2.7.2 ℓ_1 and ℓ_2 minimization

A more formal solution to mixing matrix estimation can be derived from the general MAP formulation of Eq. 2.69. When $\hat{\mathbf{A}}$ is known beforehand, it reduces to the more tractable problem

$$\hat{\mathbf{S}} = \underset{\mathbf{X}=\hat{\mathbf{A}}\mathbf{S}}{\operatorname{argmin}} \left\{ \frac{1}{2\sigma^2} \left\| \mathbf{X} - \hat{\mathbf{A}}\mathbf{S} \right\|_F^2 - \sum_{n,t} l_n(s_n(t)) \right\}. \quad (2.81)$$

In the noise-free case, it is now possible to omit the first term, in contrast with the joint optimization problem, in which the term was needed to include the mixing information in the minimization process. Thus, the problem further reduces to

$$\hat{\mathbf{S}} = \underset{\mathbf{X}=\hat{\mathbf{A}}\mathbf{S}}{\operatorname{argmin}} \left\{ - \sum_{n,t} l_n(s_n(t)) \right\}, \quad (2.82)$$

i.e., to minimize the contribution of the negative log-densities of the sources evaluated at the data points (either time samples or transformation coefficients).

If the sources are assumed to be Gaussian (i.e., non-sparse), the log-densities take the form $l_n \propto -s_n(t)^2$ and the problem becomes

$$\hat{\mathbf{S}} = \underset{\mathbf{X}=\hat{\mathbf{A}}\mathbf{S}}{\operatorname{argmin}} \left\{ \sum_{n,t} s_n(t)^2 \right\} \quad (2.83)$$

which, assuming the samples/coefficients are real, equals to minimizing the Euclidean (ℓ_2) norm of the signals. It can be shown [153, 158] that there is a closed solution to this problem, given by

$$\hat{\mathbf{S}} = \hat{\mathbf{A}}^+ \mathbf{X}, \quad (2.84)$$

where $\hat{\mathbf{A}}^+$ is the Moore–Penrose pseudoinverse, computed as

$$\hat{\mathbf{A}}^+ = \hat{\mathbf{A}}^T (\hat{\mathbf{A}} \hat{\mathbf{A}}^T)^{-1}. \quad (2.85)$$

If the sources are however assumed to be sparse, which is more appropriate for the purposes of the present work, and which corresponds more closely to the characteristics of sound data transformed into the time–frequency domain, the pseudoinverse solution is not optimal. Particularly, if the sources are assumed to be Laplacian, then $l_n \propto -|s_n(t)|$ and the formulation

$$\hat{\mathbf{S}} = \underset{\mathbf{X}=\hat{\mathbf{A}}\mathbf{S}}{\operatorname{argmin}} \left\{ \sum_{n,t} |s_n(t)| \right\} \quad (2.86)$$

becomes an ℓ_1 norm minimization problem, which can be solved by linear programming techniques. ℓ_1 minimization and its equivalent geometrical interpretation, the *shortest path* algorithm, are usual methods of choice in underdetermined staged source separation [23, 24, 98, 150, 182]. It is also the method chosen for the stereo separation cases in the present work, and therefore will be further detailed in Sect. 3.5.

Takigawa *et al.* [150] perform a thorough performance evaluation of ℓ_1 -norm minimization solutions, as well as their comparison with the results obtained by the pseudoinverse solution. It was experimentally observed that ℓ_1 solutions are better (i.e., closer to the original sources) than pseudoinverse solutions for highly sparse signals, and that both methods are nearly equivalent in respect of performance if the sources are not sparse. Similar results were reported by Vielva *et al.* [158].

2.7.3 Time–frequency masking

When working in the time–frequency domain, an intuitive way of performing separation is to consider a set of time–frequency masks, one for each source/mixture pair, such that they approximately produce the separated sources when bin-wise

multiplied with the mixture. Formally, the n -th time–frequency source $\hat{\mathbf{S}}_n(r, k)$ is produced from the m -th mixture $\mathbf{X}_m(r, k)$ by

$$\hat{\mathbf{S}}_n(r, k) = \mathbf{M}_{mn}(r, k) \circ \mathbf{X}_m(r, k) \quad (2.87)$$

where $0 \leq M_{mn}(r, k) \leq 1, \forall r, k$ and the \circ operator denotes the Hadamard (element-wise) product. Note that this corresponds to filtering the mixture with a set of time-varying frequency responses. The solution to the separation problem consists in deriving the masks from the mixture.

Binary time–frequency masking is the special case in which $M_{mn}(r, k)$ can only take the values 0 or 1. Binary masks are the basis of the DUET (Degenerate Unmixing Estimation Technique) system for the separation of delayed mixtures, proposed by Yilmaz and Rickard [181], which has shown good performance with stereo mixtures of up to 6 sources. The amplitude and delay mixing parameters are estimated as the peaks of a histogram (thus, the algorithm also includes a mixing matrix estimation stage) and the binary masks are constructed by selecting the bins close to the histogram peaks. DUET is an efficient algorithm that can be implemented in real time [11]. Although the DUET method is not applicable in the present work because it assumes delayed rather than instantaneous mixtures, some of its principles will be used in the next chapter to compare sparsity properties of time–frequency representations (Sect. 3.3).

A disadvantage of binary masking is that it often produces highly audible “musical noise” artifacts due to its unnatural, discrete nature [164]. This effect can be reduced by using non-binary time–frequency masking methods such as *adaptive Wiener filtering* [18], which can be used without any spatial information to separate single-channel mixtures.

2.8 Computational Auditory Scene Analysis

All the previously introduced methods address the separation problem from a purely mathematical point of view by observing and exploiting statistical properties of the sources or the mixtures. There is an alternative approach that seeks to imitate the human mechanisms of hearing with the hope of understanding our ability to perceive sound objects present in a mixture as separate entities. This cognitive process was called *Auditory Scene Analysis* (ASA) by psychologist Albert Bregman [25]. In the cited work, Bregman proposes five grouping principles which the brain uses to isolate and detect sound events: proximity, similarity, good continuation, closure and common fate. The grouping principles refer to the temporal and frequency parameters of perceived sounds, and can be ascribed to the ideas of the *Gestalt psychology* [55], which explains the perception of objects as a whole rather than as a sum of constituent parts.

Computational Auditory Scene Analysis (CASA) [132, 177] refers to the set of algorithms developed with the aim of simulating ASA processes. It proposes a range of computational models that mimic the stages of psychoacoustical perception, from acoustical processing in the outer and inner ear, to neural and cognitive processes

in the brain. Such models are used to produce a time–frequency signal front-end upon which Bregman’s grouping principles are applied. In contrast to BSS methods, CASA employs at most two mixture channels to simulate binaural localization.

Auditory model front-ends include as the first stage an auditory filter bank that simulates the nonuniform frequency resolution of the basilar membrane, and in advanced models also the mechanical characteristics of the vibrating hair cells. This results in a time–frequency representation with nonuniform resolution that is called a *cochleagram*. Sometimes, a second stage consisting of computing the autocorrelation on each channel is performed, yielding a 3-dimensional (time–frequency–lag) front-end called *correlogram* that is useful to examine periodicities [53].

CASA methods can be divided into two groups according to how they implement the grouping rules: data-driven (or bottom-up) and prediction-driven (or top-down) methods. The earliest CASA systems were data-driven [108, 26]: they detect local features on the front-end and systematically apply grouping. This approach was later criticized [57, 142] as lacking robustness and failing to model the importance of non-local information. To overcome this, prediction-driven methods were proposed [57], in which a set of object models are defined a priori. A set of hypotheses about which objects constitute the observed signal is evaluated. A hypothesis score measures how well the predicted objects fit the observation, and the hypothesis corresponding to the highest score is selected. Note that the objects are defined as generic sound events that do not necessarily correspond to a semantic entity such as a word, a phoneme or a musical note. For instance, Ellis [57] uses three types of generic objects: noise clouds, transient clicks and *wefts* (which group all time–frequency bins having common periodicity properties). Prediction-driven analysis can be further extended by providing source-specific a priori information, such as in the music analysis system proposed by Kashino and Murase [86].

Van der Kouwe *et al.* [154] perform a comparison of BSS and CASA methods for the task of speech separation. BSS was represented by two different ICA methods and CASA by a data-driven method. An overall better performance with BSS under noisy conditions was obtained, but this result can be hardly generalized because of the specific algorithms and test corpora used. The main advantage of CASA against BSS is its easier applicability to more realistic mixtures that do not comply with the strict statistical constraints of BSS. In this context, the convenience of combining both approaches in the form of hybrid BSS/CASA systems was pointed out.

Several methods in this work have been inspired by CASA and psychoacoustics. Frequency warpings approximating auditory resolutions are used to improve sparsity and separation quality in Chapter 3. The separation approaches proposed in Chapters 5 and 6 exploit both common-fate and good-continuation grouping principles, in combination with pre-trained spectral models that estimate instruments and overlapping partials in a prediction-driven fashion.

2.9 Summary

This introductory chapter provided a comprehensive overview of the principles and methods for blind separation, for which only very generic statistical properties of

the sources can be assumed a priori. Although several methods developed in later chapters (especially chapters 4 and 5) make use of source-dependent modeling, and are thus in effect non-blind, they rely on the basic ideas that have been discussed. Special emphasis has been made on the particular separation scenario this work deals with: underdetermined separation of instantaneous mixtures.

Possible formulations of the separation problem as either a linear, delayed or convolutive mixing model, have been presented, together with a survey on stereo recording techniques that showed the applicability of each model. A discussion on basic signal models put emphasis on two methods that play a central role in the present work: sparse transformations and Principal Component Analysis. The generic staged architecture for source separation, consisting of a mixing matrix estimation and a source resynthesis stage, was presented, followed by a review of previously proposed methods covering those two separation problems, including Independent Component Analysis and clustering methods for the mixing matrix estimation stage, and norm-minimization and time–frequency masking methods for the source estimation stage. Finally, the alternative methodology offered by Computational Auditory Scene Analysis, relying on psychoacoustics and cognitive processes, rather than on statistics, was introduced.

3

Frequency-warped blind stereo separation

Most algorithms for underdetermined separation are based on the assumption that the signals are sparse in some domain. Sparse decompositions were briefly introduced in Sect. 2.3.3, and an example was given that demonstrated the enormous gain in sparsity when moving from the temporal to the frequency or time–frequency domains. Similar observations have been made in several works [23, 24, 181]. In most cases, the sparser the sources, the less they will overlap when mixed (i.e., the more disjoint their mixture will be), and consequently the easier their separation will be. The only situation in which this affirmation does not hold is the unlikely worst-case scenario in which the sources have identical probability distributions and spatial positions.

The most widely used transform for the purpose of *sparsification* in the context of BSS has been the STFT [5, 12, 18, 21, 23, 24, 83, 122, 181]. However, the uniform frequency and time resolutions it offers are disadvantageous for the task of speech or music separation. There are several reasons for this. One is that speech and music signals concentrate most of their energy in the middle-low part of the spectrum, and therefore overlaps are more likely to occur in that area. Also, musical notes follow a logarithmic frequency relationship that does not correspond with the linearly spaced subbands of an STFT spectrogram. Notes in the lower range often fall into the same subbands and will thus overlap. A final and important aspect to note is that, for music signals, different time granularities at different frequencies are more likely to better represent the signals. It is natural in music that melodies at low pitches tend to move more slowly (such as a bass line providing the harmonic base), and high-pitched melodies to change more quickly (such as an ornamented melody line).

To overcome this, the application of multiresolution analysis to source separation has been proposed, in particular through the use of wavelets [88, 182]. The standard wavelet transform provides a constant-Q, non-uniform time–frequency representation (sometimes called *scalogram*), with high frequency resolution for low frequencies and high time resolution for high frequencies. This decomposition is adequate for music signals and resembles human auditory perception. In the cited works, it was shown to improve sparsity and therefore separation when compared to the STFT.

A different approach comes from the CASA field (Sect. 2.8), in which the several stages of auditory perception (from the acoustical processing in the ear to the neural and cognitive processes in the brain) are more closely imitated in order to characterize mixtures and perform sound separation. Such systems employ more sophis-

ticated, non-constant-Q frequency warpings derived from psychoacoustical scales, usually implemented as nonuniform auditory filter banks.

The convenience of combining such auditory simulations with the mathematically formal framework of blind separation, in the form of hybrid BSS/CASA systems, has recently been pointed out [154]. In this context, the motivation that led to the developments reported in the present chapter was to explore to what extent a representation front-end more closely related to the human hearing system than the standard spectrogram or wavelets can improve separation when applied to purely statistical spectral BSS. In other words, the purpose is to objectively evaluate the potential for improvement of the use of nonuniform-resolution representations as the sparse transformation stage of the general staged separation architecture of Fig. 2.8.

To that end, an experimental framework to measure in detail several aspects relevant to separation was set up and applied to 5 different time–frequency representations: the STFT as uniform-resolution baseline for comparison, a constant-Q (CQ) logarithmic frequency warping and three representations in which the frequency resolution has been warped according to the Bark, *Equal Rectangular Bandwidth* (ERB) and Mel psychoacoustical scales. A general definition for frequency-warped representations, together with the individual characteristics of each mentioned representation, will be introduced in Sect. 3.1. The experiments that were performed can be divided into two types, according to the general aspect they are intended to measure:

- **Intrinsic (algorithm-independent) properties of warped representations.** The goal here was to measure how the representations can facilitate separation from a general point of view, involving measures that are based on the characteristics of the representations themselves, rather than on final separation quality results, which inevitably depend on the particular separation algorithm used. These experiments involve measuring both the source sparsity and the *disjointness* of the mixtures. The latter is a more powerful concept that takes into account the degree of overlapping that occurs during the mixing process. Sparsity tests will be reported in Sect. 3.2, and the results of the disjointness experiments, together with the related objective measure of W-Disjoint Orthogonality, will be discussed in Sect. 3.3.
- **Evaluation of the separation quality in the context of a practical separation algorithm.** From a practical point of view, the definitive improvement measure is the quality of the separated signals. To evaluate it, a staged separation algorithm was implemented and adapted to admit generalized temporal and frequency resolutions. The relevant quality aspects evaluated are: accuracy of the mixing matrix estimation, separation errors due to interferences, separation errors due to artifacts, and overall distortion. The corresponding stages and experiments are presented in Sect. 3.4 for the mixing matrix estimation and Sect. 3.5 for the source estimation stage.

A summary of conclusions, and their implications in the development of the subsequent chapters, will be discussed in Sect. 3.6. Parts of this chapter have been

previously published in [30] and [31].

3.1 Frequency-warped representations

A general discrete time–frequency representation $X(r, k)$, where r is the time frame and k is the frequency or band index ($k = 0, 1, \dots, K - 1$), can be interpreted as the output of a K -channel bank of bandpass filters. Its frequency resolution is determined by the center frequencies of the filters f_k and by their bandwidths Δf_k . If the center frequencies are located nonuniformly along the linear frequency axis, the resulting time–frequency representation is said to be *frequency-warped*. Frequency-warped techniques for audio signal processing have been proposed since the early years of digital technology [120], but have rather infrequently been deployed in practical applications. A general overview of frequency warping and its application to linear prediction, adaptive filtering, equalization and physical modeling can be found in the work by Härmä *et al.* [70].

Each channel of the nonuniform filter bank can be downsampled according to its bandwidth. Note that avoiding downsampling does not increase time resolution which, according to the uncertainty principle (see Sect. 2.3.2), is bounded by the bandwidth, and only provides redundant interpolated data. Filter banks whose channels are downsampled as much as possible without data loss are called *critically downsampled* or *maximally decimated*.

The individual impulse responses $h_k(t)$ of such a filter bank can be obtained by modulating and scaling a prototype impulse response $w_{T_k}(t)$ of length T_k samples, where f_s is the sampling rate (recall that $f_k = \frac{k}{T_k} f_s$), giving:

$$h_k(t) = \frac{1}{T_k} w_{T_k}(t) e^{j2\pi f_k t / f_s}. \quad (3.1)$$

A filter bank defined in this way is called *modulated filter bank*. The bandwidth of each channel, defined as the main-lobe width of the frequency response of its impulse response, is given by

$$\Delta f_k = B_{ml} \frac{f_s}{T_k} = \frac{B_{ml}}{L_k}, \quad (3.2)$$

where T_k is the window length in bins, L_k is the window length in seconds, and B_{ml} is the main-lobe width in bins, a parameter given for each type of impulse response. For Hann windows, which are the ones used in the present chapter as impulse responses, the main-lobe width is $B_{ml} = 4$ bins [2]. Eq. 3.2 is the mathematical expression of the trade-off between time and frequency resolution resulting from the uncertainty principle conceptually introduced in Sect. 2.3.2. Specifically, the parameter $B_{ml} = \Delta f_k L_k$ can be interpreted as the constant area of the time–frequency tiles.

The output of such a general filter bank is thus given by the convolution

$$X(t, k) = x(t) * h_k(t) \quad (3.3)$$

which, after downsampling, gives the most general expression for a frequency-warped time–frequency representation:

$$X(r_k, k) = \frac{1}{T_k} \sum_{t=0}^{T_k-1} x(r_k H_k + t) w_{T_k}(t) e^{-j2\pi f_k t / f_s}, \quad (3.4)$$

where H_k is the channel-dependent hop size/downsampling factor. The notation r_k denotes that the frame rate differs among frequency bands. The normalization factor T_k is needed to compensate the resulting varying data density. The expression 3.4 is sometimes called *Generalized STFT* [146].

Analogously, a generalized modulated warping can be expressed in the time domain as a *Generalized Gabor Expansion* (see also Eq. 2.25) of the form

$$x(t) = \sum_{k=0}^{K-1} \frac{1}{T_k} \sum_{r_k=-\infty}^{+\infty} X(r_k, k) w_{T_k}(r_k H_k + t) e^{j2\pi f_k t / f_s} \quad (3.5)$$

with time–frequency atoms $b_{r_k k} = w_{T_k}(r_k H_k + t) e^{j2\pi f_k t / f_s}$ and coefficients $c_{r_k k} = X(r_k, k)$.

Formulation for piecewise linear source separation

For the purpose of source separation, the generalized STFT/Gabor coefficients $c_{r_k k}$ corresponding to each signal are subjected to lexicographic ordering (Eq. 2.23) with varying time resolution, and thus concatenated to coefficient vector

$$\mathbf{c} = (c_{11}, c_{21}, \dots, c_{R_1 1}, c_{12}, \dots, c_{R_2 2}, \dots, c_{R_K K})^T \quad (3.6)$$

of size $(\sum_{k=0}^{K-1} R_k) \times 1$, where R_k is the total number of frames in the k -th subband. Then, arranging the coefficient vectors \mathbf{c}_n corresponding to the n -th source as the rows of matrix \mathbf{C} , and the coefficient vectors \mathbf{y}_m corresponding to the m -th mixture as the rows of matrix \mathbf{Y} , it is possible to use the usual transformed linear mixing model $\mathbf{Y} = \mathbf{A}\mathbf{C}$ (see Sect. 2.4) for frequency-warped separation as long as the coefficients are linearly additive in the transformed domain. This is true for the complex-valued STFT and for the filter bank subband coefficients, which in essence are time-domain signals and thus additive in amplitude. This is however not the case for absolute-valued or power representations such as magnitude or power spectrograms.

It is important not to confuse the nonlinearity of the time–frequency resolution, such as in warped representations, with the fact that such representations are indeed linearly separable when mixed. In other words, additivity of the corresponding time–frequency tiles of different transformed signals is enough to comply with the linear mixing model, irrespective of their boundary definition. This can explicitly be expressed by rewriting the transformed mixing model as

$$\mathbf{y}_{r_k k} = \mathbf{A}\mathbf{c}_{r_k k}, \quad (3.7)$$

where this time $\mathbf{c}_{r_k k}$ represents the N -dimensional signal bin and $\mathbf{y}_{r_k k}$ the M -dimensional mixture bin at time–frequency position (r_k, k) . This is the basis of *piecewise linear source separation*, as proposed by Gribonval [68]: a transformed separation problem can be reduced to a set of local separation problems (in the present case, $\sum_{k=0}^{K-1} R_k$ problems) whose results can be added to obtain the global result. This allows using the present definition of frequency warping within a separation context.

The STFT as a filter bank

The STFT, which was introduced in Sect. 2.3.2 and defined in Eq. 2.24, is equivalent to a bank of $T = K$ filters equally spaced at the frequencies

$$f_k^{\text{STFT}} = k \frac{f_s}{K}, \quad (3.8)$$

with constant bandwidth

$$\Delta f_k^{\text{STFT}} = B_{ml} \frac{f_s}{K}, \quad (3.9)$$

and with a fixed-length prototype impulse response of length K . For details on the STFT-filter bank analogy, see, e.g., [65]. Figure 3.1(a) shows the combined frequency response of a 17-band STFT filter bank based on a Hann window as prototype impulse response. The STFT yields thus a time–frequency representation with uniform frequency and time resolutions.

Constant-Q Representation

The most straightforward way to obtain a nonuniform resolution is to force the center frequencies and bandwidths to be logarithmically related to frequency, i.e., to grow geometrically. This corresponds to a constant frequency-to-bandwidth ratio. In the context of bandpass filter processing, this ratio is called the *quality factor* Q :

$$Q = \frac{f_k}{\Delta f_k}. \quad (3.10)$$

This results in a perfectly logarithmic frequency warping which, for the same number of bands, has a higher frequency resolution at low frequencies, and a lower frequency resolution at high frequencies than the STFT (the inverse applies to time resolution). Such a *constant-Q* (CQ) representation [27] can be especially useful for the analysis of music signals, since pitches of consecutive notes on the equal-temperament chromatic scale are exactly logarithmically spaced¹. In general, a geometrical subdivision of each octave into b notes gives the center frequencies

$$f_k^{\text{CQ}} = f_0 2^{\frac{k}{b}}, \quad (3.11)$$

¹The equal-temperament chromatic scale is the result of dividing each octave into 12 geometrically-spaced frequencies, each one corresponding to a *semitone*. It is by far the most commonly used musical scale.

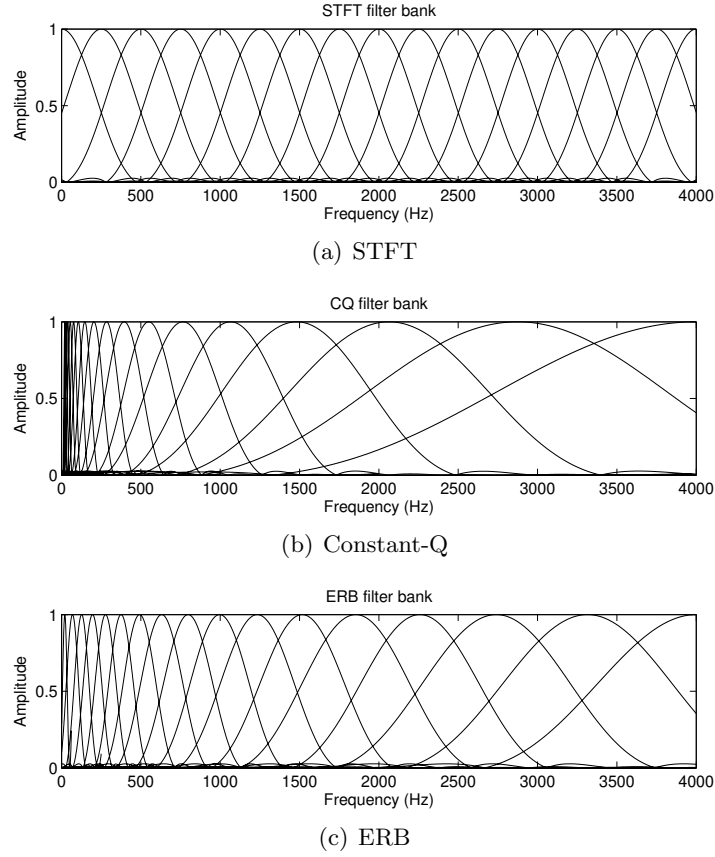


Figure 3.1: Normalized frequency responses of 17-band, Hann-window filter banks at a sampling rate of 8 kHz.

where f_0 is the lowest fundamental frequency expected. For the western chromatic scale, $b = 12$. Then, for adjacent filters, the bandwidths become

$$\Delta f_k^{\text{CQ}} = B_{ml}(f_{k+1}^{\text{CQ}} - f_k^{\text{CQ}}) = B_{ml}f_k^{\text{CQ}}(2^{\frac{1}{b}} - 1). \quad (3.12)$$

This results in the constant Q factor

$$Q = \frac{1}{B_{ml}(2^{\frac{1}{b}} - 1)}. \quad (3.13)$$

Finally, the corresponding window lengths are given by

$$T_k^{\text{CQ}} = B_{ml}Q \frac{f_s}{f_k^{\text{CQ}}}. \quad (3.14)$$

These constraints lead to the standard definition of the Constant-Q Transform (CQT) [27]:

$$X^{\text{CQ}}(k) = \frac{1}{T_k^{\text{CQ}}} \sum_{t=0}^{T_k^{\text{CQ}}-1} x(t)w_{T_k^{\text{CQ}}}(t)e^{-j 2\pi Q t/T_k^{\text{CQ}}}, \quad (3.15)$$

which can be regarded as a generalization of the DFT in which the digital frequency $2\pi k/K$ has been replaced with $2\pi Q/T_k$. Note that this standard definition of the CQT is instantaneous, i.e., it is only valid for a single signal frame, and thus only uses frequency indexing. In practice however, short-time CQTs can be computed in succession. This also differs from the more general filter bank implementation resulting from applying the previous f_k and Δf_k to Eq. 3.4, which was used here. In this case, the analysis step is not synchronous between frames, and depends on the downsampling factor that each frequency band allows. This ensures that no original signal data is lost. Another difference between the filter bank implementation and the general CQT definition is that the latter defines filter bandwidth as the distance between consecutive center frequencies, rather than taking into account the main-lobe width of the window, and thus it assumes $B_{ml} = 1$. To make this distinction clear, the term *Constant-Q* or *CQ representation* instead of CQT will be used here to refer to the filter bank implementation. A CQ representation is also offered by the *Discrete Wavelet Transform* (DWT), which can be implemented as a cascaded filter bank in which each low-pass subband is further subdivided and downsampled in successive stages.

Figure 3.1(b) shows the normalized frequency response of a 17-channel CQ filter bank with Hann windows. It can be observed that the time-resolution trade-off is strongly biased towards improving frequency resolution in the low-frequency area.

Bark representation

CQ warping is formally and computationally simple, but does not exactly correspond to the nonlinear resolution of the cochlea. More accurate approximations to it are provided by empirically obtained auditory or psychoacoustical scales, in which frequencies are mapped into a linear auditory quantity according to experimental measurements. The resulting filters are equally spaced in the auditory scale, but nonuniformly spaced in frequency. Three of the most common auditory scales will be used in the present work for frequency warping: two related to the concept of *critical bands* (the Bark and the ERB scales) and one related to the nonlinear perception of pitch ratios (Mel scale).

The Bark scale [183] defines an analytical approximation to measurements of the *critical bands* of hearing, which are ranges in the basilar membrane in which neighboring frequencies interact. This interaction occurs because each sinusoidal component arriving to the cochlea causes a certain portion of the basilar membrane to vibrate due to resonance. Although the resonating portions are approximately of constant length along the membrane (around 1 mm), the latter's uneven elasticity results in nonlinear frequency relationships. A commonly used analytical expression for the critical bandwidths is the one given by Zwicker [183]:

$$\Delta f^{\text{BARK}} = 25 + 75 \left[1 + 1.4 \left(\frac{f}{1000} \right)^2 \right]^{0.69}. \quad (3.16)$$

The Bark scale converts such a nonlinear bandwidth definition into a linear scale in which each Bark unit corresponds to each one of the 24 existing critical bands. Such

a mapping to the auditory scale ξ^{BARK} in Bark units can be approximated by [136]

$$\xi^{\text{BARK}} = 7 \operatorname{arcsinh} \left(\frac{f}{650} \right). \quad (3.17)$$

The two previous equations provide the nominal definition of the Bark scale. For the purposes of this work these definitions must be adapted as a consequence of the following two facts:

- An appropriate performance comparison between different time–frequency representations must be made under the same resolution constraints, i.e., with each representation having the same number of frequency bands. All experiments reported in the present chapter were performed by varying the number of filter subbands as the basic resolution-related parameter. This means that, in the Bark case for instance, the fixed nominal definition of 24 bands must be adapted so that a desired number of K bands is obtained instead. To that end, Eq. 3.17 is linearly sampled between the values corresponding to 0 Hz and the Nyquist frequency $f_s/2$ with the desired number of bands K . Such a linear sampling will be denoted by ξ_k^{BARK} . Then, the inverse mapping is applied to obtain the effective center frequencies:

$$f_k^{\text{BARK}} = 650 \sinh \left(\frac{\xi_k^{\text{BARK}}}{7} \right). \quad (3.18)$$

The sampled f_k^{BARK} values are then substituted into Eq. 3.16 to finally obtain the sampled bandwidths.

- The bandwidths must also be modified according to the desired band number K . Filter channels narrower than the critical bands are needed to obtain an acceptable frequency segregation for source separation. Thus, for $K > 24$, the range corresponding to one Bark unit must be subdivided, and the filter bandwidths accordingly adapted. In this way, the final adapted filter bandwidths are obtained as $\Delta f_k^{\text{BARK}} = \Delta f^{\text{BARK}}(f_k^{\text{BARK}})/B^{\text{BARK}}$, where B^{BARK} is the number of bands per Bark unit, obtained also from the linear sampling of Eq. 3.17. The window lengths are, then:

$$T_k^{\text{BARK}} = B^{\text{BARK}} B_{ml} \frac{f_s}{\Delta f_k^{\text{BARK}}}. \quad (3.19)$$

These considerations also apply to the two other auditory scales introduced next.

Equal Rectangular Bandwidth (ERB) representation

An alternative description of critical band frequency discrimination is offered by the *Equal Rectangular Bandwidth* (ERB) scale, proposed by Moore and Glasberg [112]. It differs from the Bark scale in the critical band measurement method that was employed in the experiments (the so-called *notched-noise* method instead of the probe tone/narrowband masking method employed for the Bark scale), and in the

length it assumes to correspond to a critical band, which is shorter (0.86 mm of the basilar membrane). Its nominal bandwidth definition is given by:

$$\Delta f^{\text{ERB}} = 24.7 + \frac{f}{9.26}. \quad (3.20)$$

The mapping to ERB units is:

$$\xi^{\text{ERB}} = 9.26 \ln \left(\frac{1}{228.7} f + 1 \right). \quad (3.21)$$

After linear sampling to ξ_k^{ERB} , the inverse mapping is

$$f_k^{\text{ERB}} = 228.7 \exp(\xi_k^{\text{ERB}}/9.26 - 1) \quad (3.22)$$

and, again, the actual filter bank bandwidths are $\Delta f_k^{\text{ERB}} = \Delta f^{\text{ERB}}(f_k^{\text{ERB}})/B^{\text{ERB}}$, where B^{ERB} is the number of bands per ERB unit.

The frequency response of a 17-band ERB filter bank is shown in Fig. 3.1(c). It can be observed that the resolution is more balanced between the high and low frequency areas when compared to the CQ representation.

Mel representation

The Mel scale, originally introduced by Stevens *et al.* [149], was derived from the nonlinear perception of pitch ratios and, in contrast to the ERB and Bark scales, it is not defined in terms of bandwidths, but as a direct mapping between frequencies and Mel units ξ^{MEL} :

$$\xi^{\text{MEL}} = 2595 \log_{10} \left(1 + \frac{f}{700} \right). \quad (3.23)$$

The sampled inverse mapping is:

$$f_k^{\text{MEL}} = 700 \left(10^{\xi_k^{\text{MEL}}/2595} - 1 \right). \quad (3.24)$$

The bandwidth per Mel unit (which is much smaller than an ERB or a Bark unit) can be obtained as $\Delta f^{\text{MEL}} = df^{\text{MEL}}/d\xi^{\text{MEL}}$ [70], which gives the relationship

$$\Delta f^{\text{MEL}} = \frac{1}{1127} (700 + f) \quad (3.25)$$

and, finally, $\Delta f_k^{\text{MEL}} = \Delta f^{\text{MEL}}(f_k^{\text{MEL}})/B^{\text{MEL}}$. The Mel scale is well-known in speech and music analysis applications as the warping stage of the MFCC representation front-end (see Sect. 4.7.1).

General remarks

As already mentioned, a warped auditory representation $X(r_k, k)$ can be obtained by applying one of the previous definitions of f_k and either T_k or Δf_k to the filter bank definition of Eq. 3.1. Table 3.1 summarizes the nominal definition equations for f_k

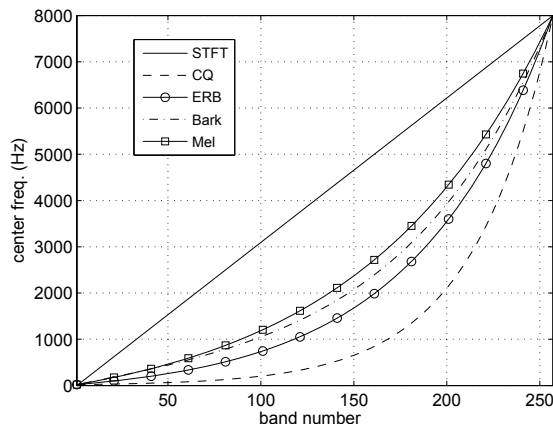


Figure 3.2: Filter bank center frequencies as a function of band number, for 16 kHz sampling rate and 257 bands.

Repr.	f_k	Δf_k	Comments
STFT	$k \frac{f_s}{N}$	$B_{ml} \frac{f_s}{N}$	Uniform resolution
CQ	$f_0 2^{\frac{k}{b}}$	$B_{ml} f_k (2^{\frac{1}{b}} - 1)$	Logarithmic resolution
ERB	$228.7 \exp(\xi_k / 9.26 - 1)$	$24.7 + \frac{f_k}{9.26}$	Approximates critical band resolution
Bark	$650 \sinh\left(\frac{\xi_k}{7}\right)$	$25 + 75 \left[1 + 1.4 \left(\frac{f}{1000}\right)^2\right]^{0.69}$	Approximates critical band resolution
Mel	$700 (10^{\xi_k / 2595} - 1)$	$\frac{1}{1127} (700 + f)$	Approximates perception of pitch ratios

Table 3.1: Summary of nominal center frequency (f_k) and bandwidth (Δf_k) definitions.

and Δf_k of all considered representations. It should be emphasized that, in order to compare representations with the same number of bands K , for each value of K the b parameter in the CQ representation and the B^{BARK} , B^{ERB} and B^{MEL} parameters must be accordingly adapted. Figure 3.2 compares the distribution of center frequencies versus subband number for the resolution-adapted representations, and for the particular case $f_s = 16$ kHz and $K = 257$.

A graphical example of the effect of frequency warping is shown in Fig. 3.3, which illustrates the effect of ERB warping on an excerpt of a clarinet playing a 5-note melody fragment. Comparing it with the magnitude spectrogram representation it can be observed that, for the same number of bands covering the same frequency range, the resolution has been enhanced in the low frequency range, where most of the signal energy is concentrated, and that the time–frequency lines corresponding to the harmonics are more clearly visible and separated. This is the main reason why auditory warpings have the potential to improve representation sparsity and

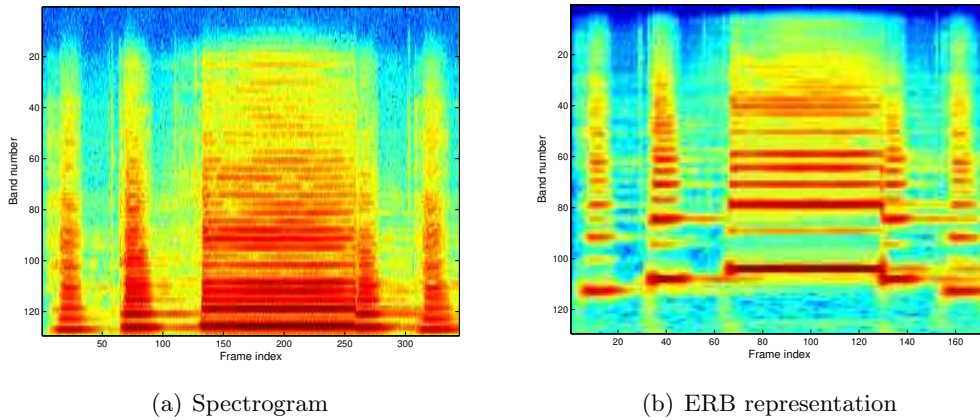


Figure 3.3: Comparison of 129-band spectrogram and ERB spectral representation of a clarinet melody.

mixture disjointness, a fact that will be confirmed in the evaluation experiments described in the rest of the present chapter. Note that the magnitude spectrogram has been used here only for illustration purposes. As was noted above, complex-valued STFTs must be used in the final algorithm to comply with the piecewise linear source separation model.

To improve the accuracy of the auditory simulations, more sophisticated auditory filter banks employ a set of special-purpose filter shapes designed to accurately model the mechanical response of the basilar membrane, such as *rounded exponential* (roex) [124] and *gammatone* [123] filters. As introduced in Sect. 2.8, the resulting time–frequency representations are called *cochleagrams*. However, in this work the emphasis is put on studying the effects of the frequency-warping stage of auditory modeling, motivated by previous results that show that spectral resolution is most crucial in improving sparsity and disjointness [11, 21, 128, 181]. For this reason, auditory filter shapes have not been used. Instead, a Hann window was chosen as the prototype impulse response/analysis window in all representations.

For the experiments reported in this chapter, a direct, downsampled implementation of the filter banks was used. This method is computationally inefficient in comparison with the STFT. However, it should be noted that more efficient implementations of frequency-warped filter banks exist, which use chains of all-pass filters [70] and can be combined with analytical expressions of the all-pass coefficient in such a way that the warping approximates an auditory scale [147].

3.1.1 Invertibility of the representations

In order for a transformation to be useful in the context of source separation, it must be invertible, so that the extracted sources can be synthesized back. The STFT is perfectly invertible in the absence of spectral modifications, as long as the analysis window fulfills the *constant-overlap-add* (COLA) condition in the time

domain [2, 65, 146]:

$$\sum_{r=-\infty}^{\infty} w(rH + t) = 1 \quad (3.26)$$

for a given hop size H . In that case, the STFT can be perfectly inverted by the *Overlap-Add* (OLA) method, consisting in adding the inverse DFTs of the successive analysis frames, as given by Eq. 2.25. In the filter bank interpretation of the STFT, this corresponds to upsampling, re-filtering and adding the outputs of the subbands, which in this context is called the *Filter Bank Summation* (FBS) method [65, 146]. The 50%-overlapped Hann window used here, as a member of the *Generalized Hamming* family of windows, fulfills the COLA condition [146].

In contrast, downsampled, nonuniform filter banks cannot generally be reconstructed perfectly by FBS [70] because they do not fulfill the COLA condition, as can be predicted from Fig. 3.1. However, perfect reconstruction is not critical in source separation, since the largest reconstruction errors are introduced by the separation algorithm itself and are much more significant than the errors introduced by inverting the transformation [160]. Thus quasi-perfect reconstruction (with inaudible error) will be sufficient, and inversion errors will not be considered harmful in the performed evaluations. Directly implemented warped filter banks can be approximately inverted by upsampling, re-filtering with the time-reversed analysis filters and adding the subbands with appropriate weighting according to their bandwidth [143].

3.2 Evaluation of source sparsity

The introductory Chapter 2 emphasized the importance of sparsity for source separation applications, especially in underdetermined scenarios. A higher degree of sparsity will a priori lead to an easier separation, no matter which specific algorithm is used. The purpose of this and the next section is to investigate the sparsity properties of warped time–frequency representations as front-ends for source separation, i.e., independently of the separation algorithm. To this end, objective measures are needed that allow an intrinsic measure of the gain in sparsity introduced by the initial transformation stage, without further consideration of other algorithm-dependent factors such as spectral artifacts or required time granularity. The present section concentrates on the measurement of sparsity of individual signals transformed with the considered warpings, and the next (Sect. 3.3) will focus on evaluating mixture disjointness, which can be loosely considered as the sparsity of the mixture signals.

3.2.1 Sparsity properties of speech and music signals

The degree of sparsity a transformation can achieve will obviously depend on the nature of the signals themselves. It is no surprise that speech and music signals possess very different properties with regard to sparsity. Speech signals are generally faster time-variant with respect to energy, and they will consequently need some minimal time resolution in order to be properly described. Music signals, specially

harmonic, sustained sounds, will be highly localized in frequency and relatively slowly time-variant.

The purpose of the first experiment was to objectively assess these differing characteristics, and to give insight into the special implications that music signals require. The normalized kurtosis $\bar{\kappa}_4$ (Eq. 2.36) was used as sparsity measure. A database of 50 speech and 50 music short audio segments of 1 s sampled at 8 kHz was used. The speech examples correspond to single-speaker utterances, and the musical ones are melody fragments played by single instruments. For each fragment, a 50% overlapped STFT, a Bark and an ERB representation based on Hann windows were computed for different band numbers, ranging from $K_0 = 17$ to $K_{P-1} = 2049$. Higher band numbers ($K_p > f_s/2$) would result in either significant data loss or zero padding of time samples at the end of the signals, thus producing less plausible measures.

Note that for real signals, the $N/2$ upper spectral bins of the STFT are redundant, and thus an N -points STFT corresponds to a spectrogram representation of $K = N/2 + 1$ bands (positive frequencies plus the DC value). For this reason, the values $K_p = N_{min}2^{p-1} + 1$ with $p = 0, 1, \dots, P - 1$ were used as evaluation points, where N_{min} is a power of two to benefit from an efficient FFT computation (in this case, $N_{min} = 32$ and $P = 8$). Figure 3.4 shows the kurtosis curves averaged for each of the databases. Table 3.2 shows the numerical values corresponding to the maxima of the respective obtained curves, together with the optimal band number for which these values were reached. The table additionally shows the result of measuring sparsity in the time domain.

A clearly differentiated behavior is obvious from these results. For all three representations considered, the speech curves have a clear peak around $K_p = [129, 513]$, whereas music sparsity increases monotonically with increasing frequency resolution (and decreasing time resolution). The decrease of speech sparsity for a high number of bands reveals the harmful effect of low time resolution: in that case, the analysis windows span a too large interval for the energy variance to be appropriately described by a single representation bin. Thus, an optimal balance between time and frequency resolution must be used when working with speech signals.

This contrasts with the music case: here, frequency resolution must be favored to maximize sparsity. Generally, the curves will decay again for extremely high frequency resolutions ($K_p > f_s/2$), as was observed in preliminary tests, but such resolutions are impractical due to computational requirements (especially for the warped representations) and to the mentioned data distortion at the end of the signal. Only in cases where music signals are highly harmonic and slowly varying (such as a slow single-voiced instrumental solo) the sparsity will be constantly, though asymptotically, increasing. This was the case of the example shown in Sect. 2.3.3 to conceptually introduce the need for sparsity. It corresponded to the extreme case where K equals the number of input samples (i.e., the STFT becomes a single DFT), for which a higher sparsity was obtained. It should be noted that, as will be shown later, factors other than sparsity also influence the final separation quality, and thus the highest sparsity does not necessarily result in the highest performance.

In global terms, a higher sparsity was achieved for music ($\bar{\kappa}_{4,max} = 385.2$ for

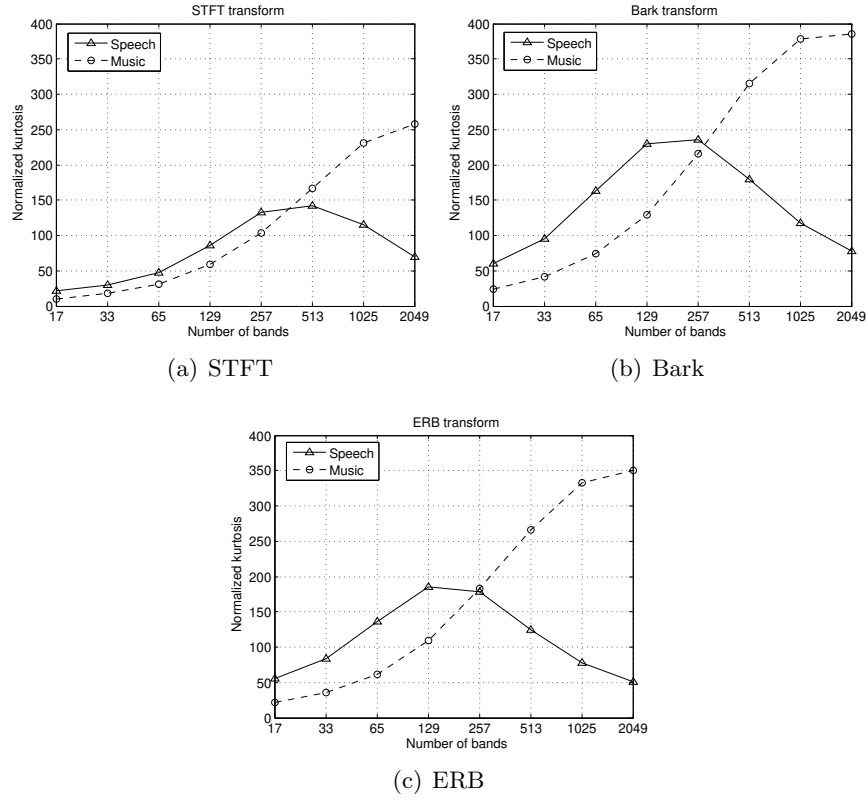


Figure 3.4: Averaged sparsity, measured by normalized kurtosis, against number of bands for speech and music sources at 8 kHz sampling rate.

Repr.	Speech		Music	
	$\bar{\kappa}_{4,max}$	opt. bands	$\bar{\kappa}_{4,max}$	opt. bands
Time	5.707	-	2.608	-
STFT	141.9	513	258.3	2049
Bark	236.0	257	385.2	2049
ERB	185.8	129	349.9	2049

Table 3.2: Maximum averaged sparsity, measured by normalized kurtosis, and optimal number of bands, for speech and music data for 8 kHz sampling rate.

a Bark warping) than for speech ($\bar{\kappa}_{4,max} = 236.0$, also with Bark) in the time–frequency domain, which corresponds to a proportional increase of 63.2%. In the time domain, however, and although the kurtosis values are much lower in both cases, speech is sparser than music ($\bar{\kappa}_4 = 5.707$ compared to $\bar{\kappa}_4 = 2.608$). This is an important fact that provides a further argument to perform a sparse transformation if the signals in study are musical.

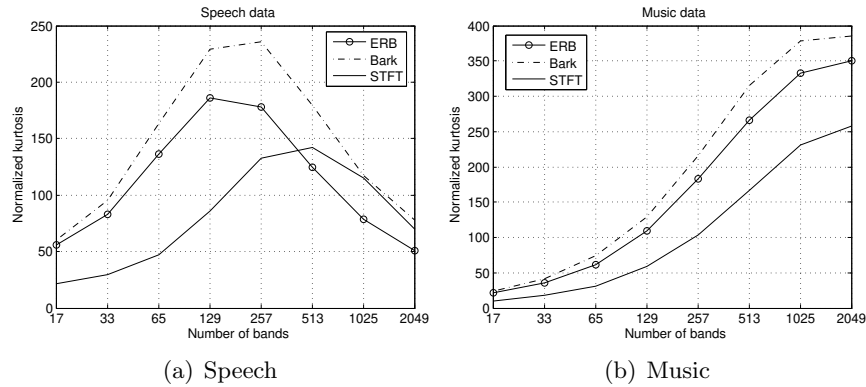


Figure 3.5: Average sparsity, measured by normalized kurtosis, against number of bands for spectrogram (STFT), Bark and ERB representations, at 8 kHz sampling rate.

3.2.2 Sparsity properties of frequency-warped signals

More interesting for the purpose of the present chapter is to compare the sparsity measurements between the representations, rather than between the audio classes. To highlight this comparison, Fig. 3.5 shows again the previous sparsity curves, this time superimposed according to representation type.

In both the speech and the music case sparsity has been considerably improved compared to the STFT using the ERB and Bark auditory warpings. The best results were obtained for the Bark representation. The proportional gain in sparsity was higher with speech (66.3% improvement from $\bar{\kappa}_{4,max} = 141.9$ with the STFT to $\bar{\kappa}_{4,max} = 236.0$ with Bark) than with music (49.1% improvement from $\bar{\kappa}_{4,max} = 258.3$ to $\bar{\kappa}_{4,max} = 385.2$).

3.3 Disjointness and W-Disjoint Orthogonality

Highly sparse sources can certainly facilitate separation, but there are several further aspects to be considered. In BSS, only the mixtures are available. Thus, more than the properties of the isolated sources, the factor that will ultimately determine the separation performance is the characteristics of the mixture itself. In particular, the crucial factor is the degree of source overlapping that occurs during the mixing process. Source sparsity alone is useless if the sources overlap to a high degree. For example, two identically distributed sources lying on very close stereo positions will be very hard to separate, no matter how sparse they are. Another important factor to note is that overlapping will not only be determined by the nature of the individual sources and the mixing process, but to the greatest extent by the mutual properties between sources, such as correlation and independence. This is especially important for music mixtures, as will be addressed in Sect. 3.3.1. All these considerations demonstrate that an objective measure of the degree of overlapping is needed in order to really assess the difficulty of separating a given mixture.

The *disjointness* D of a mixture can be defined as the degree of non-overlapping of the mixed signals. An objective measure of disjointness called *W-Disjoint Orthogonality* (WDO) was initially proposed by Jourjine *et al.* [83] and further developed by Yilmaz and Rickard [181] in the context of the DUET source separation system. WDO relies on the concept of unmixing by binary time–frequency masking, introduced in Sect. 2.7.3. If a mixture is sufficiently disjoint in some time–frequency domain, it can be used to estimate a set of unmixing masks, one for each source, that will approximately extract the desired source when applied on the mixture representation. The key idea behind the WDO-based measurement method is that the unmixing capabilities of a set of ideal masks computed from the knowledge of the sources can be also interpreted as the intrinsic disjointness of the mixture.

A pair of signals transformed into the time–frequency domain, $S_1(r, k)$ and $S_2(r, k)$, are said to be W-disjoint orthogonal if they fulfill²

$$S_1(r, k)S_2(r, k) = 0, \quad \forall r, k. \quad (3.27)$$

or, in matrix notation

$$\mathbf{S}_1(r, k) \circ \mathbf{S}_2(r, k) = 0. \quad (3.28)$$

The “W” in WDO refers to the analysis window used. Now, consider a simple one-channel linear mixture without gain factors of N sources in the transformed domain:

$$\mathbf{X}(r, k) = \sum_{n=1}^N \mathbf{S}_n(r, k). \quad (3.29)$$

If all N sources in the mixture are pairwise W-Disjoint Orthogonal, then each time–frequency bin in the mixture is only contributed by a single source. This is the condition for perfect disjointness: there is zero overlapping, and the sources will be easy to separate.

Obviously, in practice no signals fulfill Eq. 3.27 perfectly. What is needed is a measure of *approximate* WDO that describes how close the mixture is to perfect disjointness. To that end, consider the sum of all signals interfering with source n :

$$\mathbf{U}_n(r, k) = \sum_{\substack{i=1 \\ i \neq n}}^N \mathbf{S}_i(r, k). \quad (3.30)$$

In [181] it is shown that a binary time–frequency mask defined as

$$M_n(r, k) = \begin{cases} 1, & 20 \log \left(\frac{|S_n(r, k)|}{|U_n(r, k)|} \right) \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad \forall r, k \quad (3.31)$$

optimally unmixes the n -th source when applied to the mixture:

$$\hat{\mathbf{S}}_n(r, k) = \mathbf{M}_n(r, k) \circ \mathbf{X}(r, k). \quad (3.32)$$

²The multi-resolution notation r_k for the frame index will be relaxed from here on, but it will be valid in the general case.

$M_n(r, k)$ is the indicator function of the time–frequency bins to which source n contributes more than all the sources that interfere with it. Based on this ideal mask, the following two quantities can be defined. The *preserved-signal ratio* (PSR) measures the energy loss of the desired signal after unmixing:

$$\text{PSR}_n = \frac{\|\mathbf{M}_n(r, k) \circ \mathbf{S}_n(r, k)\|_F^2}{\|\mathbf{S}_n(r, k)\|_F^2}, \quad (3.33)$$

where $\|\cdot\|_F^2$ denotes the squared Frobenius norm (Eq. 2.64), or matrix energy. The *Source to Interference Ratio* (SIR) measures the energy difference between the desired signal and its interference after applying the mask:

$$\text{SIR}_n = \frac{\|\mathbf{M}_n(r, k) \circ \mathbf{S}_n(r, k)\|_F^2}{\|\mathbf{M}_n(r, k) \circ \mathbf{U}_n(r, k)\|_F^2}. \quad (3.34)$$

Finally, the approximate WDO for that particular source is defined as the normalized difference between preserved energy and interference energy:

$$\text{WDO}_n = \frac{\|\mathbf{M}_n(r, k) \circ \mathbf{S}_n(r, k)\|_F^2 - \|\mathbf{M}_n(r, k) \circ \mathbf{U}_n(r, k)\|_F^2}{\|\mathbf{S}_n(r, k)\|_F^2}, \quad (3.35)$$

which can be expressed as a function of the previous two quantities:

$$\text{WDO}_n = \text{PSR}_n - \frac{\text{PSR}_n}{\text{SIR}_n}. \quad (3.36)$$

A global measure of the disjointness of the mixture can then be measured as the averaged approximate WDO of its sources:

$$D = \overline{\text{WDO}} = \frac{1}{N} \sum_{n=1}^N \text{WDO}_n. \quad (3.37)$$

A perfect disjointness (each bin is contributed only by one source) corresponds to $\text{PSR} = 1$, $\overline{\text{SIR}} = \infty$ and $\overline{\text{WDO}} = 1$, and would result in perfect separation with the given mask.

WDO can also be used as a BSS performance measure when based on masks estimated from the mixtures without knowing the sources. However, it should be noted that in the case studied here, the above masks are derived with the sources being known, which implies that the definition of WDO used here can be interpreted as the upper bound in unmixing performance by binary masking.

Although the WDO criterion was originally defined for the STFT, it can readily be applied with other additive transformations, such as warped filter banks. It is important to note however that it is only possible to compute the PSR, SIR and WDO values in the time–frequency domain if the corresponding transform obeys Parseval’s theorem, i.e., if the signal energy in the frequency domain is proportional to the energy in the time domain. The STFT fulfills this condition. This is however not the case for transformations with nonuniform resolutions, which distribute signal energy unequally across the spectral bands, depending on the bandwidth and eventually amplitude weighting of each band. Therefore it is mandatory to invert the transform after masking and compute the energy ratios in the time domain.

3.3.1 Disjointness properties of speech and music mixtures

Like in the sparsity evaluation of the previous section, two aspects are of interest here: to compare the disjointness properties of music with those of speech, and to compare the improvement in disjointness when applying different warped transformations. This section covers the first, the next one the latter.

As already mentioned, overlapping depends not only on source sparsity and on the mixing process, but also on the mutual relationships between signals. Highly uncorrelated signals will result in a low probability of overlapping. This is even truer for statistically independent signals, since independence is a stronger requirement than uncorrelation, as was discussed in Sect. 2.6.1. Highly independent signals will be easier to separate. As it can be recalled, this is precisely the principle underlying ICA.

Speech signals most often mix in a random and uncorrelated manner, such as in the cocktail party paradigm. With music mixtures, the situation is different. Their disjointness will vary strongly according to music type. Tonal music will result in strong overlaps in frequency, and homophonic or homorhythmic music in strong temporal overlaps³. On the contrary, atonal music will be more disjoint in frequency, and contrapuntal music more disjoint in time.

These observations motivated the subdivision of the music database for the current experiments into two sub-databases. Dataset UMEL (for “uncorrelated melodies”) contains 50 fragments of instrumental solos playing unrelated melodies, i.e., melodies randomly drawn from an instrumental database which are not intended to be musically coherent when mixed. To evaluate disjointness, 50 different combinations of 3 sources were randomly extracted and mixed. Dataset CMEL (for “correlated melodies”) contains 50 sets of 3 instrumental fragments extracted from a real multitrack recording, in such a way that the resulting mixtures constitute excerpts from a coherent musical performance (in this case a saxophone quintet). Throughout the literature, uncorrelated musical mixtures are more often employed for evaluating the performance of source separators, in spite of the fact that coherent mixtures simulate closer the requirements of a practical musical unmixing application. The speech dataset (SP), like the UMEL dataset, contains 50 random mixtures of 3 speech utterances each. All files used are short fragments of 1 second, sampled at 8 kHz.

Figure 3.6 shows the disjointness D , measured as $\overline{\text{WDO}}$, averaged over all samples from each database as a function of the number of bands. As before, 50% overlapping STFT, Bark and ERB representations were chosen for the comparison. Table 3.3 shows the highest $\overline{\text{WDO}}$ achieved for each of the curves and the corresponding optimal number of bands. It also shows the disjointness of the mixtures in the time domain.

For the speech database, and for both music databases taken as a whole, the results show a correlated behavior between disjointness and the source sparsity evaluated in the previous section. In particular, speech presents an optimal balance

³The terminology for musical textures and harmony was introduced at the beginning of Chapter 2.

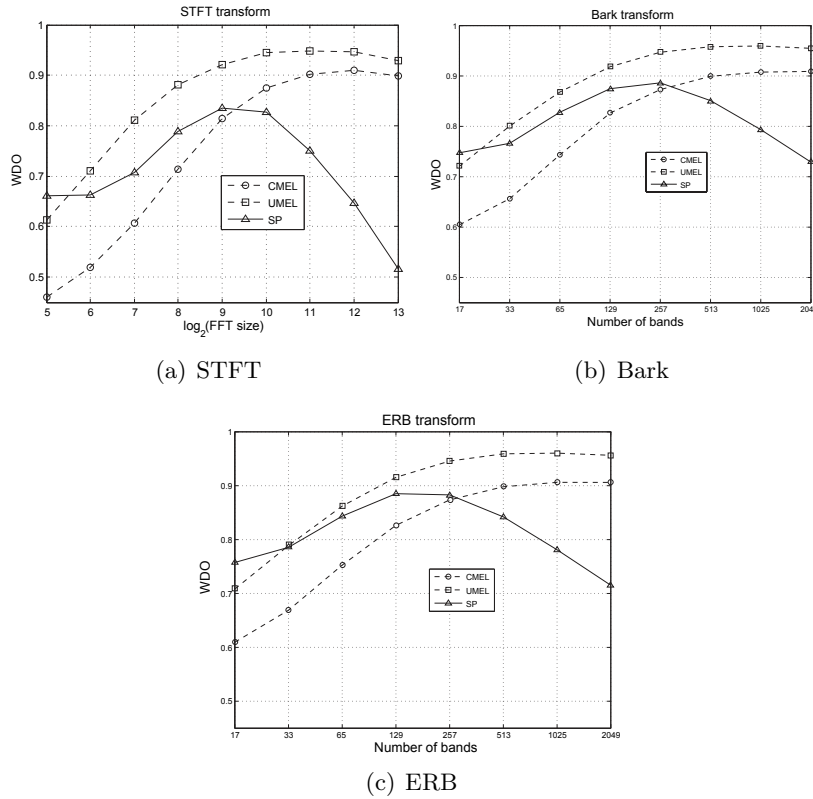


Figure 3.6: Disjointness ($\overline{\text{WDO}}$) against number of bands for speech (SP), uncorrelated music (UMEL) and correlated music (CMEL), 3-source stereo mixtures at 8 kHz sampling rate.

point between time and frequency resolution, as a consequence of the reduction in temporal resolution when moving towards large window sizes. The disjointness of music tends to monotonically increase towards the high frequency resolution area. For speech signals, a compromise should be taken to balance temporal and frequency disjointness by choosing a moderate window size (of around $K_p = f_s/2^5$), whereas for music signals, frequency disjointness plays a more important role than time disjointness and so frequency resolution should be favored. Also, a higher disjointness is possible for music than for speech, when favoring frequency resolution.

As expected, mixtures of correlated melodies are less disjoint than uncorrelated ones because of the higher amount of spectral and temporal overlapping⁴. The largest increase between UMEL and CMEL was of 5.4% $\overline{\text{WDO}}$, obtained with the ERB representation. On average, uncorrelated music is 8.8% more disjoint, correlated music however only 4% more disjoint than speech. In contrast, speech is more disjoint than music in the time domain: 70.8% $\overline{\text{WDO}}$ of speech compared to 51.1% and 55.3% $\overline{\text{WDO}}$ of CMEL and UMEL, respectively.

⁴The saxophone quintet used for the experiments is tonal and highly homorhythmic.

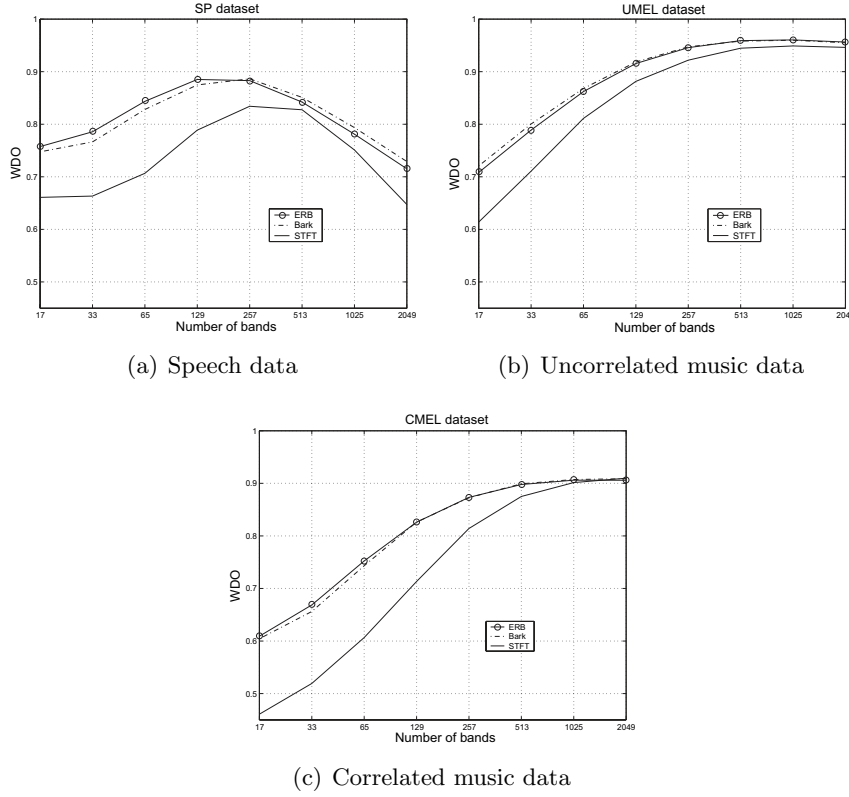


Figure 3.7: Disjointness (\overline{WDO}) against number of bands for ERB, Bark and STFT representations, 3-source stereo mixtures and 8 kHz sampling rate.

Repr.	SP		UMEL		CMEL	
	\overline{WDO}_{max}	opt. bands	\overline{WDO}_{max}	opt. bands	\overline{WDO}_{max}	opt. bands
Time	70.8	-	55.3	-	51.1	-
STFT	83.4	257	94.8	1025	90.9	2049
Bark	88.6	257	96.0	1025	90.8	2049
ERB	88.5	129	96.0	1025	90.6	1025

Table 3.3: Maximum disjointness, measured in % of \overline{WDO} , and optimal number of bands for speech (SP), uncorrelated music (UMEL) and correlated music (CMEL) data for 8 kHz sampling rate.

3.3.2 Disjointness properties of frequency-warped mixtures

In general, the previous results are analogous to the results of the sparsity experiments. When re-plotted to compare the effect of the different warping approaches (Fig. 3.7), the curves show that this analogy does not hold as clearly anymore. This is especially the case for music signals: the gain in disjointness is high when few bands are used, but decreases as the number of bands increases. In the UMEL case, the improvement is of around 5-10% in the low frequency resolution area, but

decreases to 1.2% with high band numbers. In the CMEL case, the improvement is again high in the low frequency resolution/high temporal resolution area (10-15%), but with large windows, the $\overline{\text{WDO}}$ of the STFT and the auditory scales become statistically equivalent, at around 91%. In other words, the more the sources overlap, the higher the improvement will be. For speech signals, however, the improvement is clear in all resolution areas, and more significant than with music, with an improvement of maximally achievable disjointness of 5.2% with the Bark scale. As expected, the improvement is in all cases high compared to the time domain: using the Bark scale, speech improves disjointness by 17.8%, correlated music by 39.7% and uncorrelated music by 40.7%.

These results contrast with the more pronounced improvement in sparsity obtained for all resolutions, as can be observed by comparing the $\overline{\text{WDO}}$ curves with those of Fig. 3.5. This cannot be attributed to the error introduced by inverting the warped representations, which was needed to comply with Parseval's theorem, since that error was proven to be insignificant. A further difference is that this time the behaviors of both ERB and Bark auditory scales are very similar.

Such observations demonstrate the need of studying in detail the nature of the mixing process for assessing signal separability. Maximizing intrinsic source sparsity alone does indeed improve disjointness, and thus ease of separation, at most resolutions with most of the considered mixing scenarios. There is however one situation where high source sparsity might not be enough for guaranteeing an easier separation: the case of coherent musical mixtures with a high-frequency-resolution representation. Also, note that the WDO tests were based on single-channel mixtures, as defined by Eq. 3.29. This particular situation is closer to the mentioned worst-case scenario in which all sources have identical probability distributions and spatial positions. Such a worst-case, although unlikely, would remain inseparable, independently of the degree of frequency warping applied to its representation domain. This is a first hint towards the need of adding source-related a priori information for improving separation in musically coherent mixtures. In addition, the next sections will demonstrate that, in the context of a full separation system, yet another factor arises that will prove crucial for separation quality: the distortion introduced by spectral artifacts.

3.4 Frequency-warped mixing matrix estimation

The previous two sections (3.2 and 3.3) studied the effect of warping on the representation front-end, independently of the separation algorithm used. This was achieved by objectively measuring two factors that intrinsically describe source and mixture properties: sparsity and disjointness.

Still, there are several other factors that must be assessed for a full-fledged separation scenario. Frequency warping can influence in different ways the two successive stages of a separation algorithm that follows the staged architecture that was introduced in Sect. 2.5: mixing matrix estimation and source estimation or resynthesis. The definitive test for the utility of such representations is the quality

of the separated signals. From a practical point of view, this is the most meaningful evaluation. Its disadvantage is however, that it is inevitably associated to a specific algorithm.

As baseline, the method proposed by Bofill and Zibulevsky in [23] and [24] has been chosen. It consists of a combination of kernel-based clustering for the mixing matrix estimation (Sect. 2.6.2) and of shortest path resynthesis, which is an ℓ_1 -norm minimization method, for source estimation (Sect. 2.7.2). This approach fits the purpose of the present chapter for the following reasons:

- It is designed for stereo, underdetermined, instantaneous mixtures.
- It complies with the linear piecewise separation model that has been used in the present chapter to formulate frequency-warped source separation (Eq. 3.7).
- It has shown good performance with up to $N = 6$ sources.
- It follows a fully independent staged architecture, which allows assessing mixing and source estimation separately.
- It is intuitive, which allows interpreting physically the effect of warping at each individual stage or sub-stage of the method.

This section concentrates on evaluating the effect of frequency warping on the kernel clustering mixing estimation stage, the next one will be devoted to the source resynthesis stage.

3.4.1 Kernel-based angular clustering

In Sect. 2.6, the phenomenon of direction clustering on the mixture space scatter plot for sufficiently sparse sources was introduced, together with several clustering-based methods for mixing matrix estimation. Figure 3.8 shows examples of scatter plots for a stereo ($M = 2$) mixture of $N = 3$ sources in the time, STFT and ERB domains. The plots have been normalized and downsampled so that each scatter plot contains the same number of data points. The lack of sparsity in the temporal domain prevents the mixing directions from being recognizable, and thus clustering would fail in this case. The time–frequency representations however clearly show the directions.

When performing an angular search from $\theta = 0$ to $\theta = 2\pi$, each mixing direction will correspond to two clusters lying on opposite sides with respect to the origin of the scatter plot. Alternatively, the mixture data can be projected to the first quadrant of \mathbb{R}^2 and the search range restricted to $0 \leq \theta \leq \pi/2$, with the consequent gain in computation time. For transformations yielding complex coefficients (in the present case, the STFT), the scatter plots corresponding to the real and imaginary parts, which cluster to the same directions, can be superimposed before performing

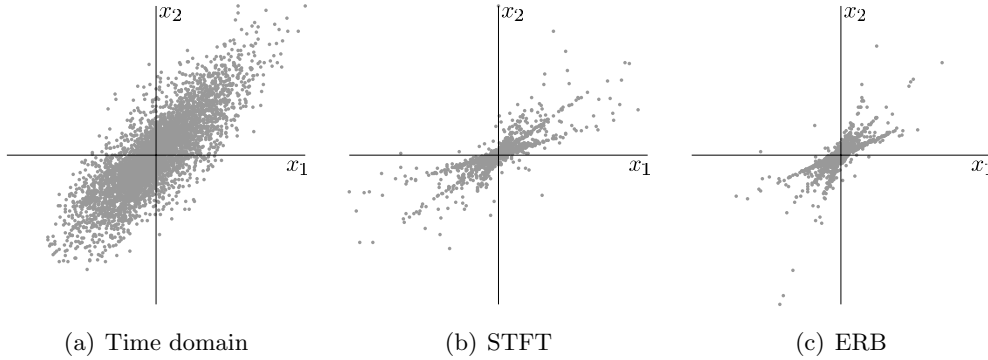


Figure 3.8: Example of 3-source, stereo mixture scatter plots for music signals.

the angular search. For these reasons, the final scatter plot that is subjected to clustering is in general given by the projection

$$\mathbf{X}_{proj} = (|Re\{\mathbf{x}_1\}|, \dots, |Re\{\mathbf{x}_C\}|, |Im\{\mathbf{x}_1\}|, \dots, |Im\{\mathbf{x}_C\}|), \quad (3.38)$$

where each data point is $\mathbf{x}_{rk} = (x_{1,rk}, x_{2,rk})$ and C is the total number of coefficients in the lexicographically ordered transformation vectors: $C = \sum_{k=0}^{K-1} R_k$. In polar coordinates, each data point is defined by its radius $\rho_{rk} = \sqrt{x_{1,rk}^2 + x_{2,rk}^2}$ and its angle $\theta_{rk} = \arctan(x_{2,rk}/x_{1,rk})$. Figure 3.9(b) shows an example of such a projection.

Due to sparsity, most of the bins accumulate near the origin. However, bins with small modules do not add much information when searching for the mixing directions, and so they can be ignored for the clustering analysis. This saves computation time and does not significantly affect the performance. The clustering threshold is denoted in Fig. 3.9(b) by the circle quadrant around the origin.

The clustering used is based on kernel density estimation or Parzen windowing (Sect. 2.6.2), which can be interpreted as a smoothed histogram obtained by assigning each data point to a weighting function, called kernel or local basis function. The angular kernel is in this case a triangular function given by

$$K(\theta) = \begin{cases} 1 - \frac{\theta}{\pi/4} & \text{if } |\theta| < \pi/4 \\ 0 & \text{otherwise} \end{cases}. \quad (3.39)$$

The estimated distribution, in this context also called *potential function*, is given by

$$\hat{p}(\theta) = \sum_{r,k} \rho_{rk} K(\lambda(\theta - \theta_{rk})), \quad (3.40)$$

where λ is a parameter controlling the width of the kernels. To perform the angular search, the quadrant must be in practice be subdivided as a radial grid of a certain angular resolution.

Figure 3.9 shows an example of clustering results. The example corresponds to a 4-source mixture transformed with a 129-band ERB warping. The original scatter

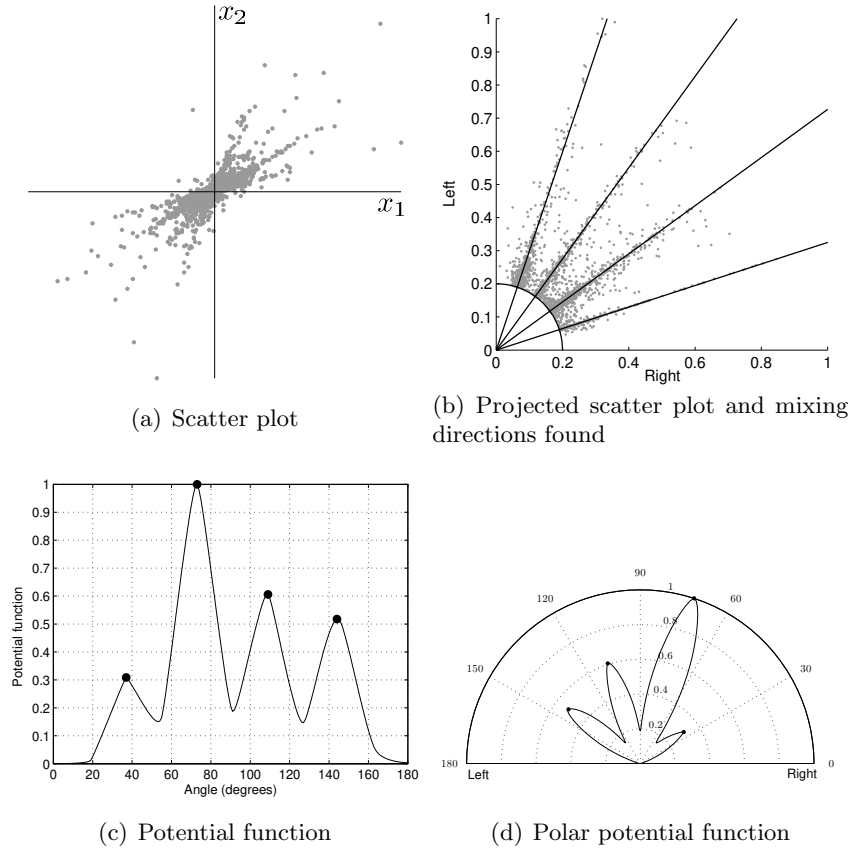


Figure 3.9: Example of mixing matrix estimation by kernel-based angular clustering.

plot (Fig. 3.9(a)) is first projected to the first quadrant and the low-amplitude bins are removed (Fig. 3.9(b)). Figures 3.9(c) and 3.9(d) show the resulting potential function in Cartesian and polar coordinates, respectively. The predicted directions $\hat{\mathbf{a}}_n$ and, implicitly, the predicted number of sources \hat{N} , are obtained by detecting the peaks of the function, marked with dots on the figure. The direction vectors found are shown superimposed to the projected scatter plot of Fig. 3.9(b).

3.4.2 Evaluation with frequency-warped representations

Improved sparsity is expected to benefit clustering accuracy, since proportionally more data points will concentrate around the true directions. In this section, the results of using the above mixing matrix estimation algorithm with the frequency-warped front-ends are presented.

For the evaluation experiments, a set of 10 stereo mixtures of $N = 3$ sources and a set of 10 stereo mixtures of $N = 4$ sources were used. The sources to be mixed were randomly extracted from a database of musical fragments of 3 s duration played by melodic instruments and sampled at 8 kHz. For each mixture, the experiment

Representation	$N = 3$ sources		$N = 4$ sources	
	DR (%)	$e_{ang}(^\circ)$	DR (%)	$e_{ang}(^\circ)$
STFT	81.3	1.22	65.0	3.38
CQ	80.0	0.75	67.5	4.82
ERB	82.5	0.76	71.3	0.83
Bark	82.5	0.78	73.8	0.90
Mel	82.5	0.76	71.3	1.50

Table 3.4: Evaluation of the mixing matrix estimation stage: averaged source detection rate (DR) and angular error (e_{ang}) in degrees, for stereo mixtures of $N = 3$ (left) and $N = 4$ sources (right).

was repeated for all time–frequency representations discussed in the present chapter (STFT, constant Q (CQ), ERB, Bark and Mel), and for a different number of representation bands K_p , ranging from $K_0 = 33$ to $K_{P-1} = 4097$. This makes a total of 800 separation experiments.

Each source was normalized, artificially panned and mixed. The mixing matrix was defined with equally spaced directions, i.e., $\theta_1 = 3\pi/4$, $\theta_2 = \pi/2$ and $\theta_3 = \pi/4$ for $N = 3$ and $\theta_1 = 4\pi/5$, $\theta_2 = 3\pi/5$, $\theta_3 = 2\pi/5$ and $\theta_4 = \pi/5$ for $N = 4$, where 0 corresponds to hard right and π to hard left. To find the direction clusters, the scatter plot was rastered using a radial grid with $0.5^\circ = 0.00873$ rad resolution.

The estimation performance of the mixing matrix can be measured by the angular error e_{ang} between the original directions \mathbf{a}_n and their predictions $\hat{\mathbf{a}}_n$, averaged across each source and across each experiment. Also, the percentage of experiments in which the correct number of sources were detected (detection rate, DR) will be given as an additional measure of detection robustness.

The resulting values for DR and e_{ang} are shown in Table 3.4. For $N = 3$, the DR does not improve significantly, but e_{ang} has been almost halved. The $N = 4$ problem is more difficult, as expected, but the performance difference between the warped representations and the STFT has increased. In particular, the angular error has been reduced by a factor of 4 with the ERB and Bark warpings. The maximum improvement in average DR of the warpings was of 5% compared to the STFT.

3.5 Frequency-warped source estimation

The performance criteria reviewed until now (source sparsity, mixture disjointness and mixing matrix estimation accuracy) will all affect the last and definitive criterion: the quality of the separated signals. In this final section, the shortest path resynthesis algorithm will be introduced, followed by a discussion of a powerful and flexible set of criteria for separation quality evaluation. Finally, the experimental results will be presented.

3.5.1 Shortest path resynthesis

Source estimation for a given estimated mixing matrix $\hat{\mathbf{A}}$ will be formulated as an ℓ_1 -norm minimization problem (Sect. 2.7.2). For the noiseless case, and assuming

the source densities are Laplacian (Eq. 2.29), the formulation was given in Eq. 2.86. When the piecewise separation model holds (Eq. 3.7), Eq. 2.86 can be rewritten in the transformed domain as

$$\hat{\mathbf{s}}_{rk} = \underset{\mathbf{x}_{rk} = \hat{\mathbf{A}}\mathbf{s}_{rk}}{\operatorname{argmin}} \left\{ \sum_{n=1}^N |s_{n,rk}| \right\}. \quad (3.41)$$

The piecewise linear mixing model can be rewritten as

$$\mathbf{x}_{rk} = \sum_{n=1}^N \mathbf{a}_n s_{n,rk}. \quad (3.42)$$

Geometrically, this corresponds to each mixture data point being contributed by the projections of \mathbf{s}_{rk} upon the mixing directions or, in other words, by a set of N segments of length $|s_{n,rk}|$ along the vectors \mathbf{a}_n . Thus, minimizing ℓ_1 amounts to finding the shortest geometric path between each data point and the origin along the mixing directions, hence the alternative name *shortest path algorithm*.

As shown in Fig. 3.10, the shortest path to the origin for the stereo case ($M=2$) is found by projecting each data point upon the two mixing directions enclosing it. More specifically, the method partitions the mixture space \mathbb{R}^2 into regions delimited by the mixing directions \mathbf{a}_n . Then, for each bin \mathbf{x}_{rk} at direction θ_{rk} , a 2×2 reduced mixing matrix $\hat{\mathbf{A}}_\rho = [\hat{\mathbf{a}}_a, \hat{\mathbf{a}}_b]$ is defined, whose columns are the delimiting directions of the region it belongs to, i.e., $\theta_L = \arctan(a_{a2}/a_{a1})$ and $\theta_R = \arctan(a_{b2}/a_{b1})$ are the closest mixing directions to the left and to the right, respectively, that enclose θ_{rk} . Source estimation is performed by inverting the determined 2×2 sub-problem and setting all other $N - M$ sources to zero:

$$\begin{cases} \hat{s}_{\rho,rk} &= \mathbf{A}_\rho^{-1} \mathbf{x}_{rk} \\ \hat{s}_n &= 0, \quad \forall n \neq a, b. \end{cases} \quad (3.43)$$

The zeroing of the source contributions $\forall n \neq a, b$ makes the problem solvable, but as a trade-off, amounts to the introduction of artificial zeros to the time–frequency representation. This will cause spectral artifacts than can be audible as a *bubbling* noise, sometimes called in this context *musical noise*.

3.5.2 Measurement of separation quality

An obvious method to evaluate the separation quality, assuming the original sources are known, is to measure the ratio between the estimation error and the corresponding original source (*Signal to Error Ratio*, SER):

$$\text{SER}_n = 10 \log_{10} \frac{\|s_n\|^2}{\|\hat{s}_n - s_n\|^2}. \quad (3.44)$$

The SER is an overall measure of all distortions and errors introduced in the process, including errors by interference with the undesired signals, artifacts introduced by the separation algorithm and distortion due to imperfect transform inversion.

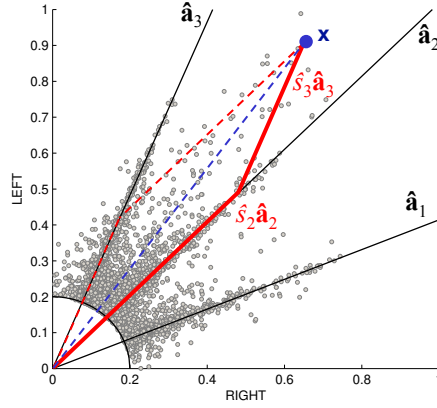


Figure 3.10: Shortest path resynthesis.

Recently, an alternative, more powerful evaluation method has been proposed by Gribonval *et al.* [163, 69], that allows gaining insight into the individual error sources in the course of the algorithm. For this reason, it was the method used here. In the noiseless case, it consists of decomposing each estimated source \hat{s}_n as the sum

$$\hat{s}_n = s_{\text{target}} + e_{\text{interf}} + e_{\text{artif}}, \quad (3.45)$$

where s_{target} is an allowed distortion of s_n (in the present case, a gain factor), e_{interf} is the error due to interferences with the other sources and e_{artif} is the error due to separation artifacts (which in this case are caused by the artificial zeros introduced by Eq. 3.43). Furthermore, the method achieves such a decomposition without knowledge of the mixing process, i.e., of \mathbf{A} . This is achieved as follows. The allowed distortion is given by the orthogonal projection

$$s_{\text{target}} = \frac{\langle \hat{s}_n, s_n \rangle s_n}{\|s_n\|^2}. \quad (3.46)$$

The error due to interferences is obtained via

$$e_{\text{interf}} = \mathbf{d}^H \mathbf{S} - s_{\text{target}}, \quad (3.47)$$

and the coefficient vector \mathbf{d} is given by

$$\mathbf{d} = \mathbf{G}_{\text{ss}}^{-1} [\langle \hat{s}_n, s_1 \rangle, \dots, \langle \hat{s}_n, s_N \rangle]^H, \quad (3.48)$$

where \mathbf{G}_{ss} is the Gram matrix of the original sources, whose elements are the mutual scalar products of all possible combinations of two source signals:

$$(\mathbf{G}_{\text{ss}})_{ij} = \langle s_i, s_j \rangle. \quad (3.49)$$

The remaining errors are assumed to arise from the artifacts introduced by the algorithm:

$$e_{\text{artif}} = \hat{s}_n - s_{\text{target}} - e_{\text{interf}} = \hat{s}_n - \mathbf{d}^H \mathbf{S}. \quad (3.50)$$

Repr.	$N = 3$ sources			$N = 4$ sources		
	SDR (dB)	SIR (dB)	SAR (dB)	SDR (dB)	SIR (dB)	SAR (dB)
STFT	16.47	31.39	16.74	10.41	23.81	10.81
CQ	18.23	31.71	18.59	12.77	24.75	13.42
ERB	18.46	31.56	18.83	12.98	24.71	13.53
Bark	18.02	31.46	18.36	12.79	24.58	13.34
Mel	18.54	31.41	18.92	13.00	24.62	13.57

Table 3.5: Evaluation of the source resynthesis stage: maximum achieved SDR, SIR and SAR for stereo mixtures of $N = 3$ (left) and $N = 4$ sources (right).

Once such decomposition has been performed, the following objective measures can be defined. The *Source to Distortion Ratio* (SDR), which plays the role of the global error measure SER, is defined based on the total error produced:

$$\text{SDR}_n = 10 \log_{10} \frac{\|s_{\text{target}}\|^2}{\|e_{\text{interf}} + e_{\text{artif}}\|^2}. \quad (3.51)$$

The *Source to Interference Ratio* (SIR) is

$$\text{SIR}_n = 10 \log_{10} \frac{\|s_{\text{target}}\|^2}{\|e_{\text{interf}}\|^2}. \quad (3.52)$$

Note that this SIR is a redefinition of the measure defined in Eq. 3.34 in the context of time–frequency masking, and both quantities are conceptually equivalent. Finally, the *Source to Artifacts Ratio* (SAR) is given by

$$\text{SAR}_n = 10 \log_{10} \frac{\|s_{\text{target}} + e_{\text{interf}}\|^2}{\|e_{\text{artif}}\|^2}. \quad (3.53)$$

The artifacts introduced by the separation algorithm are often the main cause of distortion in source separation, which justifies the utility of a separate SAR measure.

3.5.3 Evaluation with frequency-warped representations

The same 800 mixtures used in the evaluation of the mixing matrix estimation stage were subjected to shortest path resynthesis, given the estimated matrix obtained in that stage. Only those mixtures for which the correct number of sources was detected were forwarded to the resynthesis block. For each experiment run, the averaged values of SDR, SIR and SAR across all sources, and for band numbers $K_0 = 33$ to $K_{P-1} = 4097$ were computed.

The results are shown in Fig. 3.11 for the 3-source mixtures and in Fig. 3.12 for the 4-source mixtures. Table 3.5 shows the maximum values achieved. The improvement of the warped representations over the STFT is clear for the SDR and SAR measures. As expected, the unmixing of 4 sources gets worse quality values; the gain of using warpings however increases. The SIR curves show a different behaviour. While for 4 sources the improvement, though smaller than for SDR and SAR, is noticeable, for the 3-source case the maximum value achieved (with 1025

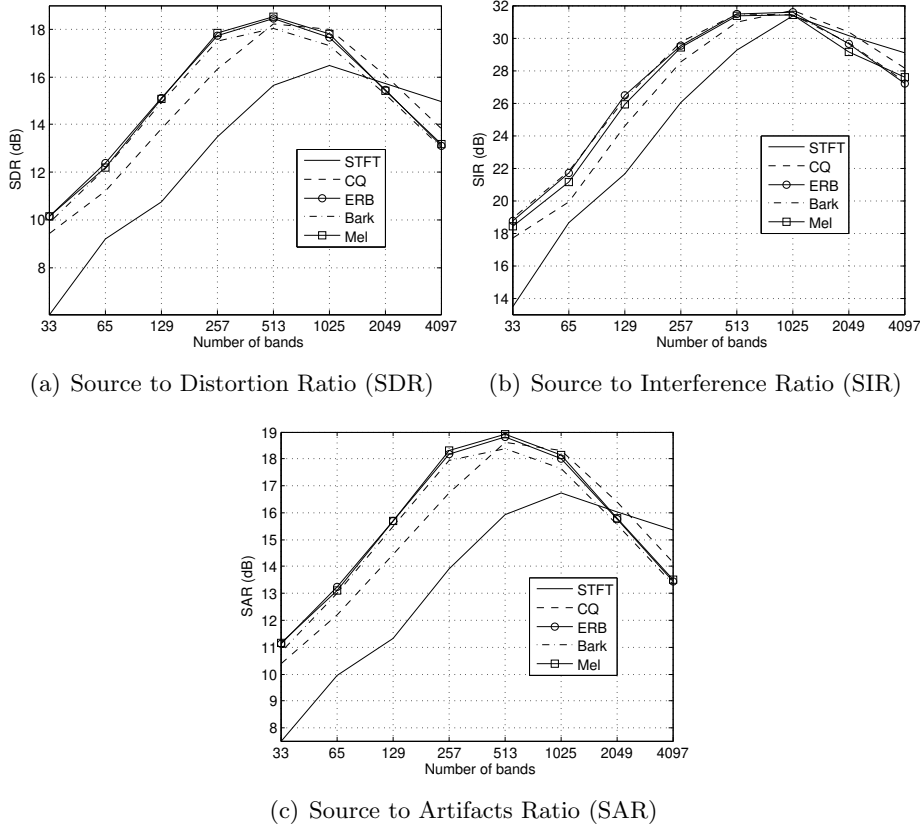


Figure 3.11: Evaluation of the source resynthesis stage: SDR, SIR and SAR as a function of number of subbands L , for stereo mixtures of $N = 3$ sources..

bands) is statistically equivalent for all representations. Overall, SIR is the least improved measure. In other words, with high frequency resolutions, warping mostly improves the results by reducing the effect of artifacts; the improvement by reducing interferences plays, in comparison, a smaller role. For medium and low frequency resolutions, frequency warping reduces both interferences and the effect of artifacts.

This observation can be explained by the fact that the SIR measure is the one most related to the concept of disjointness; although approached from a different perspective, both constitute a measure of the degree of overlapping. Recalling Fig. 3.7, it can be observed that for the music mixtures both the $\overline{\text{WDO}}$ and the SIR curves follow similar behaviours: the improvement of warping is clear for highly overlapping sources, and gets small with highly disjoint sources. In contrast with $\overline{\text{WDO}}$, the SIR measure is algorithm-dependent, and is influenced by the errors in mixing matrix estimation. Their goal is however similar, and thus a correlation between both results is plausible. Such a correlation is also readily observable from Eq. 3.36: when SIR_n increases, WDO_n increases, and vice versa.

An explanation for the reduction of the artifact distortion achieved with the

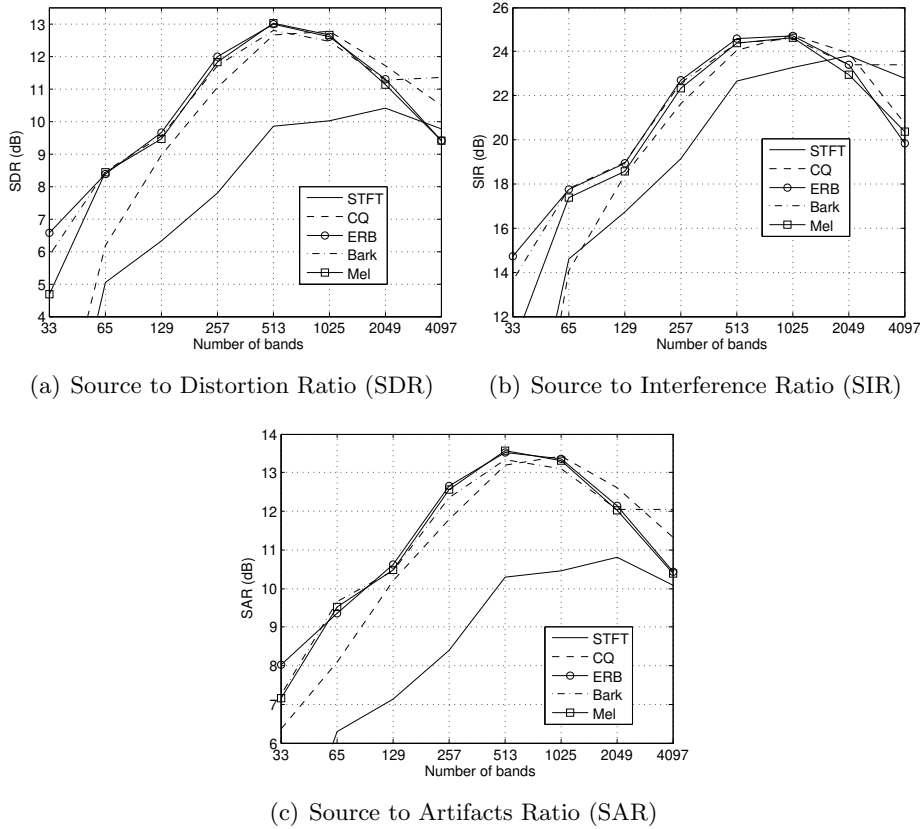


Figure 3.12: Evaluation of the source resynthesis stage: SDR, SIR and SAR as a function of number of subbands L , for stereo mixtures of $N = 4$ sources..

warped representations is that with STFT, the frames of all bands are synchronous, and thus the spectral zeros change all at the same time, increasing the measurable effect of the artifacts. With the warped representations, however, the temporal boundaries of the zeros are different in each band, and thus the instants of the spectral changes to zero are spread across time and the effect becomes less noticeable.

In general, all three auditory warpings (ERB, Bark and Mel) showed similar behaviors, with Mel obtaining a slightly better performance in both measures, followed by ERB and Bark. However, the overall difference with CQ is greater. As can be seen in Fig. 3.2, CQ is the transformation offering the highest frequency resolution in the low frequency area. It turns out that a more balanced trade-off between low frequency and high frequency resolution, as offered by the auditory warpings (whose warping curves lie between those of CQ and STFT) is more advantageous for the purpose of source separation.

3.6 Summary of conclusions

This chapter provided a thorough evaluation of the effect of using warped representations with nonuniform time–frequency resolution as the transformation stage within an underdetermined source separation framework. The representations were evaluated both under algorithm-independent conditions (by measuring source sparsity with normalized kurtosis and mixture disjointness with average W-Disjoint Orthogonality, $\overline{\text{WDO}}$), and in combination with a practical separation system based on kernel clustering and ℓ_1 -norm minimization (by measuring the accuracy of the mixing matrix estimation and the quality of the separated signals). Also, the special characteristics of music signals, together with their implications for the separation problem, were compared to those of speech signals. The conclusions drawn from the experiments can be summarized as follows:

- Speech is most sparse and most disjoint for a balanced trade-off between time and frequency resolution, with around $K = f_s/2^5$ bands. For music signals however, frequency resolution needs to be favored. In general, music is more sparse and disjoint at high frequency resolutions than speech at the optimal trade-off point.
- Frequency warping improves sparsity in comparison to the STFT. The maximum improvement was of 66.3% for speech and of 49.1% for music.
- Frequency warping also improves disjointness. The improvement is higher the more the sources overlap (i.e., in the low frequency resolution area for music signals and in both the lowest and highest frequency resolution areas at both ends of the optimal point for speech signals). For uncorrelated music mixtures, the improvement is of around 5-10% $\overline{\text{WDO}}$ for low frequency resolutions and of maximally 1.2% $\overline{\text{WDO}}$ for high frequency resolutions. For correlated music mixtures, the improvement is of 10-15% $\overline{\text{WDO}}$ in the low frequency resolution area; for high frequency resolutions, the disjointness of the STFT and of warpings are equivalent for music mixtures. With speech, the improvement is clear at all resolutions, achieving a highest difference of 5.2% $\overline{\text{WDO}}$.
- Within a practical application scenario with music mixtures, warping reduces the angular error when estimating the mixing directions, since proportionally more data points concentrate around the true mixing directions. Warpings reduce average angular errors by a factor of 2 for mixtures of 3 sources and by a factor of 4 for mixtures of 4 sources. The source detection rate (the percentage of the experiments in which the correct number of sources was detected) also improved, however less significantly, by an average of 5%.
- The separation quality was measured with respect to interference errors (SIR), artifacts errors (SAR) and overall distortion (SDR). The improvement due to the reduction of artifact errors is more important than the improvement due to the reduction of interference errors. In general, the improvement is the higher the more underdetermined the mixture is. For 3 sources, the maximum overall

maximum distortion improvement was of 2.07 dB SDR and for 4 sources of 2.59 dB SDR. The reduced effect of the artifacts is due to the between-band asynchronicity of the spectral zeros.

- From the improvement in SAR being higher than the improvement in SIR follows that maximizing sparsity and disjointness is not the only criterion that should be taken into account within a practical separation context. In fact, the experiments showed that separation quality can decay even if the overall sparsity/disjointness is maximal. Thus, it is important to acknowledge and evaluate the particular effect of the spectral artifacts introduced by time–frequency–masking-like algorithms, and to choose a balance between sparsity maximization and artifact error reduction.
- The ERB, Bark and Mel auditory warpings generally show a very similar behaviour, the results with Mel being slightly better. The CQ transformation globally performs worse, especially for low frequency resolutions. Consequently, an auditory-related distribution of time–frequency resolution, which is more balanced between the low and high resolution areas than the perfectly logarithmic resolution of the CQ, is generally more adequate for source separation.

These observations support the convenience of performing source separation in a frequency-warped domain. The results show that it is possible to improve the performance by only changing the transformation stage of the general staged separation approach of Fig. 2.8.

Obviously, this is just one of the possibilities to approach better performance, and further improvements are possible by studying other parts of the process. A crucial observation in this respect is that if two signal components, for example partial tracks in a tonal music mixture, overlap in exactly the same or in a very narrow time–frequency–space region, they will be impossible to separate, no matter how much warping is applied to the mixture. Since frequency-domain overlapping is especially significant for music signals, this will potentially pose a problem when separating in the spectral domain, and in highly correlated mixtures, warping will only be able to improve performance up to a certain degree.

A possibility to overcome this problem is to use a more sophisticated model of the sources, which equivalently means to increase the a priori information. The separation performed by the system discussed in this chapter was, according to the usual definition, completely blind: it was solely based on spatial information and on a broad sparsity assumption on the sources (a Laplacian distribution). In order to further improve performance and robustness, knowledge about the nature of the sources can be added. This can take the form of source-dependent models of spectral content or temporal structure. For the case of music mixtures, this can be achieved by exploiting the very specific timbral characteristics associated with different musical instruments. The development of such a model will be the topic of the following chapter.

4

Source modeling for musical instruments

The main goal of audio source separation in the time–frequency domain, when applied to harmonic or quasi-harmonic sounds, is to segregate the overlapping sinusoidal peaks. Algorithms exploiting only spatial information fail if the problem is underdetermined and highly overlapping. As discussed in the previous chapter, frequency-warping the representation front-end helps to improve disjointness and reduce artifact errors, but it will not help if two partials fall on exactly the same frequency, or on very narrow frequency margins. To overcome this and improve separation quality, the high generality of BSS algorithms (i.e., their complete “blindness”) can be sacrificed in favor of a higher degree of a priori knowledge about the nature of the signals that are expected to appear in the mixture. Methods following this approach are said to perform *Semi-Blind* Source Separation (SBSS) [162, 164].

Such a priori information results in source-specific models providing more detailed temporal and spectral information than the general statistical assumption of sparsity. Source modeling can consist of either developing a detailed structural description or probabilistic framework of the time–frequency events that can constitute the sources, or on training a statistical model based on a database of previously available source examples. These two distinctive methods are called, respectively, unsupervised and supervised source modeling approaches.

The unmixing of musical signals calls for the exploitation of the very specific temporal, spectral and statistical characteristics of sounds produced by musical instruments. *Timbre* is the musicological term employed to denote the perceptual qualities that enable the listener to distinguish between different instruments playing the same notes with the same dynamics. In contrast to the other sound quality attributes (pitch, intensity, duration), timbre is a concept more difficult to describe objectively, and is not univocally associated with an easily measured physical quantity. Instead, several factors play important roles in timbral perception, such as the temporal and spectral envelopes, the degree of harmonicity of the partials, noise content or *transients*¹. Any source modeling approach aiming at a description of any of those aspects, or a combination thereof, will consequently also be a model of timbre.

This chapter addresses the development of a novel timbre modeling approach for musical instruments from a general-purpose point of view. Although the main mo-

¹Transients are the pulse-like, non-harmonic segments occurring in the attack phase of a note, or in the transitions between consecutive notes.

tivation was to provide source knowledge for source separation, it was intended that the models could also be used as a feature in classification or recognition applications. Evaluation experiments of the models applied to classification and polyphonic instrument recognition tasks will be reported later in the chapter. Their application within a source separation framework will be discussed in more detail in Chapters 5 and 6. Correspondingly, previous work related with the development of general-purpose timbral descriptions will be introduced in this chapter (Sect. 4.3), whereas models specifically designed for source separation will be introduced in Sect. 5.1 of the next chapter. Parts of this chapter were previously published in [28], [29] and [105].

Requirements on the model

The following design criteria were followed and evaluated during the development process:

- **Generality.** The model should be able to handle unknown, real-world input signals. Thus, it must represent each musical instrument with enough generality to encompass the different qualities of, e.g., different violins or different pianos, or even of the same instrument at different pitch ranges. This requirement calls for a framework of database training and a consequent extraction of prototypes for each trained instrument. The presented modeling approach will consequently be supervised.
- **Compactness.** A compact model does not only result in more efficient computation and retrieval but, together with generality, implies that it has captured the essential characteristics of the source.
- **Accuracy.** In the source separation context, a timbre model will serve as a mask or template guiding the unmixing of overlapping partials. This requires a high representation accuracy, since the presence of artifacts resulting from deviations in the spectral masking process is the factor that most reduces perceptual separation quality, such as noted in the previous chapter and in works like [163]. Model accuracy is a demanding requirement that is not always necessary in other applications such as classification or retrieval by similarity, where the goal is to extract global, discriminative features.

The next two sections will introduce two concepts that will be central in the development of the timbre models: the spectral envelope (Sect. 4.1) and sinusoidal modeling (Sect. 4.2).

4.1 The spectral envelope

From the mentioned factors that contribute to timbre (envelopes, harmonicity, noise, transients), the temporal and spectral envelopes are two of the most important ones. In many situations, they arguably play the major role. Assuming that the expected

musical instruments are harmonic (winds, bowed strings) or quasi-harmonic (piano), harmonicity will not be a highly discriminative feature. Also, recent studies have noted the importance of the sinusoidal part compared to the noise part for the purpose of instrument recognition, such as in the work by Livshin and Rodet [101], where a 90.53% classification rate was obtained using only the sinusoidal part, only 4.36% less than using the original signals including sinusoids and noise. The first design decision was thus to base the model on the temporal and spectral envelopes.

The temporal envelope is usually divided into Attack, Decay, Sustain and Release phases, and is therefore often called the ADSR envelope. ADSR characteristics will be a valuable feature to distinguish, for instance, between sustained (bowed strings, wind instruments) and constantly decaying instruments (plucked or struck strings).

The spectral envelope can be defined as a smooth function of frequency that approximately matches the individual partial peaks of each spectrum frame. The frame-wise evolution of the partial amplitudes, and consequently of the spectral envelope, corresponds, when considered globally, to the temporal envelope. Thus, considering the spectral envelope and its evolution in time makes it unnecessary to consider the time-domain envelope separately. It even provides more detailed descriptions, since the time envelopes corresponding to the individual partials can be treated separately. In this work, the term “spectral envelope” will denote both the frame-wise spectral envelope and its dynamic temporal evolution and thus, the temporal envelope implicitly.

Apart from the temporal variations that occur during the course of a single note, the spectral envelope can be greatly dependent on dynamics. For example, playing the same note louder can excite upper partials that were previously masked by the noise floor, changing the envelope. It is well-known that many acoustic instruments exhibit fairly different timbres when played with different dynamics. Examples of instruments with great dynamic-dependency of timbre include the French horn, the clarinet and the piano. Less dynamic-dependent is the timbre of, for example, the oboe, the trumpet, a distorted electric guitar or the harpsichord, the latter having in effect virtually no dynamic range.

Some instruments feature formants in a similar way as the human voice does, produced by the resonances of the resonating body. Furthermore, physical characteristics of the instrument can greatly constrain the nature of the partials (e.g., the fact that the clarinet contains a cylindrical tube closed at one end causes the odd harmonics to predominate). Formant- or resonance-like spectral features can either lie at the same frequency, irrespective of the pitch, or be correlated with the fundamental frequency. In this work, the former will be referred to as f_0 -invariant features, and the latter as f_0 -correlated features. Formants that are f_0 -invariant motivate that the same instrument can have fairly different timbres in the lower and upper pitch ranges.

Other modifications, such as different playing styles or effects (e.g., *pizzicati* — plucking the string with the fingers—, *sul ponticello* —bowing near the bridge— or playing harmonics) or the application of external sound modifiers (dampers, mutes, prepared pianos, analog or digital effects) will obviously have a direct influence on the envelope and on timbre. Reverberation will also modify the perceived envelope, but

it will not be considered an intrinsic timbral feature, since it depends on the listener's position and on the characteristics of the room, and is furthermore irrelevant in the instantaneous mixing scenario that is being considered throughout the present work.

The source–filter model of sound production

The source–filter model [6] provides a simple but powerful approximation to the generation of sounds whose timbre is highly determined by a spectral envelope, such as speech or music signals. It assumes that such sounds are produced by an excitation signal with a flat spectrum (usually white noise or a set of harmonic partials with constant amplitude) that is subsequently filtered with a frequency response resembling the spectral envelope. Thus, it decouples the contribution of the spectral envelope (produced by filtering due to the properties of the resonating body, in musical instruments, or of the vocal tract for speech or singing) from that of the original raw sound material (produced by the vibrating sound generator, e.g., the string in a guitar or the vocal cords in human voice).

The source–filter model differs from reality in that it collects all spectral modifications into the filtering stage, whereas in real situations the vibrating source already exhibits some spectral coloring. For most modeling applications such a distinction is unnecessary.

Next, some popular methods for the estimation of the spectral envelope will be introduced. They can be divided into three groups: methods based on assuming an *autoregressive* (AR) model of the signal to be analyzed, methods relying on *cepstral smoothing*, and methods relying on *interpolation*. All methods operate on a frame-wise basis. A comparative study of spectral envelope estimation methods can be found in the work by Schwarz [137].

Autoregressive methods for spectral envelope estimation

An AR signal model assumes that the observed signal $s(t)$ has been produced according to

$$s(t) = \sum_{n=1}^N a_n s(t-n) + u(t), \quad (4.1)$$

where $u(t)$ is the input signal, a_n are the AR coefficients and N is the model order. Thus, each output sample can be predicted by a linear combination of the N previous samples, as acknowledged by its alternative designation of *linear predictive* model. Under certain circumstances, the AR coefficients are able to sufficiently approximate the signal, so that a high compression is possible by transmitting only the coefficients and the excitation (this is called *Linear Predictive Coding*, LPC). The optimal coefficients in the MSE sense can be obtained by a variety of approaches, such as the Levinson-Durbin or Burg algorithms. Such techniques have become classical signal processing methods, and are widely documented in the literature (see, e.g., [127]).

A spectral analysis of the AR model reveals a direct relationship with the source–filter production model. In particular, the AR coefficients are also the coefficients of an N -th order all-pole filter whose frequency response approximates the spectral envelope. Thus, linear prediction corresponds to filtering the excitation signal $u(t)$ with the obtained all-pole filter. A disadvantage of the LPC method is that for high model orders, and if the partials are sufficiently separated, the estimated envelope tends to “wrap” the partials too closely, and to “fall” towards the noise floor, as has been pointed out by Schwarz [137].

The all-pole approximation is optimal if the excitation signal $u(t)$ is assumed to be white, uncorrelated noise, but was shown to be suboptimal for periodic or quasi-periodic excitations, which are more interesting for analyzing pitched instruments, as noted by El-Jaroudi and Makhoul [56]. In the cited work, an alternative LPC-based method was proposed, called the *Discrete All-Pole* (DAP) method, which is based on sampling the input spectrum at the partial peaks, and modifying the error criterion accordingly. This was shown to reduce the systematic estimation errors of classic LPC when analyzing harmonic sounds, while keeping a good performance for noisy excitations. As disadvantages of the DAP method, difficulty in finding the optimal model order and high computational requirements have been pointed out [159].

Cepstral methods

An alternative family of methods arises from the concept of *cepstrum*, which in the real-valued, discrete version is defined as

$$c(q) = \frac{1}{K} \sum_{k=0}^{K-1} \log(|S(k)|) e^{j \frac{2\pi}{K} kq}, \quad (4.2)$$

where $S(k)$ is the DFT of the input signal $s(t)$. The cepstrum is thus the inverse DFT of the log-amplitude spectrum. The magnitude in the cepstral domain, indexed by q , is called *quefrequency*. Some authors define the cepstrum as the direct DFT of the log amplitude spectrum, instead of the inverse transform. Since the quantity $\log(|X(k)|)$ is always real and even for real signals $x(t)$, both definitions are equivalent.

The cepstrum can be interpreted as a spectrum of the spectrum, i.e., the original spectrum is considered as a signal and its “frequency” contents is analyzed. An advantage of cepstral processing is that filtering, which is a convolution in the time domain, and a multiplication in the spectral domain, turns into a sum in the cepstral domain. In much the same way that low-pass filtering a time domain signal results in a smoothing of the fast amplitude changes, keeping the low cepstral coefficients smoothes the spectrum, i.e., produces a spectral envelope. For a signal following the source–filter model, the low quefrequencies will correspond to the spectral envelope and the high quefrequencies to the detailed structure of the spectrum. Harmonic signals will show a cepstral peak corresponding to the fundamental frequency. Thus, by taking the $L < N$ lowest cepstral coefficients it is possible to isolate an approximate spectral envelope (the generation filter) from the excitation (source) signal. The

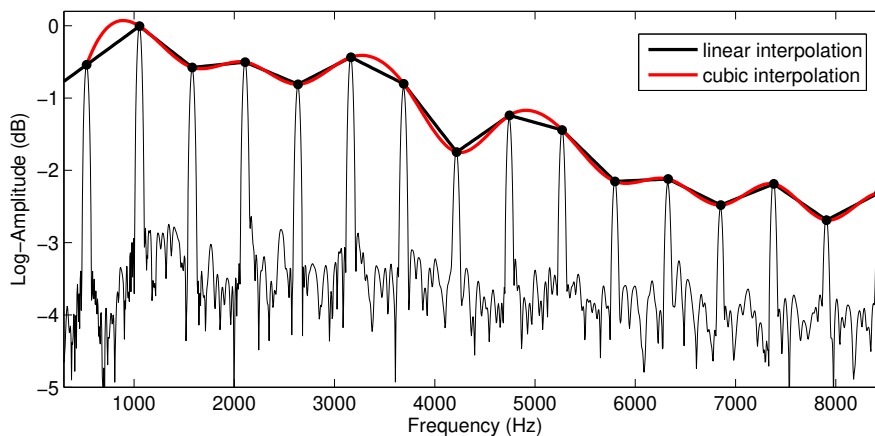


Figure 4.1: Comparison between linear and cubic interpolation for spectral envelope estimation.

action of keeping these low coefficients is called *cepstral smoothing* or *liftering* (in analogy to filtering), and L is called the *smoothing order*. In this work, cepstral smoothing will be used to improve the robustness of sinusoidal peak picking (see Sect. 4.2).

In [137], two disadvantages of cepstral smoothing are noted: the fact that it produces an averaged envelope, rather than an envelope closely matching the peaks, and that, like LPC, it tends to the noise level in between-peak valleys. Following a similar reasoning as for the DAP approach, this can be avoided by picking the prominent peaks of the spectrum. To that end, the *Discrete Cepstrum* method by Galas and Rodet [64] selects the partial peaks and defines an MSE minimization problem thereupon. It has however been criticized by Cappé and Moulines [34] as being ill-conditioned in many cases of interest.

An approach that tackles both the averaging and the “valley” problems is the *True Envelope* estimation method, presented by Röbel and Rodet [131], which uses a simple but effective rule to iteratively update the cepstral-smoothed envelope. It has the advantage over Discrete Cepstrum that it does not rely on a previous peak picking stage.

Interpolation methods

Interpolation-based spectral envelope estimation consists in selecting the prominent sinusoidal peaks and defining a function between them by interpolation. The most basic form of interpolation is linear interpolation, in which a straight line is traced between each two neighboring peaks. The amplitude A between peaks at frequencies f_0 and f_1 is thus given by

$$A(f) = A(f_0) + \frac{A(f_1) - A(f_0)}{f_1 - f_0}(f - f_0). \quad (4.3)$$

Cubic polynomial interpolation consists in finding a smooth function defined by the polynomial

$$A(f) = a_0 + a_1f + a_2f^2 + a_3f^3 \quad (4.4)$$

that passes through the peaks. An illustration of linear and cubic interpolation is given in Fig. 4.1 for a DFT frame of an oboe playing a C5 note. Linear interpolation results in a piecewise linear envelope containing edges. In spite of its simplicity and roughness, it has proven adequate for several applications, such as envelope-preserving pitch shifting, spectral shape shifting and residual shape modeling [4]. Cubic interpolation results in a smooth curve, but is much more computationally expensive. For its effectiveness, simplicity and flexibility, the interpolation approach was chosen in the present work as the envelope estimation method.

4.2 Sinusoidal modeling

All envelope estimation methods relying on the sampling of the spectrum at the partial frequencies (DAP, Discrete Cepstrum, interpolation methods) need a way of accurately and robustly detecting the amplitude peaks at each frame of the STFT. Note that, under the general signal expansion framework introduced in Sect. 2.3.1, this would correspond to frame-wise decomposing the signal as a sum of sinusoids with the frequency and amplitude values of the detected peaks, thus ignoring the noise floor. It can therefore be understood as a generalization of a Fourier Series, in which the sinusoids do not need to be in harmonic relationship.

When considering temporal evolution, each detected peak will change in amplitude and frequency, building a *partial track*. Following the same reasoning, this can be viewed as a generalization of the STFT in which the expansion sinusoids can vary in frequency from frame to frame. The collection of partial tracks is called the *sinusoidal part* of the signal, and its estimation is the goal of *sinusoidal modeling* [65, 138, 146], also called *additive analysis*. The remainder of the signal, the time-varying noise floor, is the *noise part*, *stochastic part* or *residual*.

The formulation of sinusoidal modeling as a signal decomposition problem is

$$s(t) \approx \hat{s}(t) = \sum_{p=1}^{P(t)} A_p(t) \cos \theta_p(t). \quad (4.5)$$

Here, $P(t)$ is the number of partials, possibly time-varying, $A_p(t)$ are their amplitudes and $\theta_p(t)$ is the instantaneous phase. Because the instantaneous frequency $f_p(t)$ is the derivative of the instantaneous phase, the unwrapped² phase is given by

$$\theta_p(t) = \theta(0) + 2\pi \sum_{\tau=0}^t f_p(\tau), \quad (4.6)$$

²*Unwrapped phase* refers to a phase whose value can exceed the interval $[0, 2\pi]$ or $[-\pi, \pi]$. An unwrapped phase is equivalent to a wrapped one constrained to those intervals, but it is analytically convenient for formulations such as Eq. 4.6.

where $\theta(0)$ is the initial phase.

In practice, additive analysis consists of performing a frame-wise approximation of this model, yielding a triplet of estimated amplitude, frequency and phase information

$$\hat{s}_{pr} = (\hat{A}_{pr}, \hat{f}_{pr}, \hat{\theta}_{pr}), \quad (4.7)$$

for each partial p and each time frame r . Additionally, the frame-wise number of partials P_r must also be approximated from $P(t)$. These approximations are implemented by the successive stages of STFT, peak picking and partial tracking, with an optional pitch detection stage at the beginning. All these tasks have been the subject of extensive research, and a range of well-established techniques [138, 146] are available³. It is beyond the scope of the present work to describe them in detail; instead, they will be briefly introduced conceptually.

Given a set of additive analysis data triplets, the spectral envelope can be estimated by frame-wise interpolating the amplitudes \hat{A}_{pr} at frequencies \hat{f}_{pr} for $p = 1, \dots, P_r$ and for each frame r . The set of frequency points \hat{f}_{pr} for all partials during a given number of frames is called *frequency support*. The phases $\hat{\theta}_{pr}$ will be ignored for the analysis applications presented in this work, but can be stored and reused for resynthesis.

Peak picking

Choosing adequate resolution parameters for the STFT is crucial to make a successful detection of prominent peaks. A sufficiently high frequency resolution should be chosen to avoid the overlapping of close peaks, such as suggested by the disjointness experiments of Chapter 3. Also, the frequency response of the analysis window determines the shape of the peaks, so window parameters such as main-lobe width and side-lobe attenuation must be taken into account. If the pitch of the signal is known beforehand, or if peak detection follows an initial optional pitch detection stage, the analysis parameters can be adapted to adequately accommodate the expected frequency components. Reasonable choices for a given expected fundamental frequency f_0 are a window length $L = 4f_0$ for Hamming windows and $L = 5f_0$ for Blackmann windows, with overlapping factors of 75% or 87.5%, in which cases the COLA condition (Eq. 3.26) is met. If the fundamental frequency is not known beforehand, a reasonable power-of-two window length is $L = 8192$ for a sampling rate $f_s = 44.1$ kHz.

Note that knowing the fundamental frequency does not only help to set the STFT parameters, but also to predict that the prominent peaks will appear at integer multiples of f_0 , improving the robustness of peak picking. This is also true for the next stage of partial tracking. In general, additive analysis can thus work in two different modes: harmonic, when the f_0 is known or extracted via pitch detection, or inharmonic, when the f_0 is unknown or cannot be reliably detected.

³This work uses IRCAM's implementation of an additive analysis/synthesis framework, called pm2.

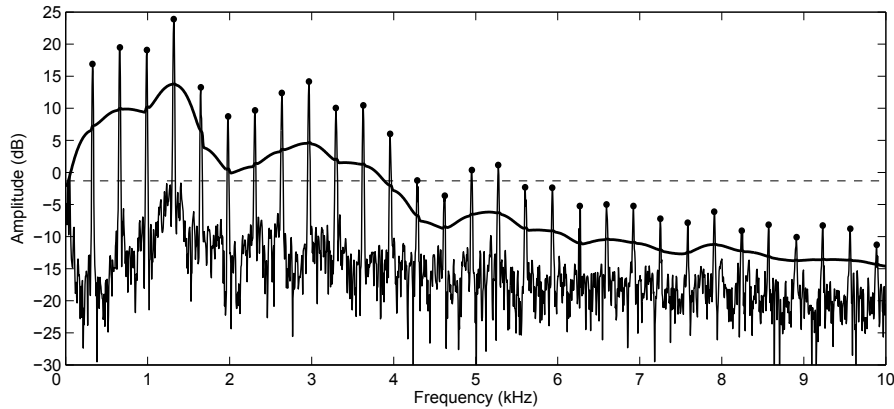


Figure 4.2: Frequency-dependent thresholding for peak picking.

In the present work, both analysis modes will be encountered. In the training procedure discussed later in this chapter, the notes belonging to the training database are previously labeled with their instrument, dynamics and pitch, and a harmonic analysis can thus be performed. For the classification of isolated instrument notes (Sect. 4.7), the pitch of the unknown notes was previously extracted using the autocorrelation-based YIN method by de Cheveigné and Kawahara [44].

However, whenever mixtures must be subjected to additive analysis, such as in Sect. 4.8 and in Chapter 5, pitch estimation is unreliable and for highly polyphonic mixtures even virtually impossible. In that case, inharmonic analysis must be used, which increases the possibility of detecting spurious or noisy peaks near the noise floor. To avoid that and increase the robustness of the extraction, an additional preprocessing step was added in the present approach, as suggested by Every and Szymanski [62], consisting of computing a rough approximation to the spectral envelope of each frame by simple cepstral smoothing. In particular, the spectrum is *lifted* by convolving a Hamming window with each log-amplitude DFT frame. The rough envelope is then scaled in amplitude so that it appropriately covers the noise floor, and is used as a frequency-dependent threshold for peak picking. This is illustrated in Fig. 4.2, where the solid line represents the obtained threshold and the points denote the detected peaks. Note that using a constant threshold minimized to cover the highest level of the noise floor, as showed by the dashed line, would result in missing most upper partials.

Finally, an important factor to note in peak detection is that the sampled nature of the STFT unavoidably introduces rounding errors when estimating frequency and amplitude values lying in the area between two frequency bins, separated by an interval of f_s/K Hz for a FFT of size K . It is possible to increase the detection resolution without increasing the size of the FFT by using one of a range of refinement methods based on interpolation⁴. As an example, Amatriain *et al.* [4] use parabolic

⁴Such a between-bin interpolation should not be confused with the between-peak interpolation for extracting the spectral envelope explained previously in the chapter.

amplitude interpolation of the frequency bins.

Partial tracking

Once the peaks have been detected, it is the purpose of the subsequent *partial tracking* or *peak continuation* block to trace the frame-to-frame evolution of each peak, resulting in a set of partial tracks or trajectories. Partial tracking relies on measuring the continuity of the sinusoids as measured by the frame-wise relative and absolute differences in amplitude and frequency. By observing a certain range of neighboring frames, the algorithm then decides when a partial trajectory (called *guide* in this context) starts or ends. The sensitivity to amplitude or frequency variations can be set up by appropriate weighting parameters. Partial following is often performed backwards, since it is easier to establish the most likely partial candidates in the more stable decay phase of the notes, rather than in the course of the noisy attack transients. The classical method for partial tracking is the McAulay–Quatieri algorithm [107].

In the developments of the present chapter, an explicit grouping of subsequent peaks into tracks is not needed. However, it is important to perform partial tracking anyway, since its continuation rules retroactively influence the decision of which peaks of the peak picking stage can be reliably associated to a partial, and which cannot. In the source separation methods that will be proposed in the next two chapters, such a track-wise grouping will be however crucial. To denote a track \mathbf{t}_t , the following notation will be used:

$$\mathbf{t}_t = \{\hat{s}_{pr_t} | r_t = 1, \dots, R_t\}, \quad (4.8)$$

where R_t is the track length in frames, r_t is the frame index relative to the first frame of the track, and \hat{s}_{pr_t} is the estimated parameter triplet of Eq. 4.7. Note that this definition implicitly assumes that each track is contributed by the same partial p during its whole length, which should always be the case. The previous considerations about harmonic and inharmonic analysis apply here as well. Partial tracking algorithms can be considered computational implementations of the ASA grouping principles of good continuation and common fate (see Sect. 2.8).

A graphical example of the detected frequency support of a set of partial tracks after peak picking and partial tracking can be found in Figs. 5.2(a) and 5.6 of the next chapter.

Resynthesis

Resynthesis of a time-domain signal from the sinusoidal parameters (*additive synthesis*) is straightforward and consists of driving a set of sinusoidal oscillators with the parameters and adding the resulting sinusoids. To avoid clicks at the frame boundaries, the parameters are usually interpolated between frames [138]. Amplitude is smoothed by linear interpolation, and instantaneous phase (the integral of the frequency) by cubic interpolation. This results in the sinusoidal or deterministic part of the signal.

Extensions to the sinusoidal model

To further improve the model and also to take into account the noise part, the sinusoidal part can be subtracted from the original signal, yielding the noise residual. This residual can be modeled by frame-wise fitting the noise spectrum to a certain filter frequency response, and keeping the coefficients as features. By means of this analysis, a powerful and flexible model of the signal is obtained, called *sinusoidal plus noise model*, which is the basis of the *Spectral Modeling Synthesis* (SMS) framework by Serra [138]. An extension thereof, presented by Verma *et al.* [157] called *Transient Modeling Synthesis* (TMS), uses an explicit model of the note transients.

4.3 Modeling timbre: previous work

In this section, previous approaches having as main goal a detailed and systematic extraction of timbral descriptions will be reviewed. Timbral and other sophisticated source models arising within a source separation context will be introduced later in Sect. 5.1 for mono and in Sect. 6.1 for stereo separation systems.

Probably the first attempt to thoroughly and systematically assess the factors that contribute to timbre was the 1977 work by Grey [67]. He conducted listening tests to judge perceptual similarity between pairs of instrumental sounds, and applied *Multidimensional Scaling* (MDS) to the results. MDS is a dimensionality reduction technique very similar to PCA that finds the most informative projections of the data by trying to preserve a given dissimilarity matrix (such as a matrix of Euclidean distances), instead of by trying to maximize variances, as PCA does. In the cited work, MDS was used to produce a three-dimensional *timbre space* where the individual instruments clustered according to the evaluated similarity. It was obtained that the first dimension was related to the spectral flatness, the second to the amount of synchronicity of the partials at the onsets and offsets of each note, and the third to the energy of the attack transient.

In later works, similar results were obtained by substituting the listening tests by objectively measured sound parameters. Hourdin, Charbonneau and Moussa [76] apply MDS to obtain a similar timbral characterization from the parameters obtained from sinusoidal modeling. They represent trajectories in timbre space corresponding to individual notes, and resynthesize them to evaluate the sound quality. It was obtained that, after MDS, 75% of information was needed for musically acceptable sounds, and 90% of the information for sounds indistinguishable from the original.

Sandell and Martens [134] use PCA as a method for data reduction of additive analysis/synthesis parameters. They performed hearing experiments to evaluate the compression efficiency of single notes. In this case, depending on the instrument, a 40-70% data reduction was obtained for nearly identically-sounding resynthesized sounds. De Poli and Prandoni [126] propose their *sonological models* for timbre characterization, which are based on applying PCA or Self Organizing Maps (SOM) to a description of the spectral envelope based on MFCCs. SOMs are neural networks trained to produce a low-dimensional representation that keeps the topological

properties of the original data. A similar procedure by Loureiro, de Paula and Yehia [102], which applies either PCA plus k-Means clustering or SOM directly to the sinusoidal parameters, has recently been used to perform clustering based on timbre similarity.

Jensen [81] develops a sophisticated framework for the perceptually meaningful parametrization of additive analysis parameters. Different sets of parameters are intended to describe in detail the spectral envelope, the mean frequencies, the temporal envelopes containing the ADSR segments plus an additional “End” segment, and amplitude and frequency irregularities (called, respectively, *shimmer* and *jitter*). Expressive additions such as vibrato and tremolo⁵ can be included as explicit extensions to the model. The intended main application is parameter-driven synthesis.

The fields of Music Information Retrieval and Music Content Analysis provide a huge diversity of spectral features. Most of them are basic measures of the spectral shape (centroid, flatness, energy rolloff, etc.), and are too simple and inaccurate to be considered timbre models as understood here. More sophisticated measures make use of psychoacoustical knowledge to produce a compact description of spectral shape. This is the case of the very popular MFCCs [50], which are based on a mel-warped filter bank and a cepstral smoothing and energy compaction stage achieved by a DCT. The MFCC algorithm can be considered a psychoacoustically-adapted cepstral smoothing for spectral envelope estimation. It has proven to be very efficient for speech and music description. As an example, Helén and Virtanen [71] combine MFCC extraction with a parametrization of the ADSR temporal envelopes of the individual coefficients; the intended application thereof is *Structured Audio Coding* (SAC). However, the poor frequency resolution of the MFCC filter bank, together with the fact that the envelope portions correspond to fixed frequency bands, and not to time-varying partials, makes them unsuitable for an accurate spectral envelope description. MFCCs will be addressed in more detail in Sect. 4.7.1, since they will be subjected to a comparative evaluation in the scope of the present work.

The MPEG-7 standard [80] makes use of spectral basis decomposition as feature extraction to train a statistical model [38]. However, the goal there is again the classification of general audio signals, and the extraction is based on an estimation of a rough overall spectral shape, defined as the energies in a set of fixed frequency bands. Although this shape feature is called Audio Spectrum Envelope in the standard, it is not a spectral envelope in the strict sense of matching the partial peaks, and so it does not accurately follow the peaks, either.

Tardieu and Rodet [152] propose a statistical model of instrumental timbre based on estimating GMMs from general-purpose features such as spectral flatness, noisiness, attack time and energy modulation. Each model is intended to capture the instrument’s whole pitch and dynamic ranges; however, different playing styles and effects require different models. In order to increase the per-class sample population, needed for robust training, they propose a factorization of the GMMs. In

⁵Although interrelated, it is often assumed that vibrato is only produced by frequency modulation, and tremolo by amplitude modulation.

this way, it possible to learn, e.g., an instrument-independent vibrato model and a playing-style-independent flute model, and afterwards combine both models to obtain a vibrato flute model, even if few or no samples belonging to that particular class were present in the database. The intended application is a very novel one: *Computer-Aided Orchestration* (CAO). Given a target sound, the goal is to propose a certain combination of acoustic orchestral instruments, and their respective pitches, dynamics and playing styles, so that, when combined, the resulting sound will resemble the original target sound.

Relatively few works have been devoted to discuss the limitations of traditional statistical models such as GMMs or *Hidden Markov Models* (HMM), or of vector-quantization clustering techniques, in which each input data vector is replaced by the corresponding cluster centroid (such as k-Means) when applied to individual instrumental samples. The issue here is that an accurate description of the temporal evolution of the spectral, or any other kind of short-time feature, is lost or severely reduced, and replaced by a discrete set or sequence of states or cluster centroids.

Within this context, Klapuri *et al.* [92] propose a state-based modeling method in which the between-state transitions do not occur instantly, such as in HMM and vector quantization, but during a given time interval at a constant speed, following a path defined by linear interpolation (and thus called *interpolating state model*). However, the method is completely deterministic and thus not appropriate to the statistical characterization of large databases, which is the purpose aimed at here. The proposed application in the cited work is to encode individual data sets for SAC purposes. The issue of accurate modeling of temporal evolution will play the central role in the prototyping (training) stage of the model proposed here (Sect. 4.6).

4.4 Developed model

The approaches reviewed above fulfill some of this work's design requirements of generality, compactness and accuracy, but none of them meets the three conditions at the same time. The modeling approach presented in the present chapter was motivated by the goal of combining all three advantages into the same algorithm. What follows is a detailed discussion of how the previous approaches fail to meet the criteria, and how those limitations are proposed to be overcome.

- **Generality.** Most of the reviewed methods are intended for the timbral characterization of individual notes, and do not propose a training procedure. Most often [76, 102, 126, 134], a few consecutive notes are indeed concatenated to obtain common bases of the timbre space; however, no method is proposed to summarize the projections of all notes from each instrument into a single prototype. MFCCs and the MPEG-7 approach are intended for large-scale training with common pattern recognition methods, but as mentioned they do not meet the requirement of accuracy of the envelope description. In the present work, a training procedure consisting of extracting common reduced-dimensional bases and describing each instrument's training database as a prototype curve in timbre space will be proposed (Sect. 4.6).

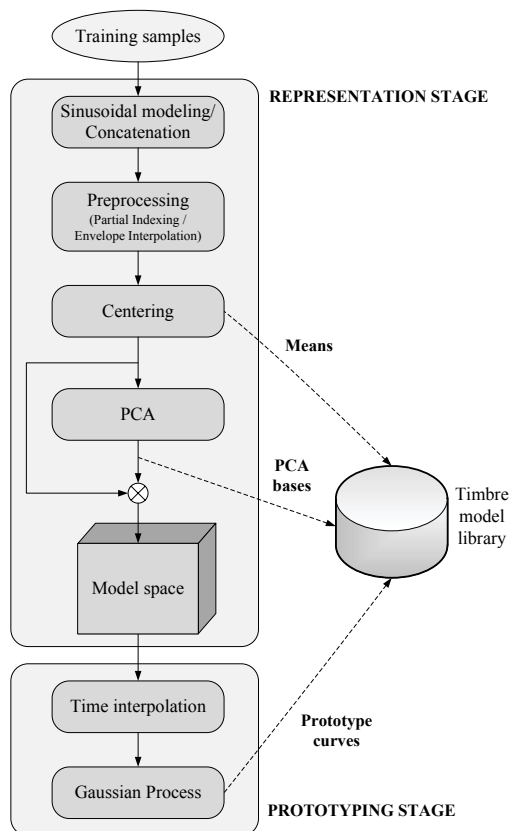


Figure 4.3: Overview of the timbre modeling process.

- **Compactness.** In [134], compactness was considered one of the goals, but no training phase takes place. MFCCs, used in [71, 126], are highly compact but, again, inaccurate. This work will use PCA-based spectral basis decomposition to attain compactness.
- **Accuracy.** All approaches relying on sinusoidal modeling [76, 102, 126, 134] are based on highly accurate spectral descriptions, but as mentioned, fail to fulfill either compactness or generality. The model used here relies on an accurate description of the spectral envelope by means of additive-analysis-based interpolation. Also, the dynamic evolution of timbre will be modeled in a more detailed way than can be obtained by traditional methods like GMMs and HMMs.

As will be discussed in detail, the fulfillment of the three criteria simultaneously creates additional challenges that need to be addressed, such as the misalignment of frequency supports (Sect. 4.5.2) and the preservation of a faithful description of the spectral envelope's temporal variation (Sect. 4.6). The modeling approach proposed here can be divided into a representation and a prototyping stage. In the context

of statistical pattern recognition, this corresponds to the traditional division into feature extraction and training stages. Figure 4.3 shows an overview diagram of the whole modeling approach. Each individual processing block will be addressed and evaluated in detail in the next two sections: Sect. 4.5 deals with the representation stage and Sect. 4.6 with the prototyping stage.

4.5 Representation stage

The aim of the representation stage of the modeling procedure is to produce a set of coefficients describing the individual training samples. It thus can be thought of as a feature extraction process such as found in classification or general pattern recognition applications. The process of summarizing all the coefficients belonging to an instrument into a prototype subset representative of that particular instrument will be the goal of the second modeling step, the prototyping or training stage, addressed in the next section.

The requirement of accuracy, however, makes the presented approach differ from traditional feature extraction methods, which in general lose lots of information and are not invertible (think for instance of the spectral shape features like centroid or flatness, which describe the whole spectrum by a single scalar). An appropriate balance between compactness and accuracy can be achieved by adaptive basis decomposition methods (introduced in Sect. 2.3) applied to the time–frequency spectrum.

4.5.1 Basis decomposition of spectral envelopes

The application of adaptive basis decomposition to time–frequency representations was proposed by Casey [38, 40] as a powerful feature extraction method that adequately characterizes a wide range of sounds. His approach has been widely popular and was adopted as part of the MPEG-7 standard [80]. Following the notation used for PCA in Sect. 2.3.4, spectral basis decomposition consists of performing a factorization of the form

$$\mathbf{X} = \mathbf{P}\mathbf{Y}, \quad (4.9)$$

where \mathbf{X} is the data matrix containing the original signal, \mathbf{P} is the transformation basis whose columns \mathbf{p}_i are the basis vectors, and \mathbf{Y} is the projected coefficient matrix. However, instead of considering a set of time-domain signals as the rows of matrix \mathbf{X} , as was done in Chapter 2, this time the input data matrix is a time–frequency representation constituted by a set of K spectral bands and R time frames (usually $R \gg K$).

As introduced in that chapter, the rows are considered the random variables, and the columns their realizations. Therefore, the interpretation of the decomposition defined by Eq. 4.9 will depend on the data organization on that matrix. If the matrix is in *temporal orientation* (i.e., it is a $R \times K$ matrix $\mathbf{X}(r, k)$), a temporal $R \times R$ basis matrix \mathbf{P} is obtained. If it is in *spectral orientation* ($K \times R$ matrix $\mathbf{X}(k, r)$), the result is a spectral basis of size $K \times K$. Having as goal the extraction of spectral

features, the latter case is of interest here. In spectral orientation, the projected coefficients \mathbf{Y} (i.e., the principal components) constitute a set of uncorrelated time-varying weights.

Using adaptive transforms like PCA (Sect. 2.3.4) or ICA (Sect. 2.6.1) for time–frequency decomposition has proven to yield valuable features for content analysis [38]. Since PCA yields an optimally compact representation, in the sense that the first few basis vectors represent most of the information contained in the original representation, while minimizing the reconstruction error, it is most appropriate as a method for dimensionality reduction. ICA additionally makes the transformed coefficients statistically independent. When applied to a time–frequency representation, ICA is called *Independent Subspace Analysis* (ISA), and its main application is source separation from mono mixtures [39] (see also Sect. 5.1). However, since the minimum reconstruction error is already achieved by PCA, ICA is not needed for the current compact representation purposes. This fact was confirmed by preliminary experiments. PCA was thus chosen for the present model.

In practice, the input data must be centered and it is convenient to whiten the output (Sect. 2.3.4) in order to balance the influence of each timbral axis. Thus, the final projection of reduced dimensionality $D < K$ is given by

$$\mathbf{Y}_\rho = \mathbf{\Lambda}_\rho^{-1/2} \mathbf{P}_\rho^T (\mathbf{X} - E\{\mathbf{X}\}), \quad (4.10)$$

where $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_D)$ and λ_d are the D largest eigenvalues of the covariance matrix $\mathbf{\Sigma}_\mathbf{x}$, whose corresponding eigenvectors are the columns of \mathbf{P}_ρ . The truncated model reconstruction will then yield the approximation

$$\hat{\mathbf{X}} = \mathbf{P}_\rho \mathbf{\Lambda}_\rho^{1/2} \mathbf{Y}_\rho + E\{\mathbf{X}\}. \quad (4.11)$$

The original approach by Casey performs basis decomposition upon the STFT spectrogram, with fixed frequency positions given by the regular frequency-domain sampling of the DFT. This is a substantial difference to the aim of the present work, which is to apply such decomposition methods on the (dynamic) spectral envelope. Since the spectral envelope is defined here as a set of partials with varying frequency supports plus an interpolation function, the arrangement into the data matrix is not so obvious and calls for several additional considerations. They will be addressed thoroughly in the next section.

Examples of spectral envelope decompositions of single notes

Figure 4.4 shows an example of PCA spectral basis decomposition performed upon the spectral envelope of a single violin note extracted by additive analysis and linear interpolation. Note that the surface is only defined at the time-varying frequency support of the sinusoids. Only the first 3 dimensions of the PCA decomposition are shown. The original data matrix (left), containing logarithmic amplitudes, is decomposed as the product of a spectral basis matrix (center) and of a coefficient matrix (right). Note that the first time coefficient function represents the overall temporal ADSR envelope, consisting in this case of a sustained phase and of a

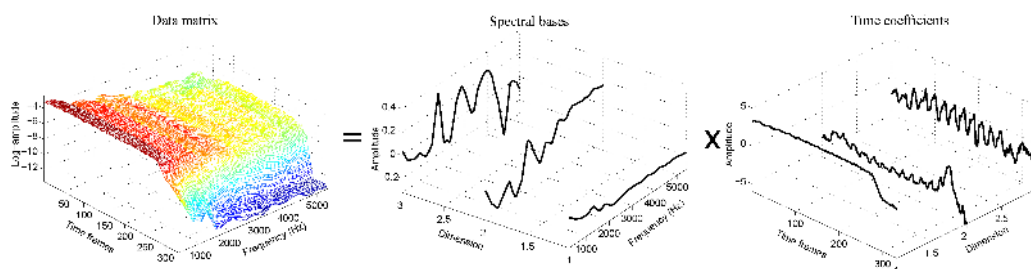


Figure 4.4: Example of basis decomposition of a spectral envelope by PCA. The data matrix is the product of the coefficient and basis matrices. The figure shows the first 3 bases of a violin note played with vibrato.

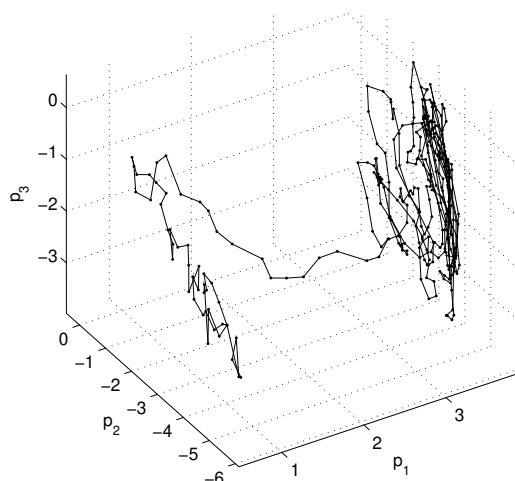


Figure 4.5: Interpretation of the basis decomposition of Fig. 4.4 as a projection onto the space spanned by the bases.

pronounced release phase. The second and third coefficient functions capture the effect of the vibrato. Thus, PCA has been able to decouple the influence of vibrato from the overall spectral shape. Similarly, the first spectral basis is the overall shape of the spectral envelope, and the higher order bases describe finer spectral variations that modulate the envelope at a rate dictated by the corresponding time functions.

Figure 4.5 shows the same decomposition interpreted as a projection onto the reduced-dimensional space defined by the first three PCA bases. The projected coefficients constitute a multivariate data cloud whose data vectors are ordered in time, as can be denoted by explicitly tracing a trajectory between them. Each point in PCA space corresponds to one time frame of the spectral envelope, whose shape will be determined by the position of the point relative to the axes. The cluster to the right corresponds to the sustained phase, in which the first coefficient (overall amplitude) is nearly constant, whereas the second and third coefficients oscillate due to the vibrato. The outcoming tail to the left corresponds to the release phase.

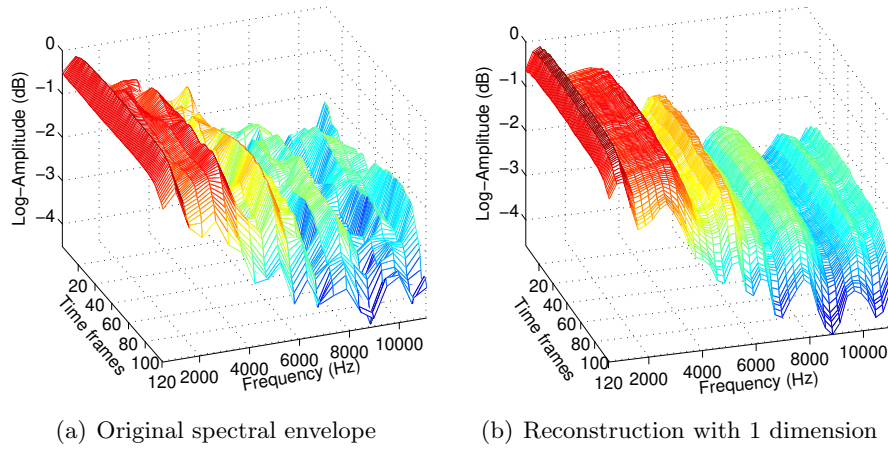


Figure 4.6: Example of envelope reconstruction with only the first PCA basis.

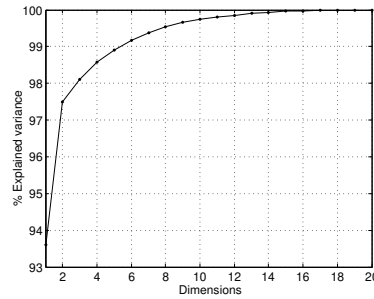


Figure 4.7: Explained variance as a function of dimensionality for a single French horn note.

Previous works [76, 134], as well as experiments performed in the scope of the present work, have shown the efficiency of PCA when applied to additive analysis data. This is due to the fact that partial amplitude trajectories are usually highly correlated, in consonance with the CASA grouping principles of similarity and common fate. As an example, the first dimension of PCA extracted from the additive parameters of single notes already accounts for more than 90% of the total variance, often reaching 99% with the first 5 or less dimensions. Figure 4.6(b) shows the reconstruction of a linearly interpolated spectral envelope using only the first PCA basis and the first coefficient function. When compared to the original envelope (Fig. 4.6(a)), it can be seen that it smooths the detailed spectral variations; nevertheless, the overall shape has been reasonably retained. The reconstruction with one dimension is just a constant envelope multiplied by a time-varying gain⁶.

Figure 4.7 gives a quantitative example. The plotted curve is the normalized ex-

⁶Apart from the amplitude adjustment produced by the term $E\{\mathbf{X}\}$ when de-centering.

plained variance as a function of the number of retained dimensions. As was shown in Sect. 2.3.4, the variance of each principal component equals the corresponding eigenvalue of the covariance matrix. The global explained variance is thus the sum of all previously extracted eigenvalues. The first dimension already attains almost 94% of the variance, and 99% is reached with only 5 of the original 20 dimensions, corresponding to a data reduction of 75%. This kind of measurement will be performed in a more systematic way in the evaluation of the training framework described later in this section.

When resynthesized using the original frequency information, similar results are obtained in terms of perceived quality. The difference is clearly audible when using only the first dimension. The sound is however indistinguishable from the original sinusoidal model when using 5 or more dimensions.

4.5.2 Dealing with variable frequency supports

Training a model based on adaptive spectral basis decomposition requires the extraction of a common set of bases for the training set. This is achieved by concatenating the spectra belonging to the classes to be trained (in this case, musical instruments) into a single input data matrix. As mentioned above, the spectral envelope may change with the pitch, and therefore training one single model with the whole pitch range of a given instrument may result in a poor timbral characterization. However, it can be expected that the changes in envelope shape will be minor for neighboring notes. Training with a moderate range of consecutive semitones will thus contribute to generality, and at the same time will reduce the size of the model.

In the case of additive data, the straightforward way to arrange the amplitudes into a spectral data matrix is to fix the number of partials to be extracted ($P_r = P$) and use the partial index p as frequency index, obtaining $\mathbf{X}(p, r)$ with elements $x_{pr} = \hat{A}_{pr}$. This approach will be referred to as *Partial Indexing* (PI). This is the method used in previous works like [76, 134].

However, when concatenating notes of different pitches for the training, their frequency support will change logarithmically. This has the effect of misaligning the f_0 -invariant features of the spectral envelope in the data matrix. This is illustrated in Fig. 4.8, which shows the concatenated notes of one octave of an alto saxophone. In the partial-indexed data matrix depicted in Fig. 4.8(c) (where coloring denotes partial amplitudes), diagonal lines descending in frequency for subsequent notes can be observed, which correspond to a misalignment of f_0 -invariant features. On the contrary, possible features that follow the logarithmic evolution of f_0 will become aligned.

An alternative approach will be evaluated, consisting on setting a fixed maximum frequency limit f_{max} before the additive analysis and extracting for each note the required number of partials to reach that frequency. This is the opposite situation as before: now the frequency range represented in each model is always the same, but the number of sinusoids is variable. To obtain a rectangular data matrix, an additional step is introduced in which the extracted spectral envelope is sampled in frequency at points defined by a grid of G points uniformly spaced within the

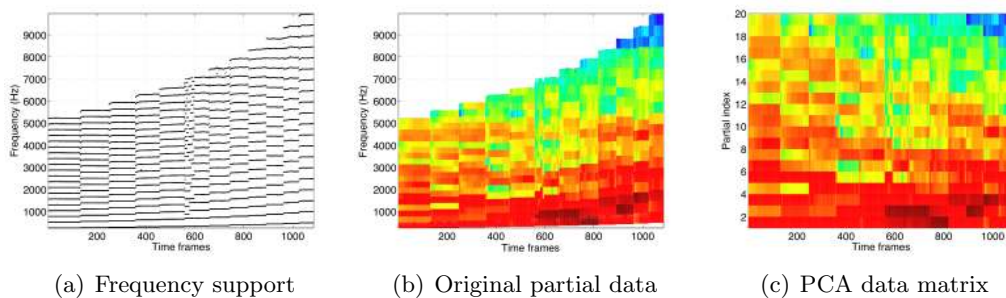


Figure 4.8: PCA data matrix with Partial Indexing (1 octave of an alto saxophone).

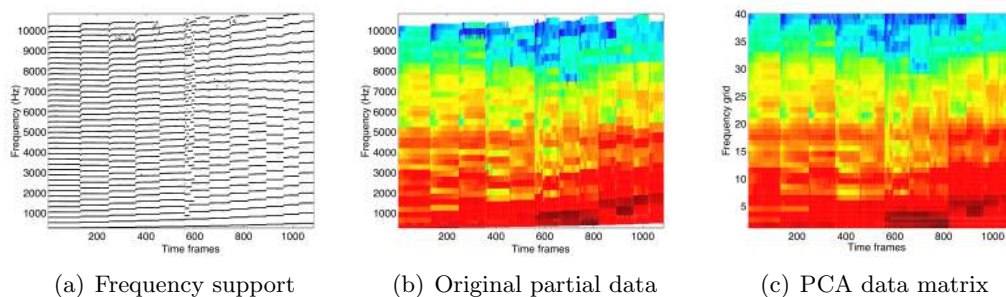


Figure 4.9: PCA data matrix with Envelope Interpolation (1 octave of an alto saxophone).

frequency range:

$$f_g = \frac{f_{max}}{G}g. \quad (4.12)$$

An example of this interpolation plus sampling approach is shown on Fig. 4.10. The spectral matrix is now defined by $\mathbf{X}(g, r)$, where $g = 1, \dots, G$ is the frequency grid index and r the frame index. Its elements will be denoted⁷ by $x_{gr} = \tilde{A}_{gr}$. This approach shall be referred to as *Envelope Interpolation* (EI). This strategy does not change frequency alignments (or misalignments), but additionally introduces an interpolation error. In the experiments, the two different interpolation methods already introduced will be evaluated: linear and cubic interpolation.

Figure 4.9 illustrates the effect of preprocessing the PCA data matrix with EI. The frequency support varies in density, but covers a nearly constant frequency range. The preprocessed data matrix preserves the frequency alignment of the formant-like features.

It is worth emphasizing at this point that the representations shown in Figs. 4.8(b) and 4.9(b) are not spectrograms (in which case the f_0 -invariant features would always be aligned), but amplitude diagrams of the extracted partials, which correspond exactly to the peaks defining the spectral envelope.

⁷The tilde notation ($\tilde{}$) will be used to denote interpolation.

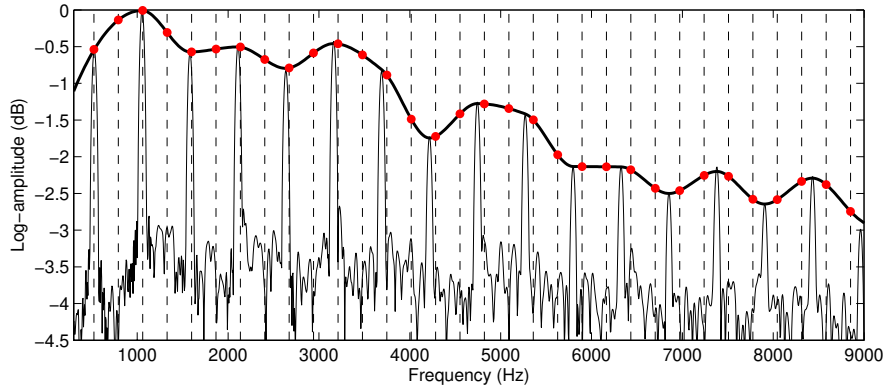


Figure 4.10: Cubic envelope interpolation at a regular frequency grid.

Frequency alignment is desirable for the present modeling approach because, if subsequent training samples share more common characteristics, prototype spectral shapes will be learnt more effectively. In other words, the data matrix will be more correlated and thus PCA will be able to obtain a better compression. In this context, the question arises of which one of the alternative preprocessing methods—PI (aligning f_0 -correlated features) or EI (aligning f_0 -invariant features)—is more appropriate. In order to answer to that, the experiments outlined in the next section were performed.

The issue of taking into account the f_0 -dependency of timbre within a computational model has only been addressed recently, although from a different perspective. For instance, in the work by Kitahara, Goto and Okuno [89], the aim is to explicitly model the pitch correlation of the features, instead of trying to accommodate the feature extraction process to reduce the error produced by not considering it, which is the approach proposed here. To that end, the cited work employs a multivariate Gaussian distribution in which the mean is f_0 -dependent and the covariance is constant. Results showed an improvement of around 4% classification accuracy for a database of isolated notes when using the explicit f_0 -dependency modeling.

4.5.3 Evaluation of the representation stage

The cross-validation⁸ experimental setup shown in Fig. 4.11 was implemented to test the validity of the representation stage and to evaluate the influence of the different preprocessing methods introduced: PI, linear EI and cubic EI. The audio samples belonging to the training database are subjected to sinusoidal modeling, concatenated and arranged into a spectral data matrix using one of the three methods. PCA is then applied to the data matrix. Dimension reduction is performed by keeping $D < K$ spectral dimensions ($K = P$ for PI and $K = G$ for EI), thus yielding

⁸ k -fold cross-validation refers to the experimental setup in which a database is successively partitioned k times into training and test sets. The results of the k experiment runs are averaged.

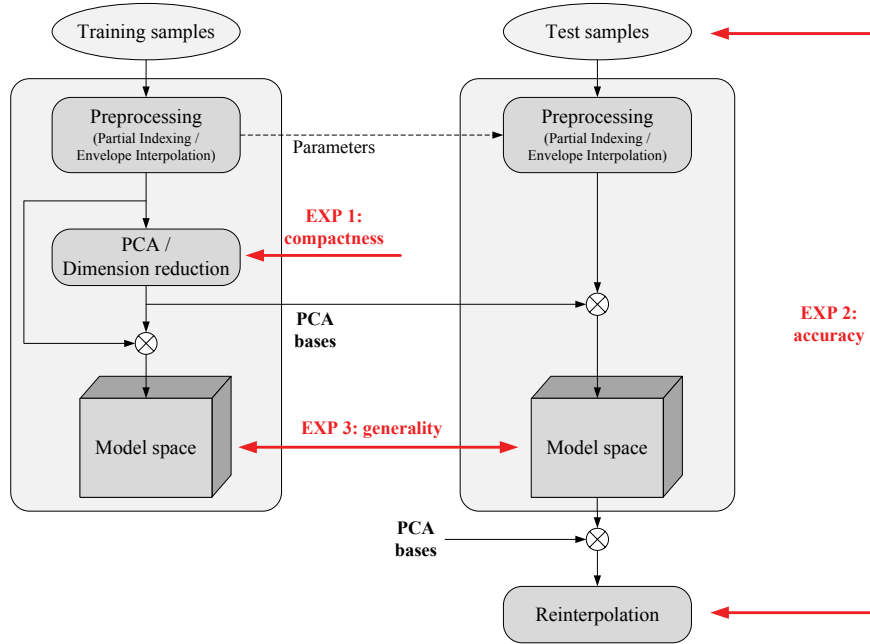


Figure 4.11: Cross-validation framework for the evaluation of the representation stage.

a common reduced basis matrix \mathbf{P}_ρ of size $K \times D$. The data matrix is then projected onto the obtained bases, and thus transformed into the reduced-dimension model space. The test samples are subjected to the same preprocessing, and afterwards projected onto the bases extracted from the training database. The test samples in model space can then be projected back into the time–frequency domain and, in the case of EI preprocessing, reinterpolated at the original frequency support. Each test sample is individually processed and evaluated, and afterwards the results are averaged over all experiment runs.

By measuring objective quantities at different points of the framework, it is possible to evaluate the requirements of compactness (experiment 1), reconstruction accuracy (experiment 2) and generality (experiment 3). Although each experiment was mainly motivated by its corresponding design criterion, it should be noted that they do not strictly measure them independently from each other.

In the following, the results obtained with three musical instruments belonging to three different families will be presented: violin (bowed strings), piano (struck strings or percussion) and bassoon (woodwinds). The used samples are part of the RWC Musical Instrument Sound Database [66]. One octave (C4 to B4) of two exemplars from each instrument type was trained. As test set, the same octave from a third exemplar from the database was used. For the PI method, $P = 20$ partials were extracted. For the EI method, f_{max} was set as the frequency of the 20th partial of the highest note present in the database, so that both methods span the same maximum frequency range, and a frequency grid of $G = 40$ points was defined.

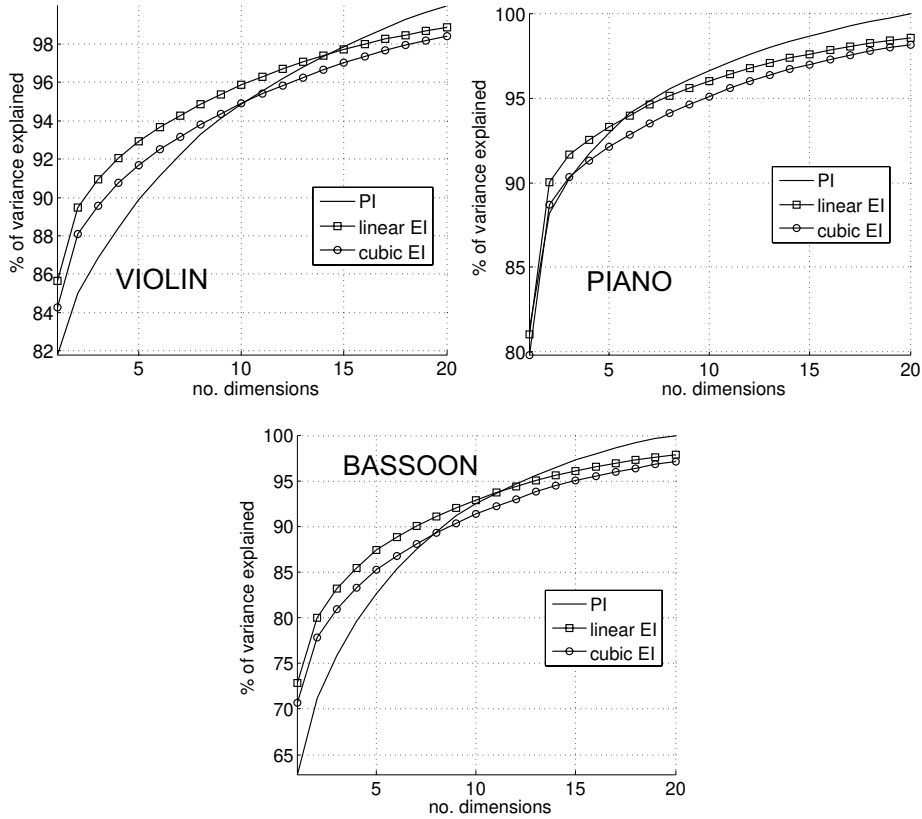


Figure 4.12: Results from experiment 1: explained variance.

Experiment 1: compactness.

The first experiment evaluates the ability of PCA to compress the training database by measuring the explained variance. As was shown in Sect. 2.3.4, the variance of each principal component equals the corresponding eigenvalue of the covariance matrix, and thus the total explained variance is the accumulated sum of the previously extracted eigenvalues. A normalized version has been used so that no dimension reduction corresponds to 100% of the variance:

$$EV(D) = 100 \frac{\sum_i^D \lambda_i}{\sum_i^K \lambda_i}, \quad (4.13)$$

where λ_i are the PCA eigenvalues, D is the reduced number of dimensions, and K is the total number of dimensions ($K = 20$ for PI and $K = 40$ for EI).

Figure 4.12 shows the results. The curves show that EI is capable of achieving a higher compression than PI for low dimensionalities ($D < 14$ for the violin, $D < 5$ for the piano and $D < 10$ for the bassoon). A 95% of variance is achieved with $R = 8$ for the violin, $R = 7$ for the piano and $R = 12$ for the bassoon.

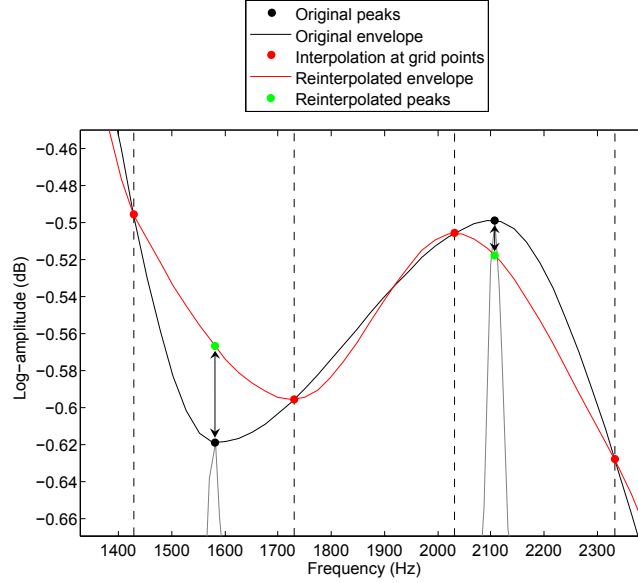


Figure 4.13: Reinterpolation error.

Experiment 2: reconstruction accuracy.

To test the amplitude accuracy of the envelopes provided by the representation stage, the dimension-reduced representations were projected back into the time–frequency domain, and compared with the original sinusoidal part of the signal. To that end, the *Relative Spectral Error* (RSE)[75] was measured:

$$\text{RSE} = \frac{1}{R} \sum_{r=1}^R \sqrt{\frac{\sum_{p=1}^{P_r} (\hat{A}_{pr} - \tilde{A}_{pr})^2}{\sum_{p=1}^{P_r} \hat{A}_{pr}^2}}, \quad (4.14)$$

where \hat{A}_{pr} is the original amplitude, \tilde{A}_{pr} is the reconstructed and reinterpolated amplitude, both at support point (p, r) , P_r is the number of partials at frame r and R is the total number of frames. In order to measure the RSE, the envelopes must be compared at the points of the original frequency support. This means that, in the case of the EI method, the back-projected envelopes must be reinterpolated using the original frequency information. As a consequence, the RSE accounts not only for the errors introduced by the dimension reduction, but also for the interpolation error itself, introduced by EI.

Note that in the EI approach there is always an error produced by reinterpolation, even if no dimension reduction is performed. This is illustrated in Fig. 4.13, which shows a close-up of the estimated envelope between two consecutive partial peaks in a given time frame. The black curve shows the original spectral envelope estimated by cubic interpolation between the original peaks \hat{A}_{pr} . This envelope is then sampled at the frequency grid, denoted by the vertical dashed lines, and yielding the sampled amplitudes \hat{A}_{gr} indicated by red points, which will be subjected to

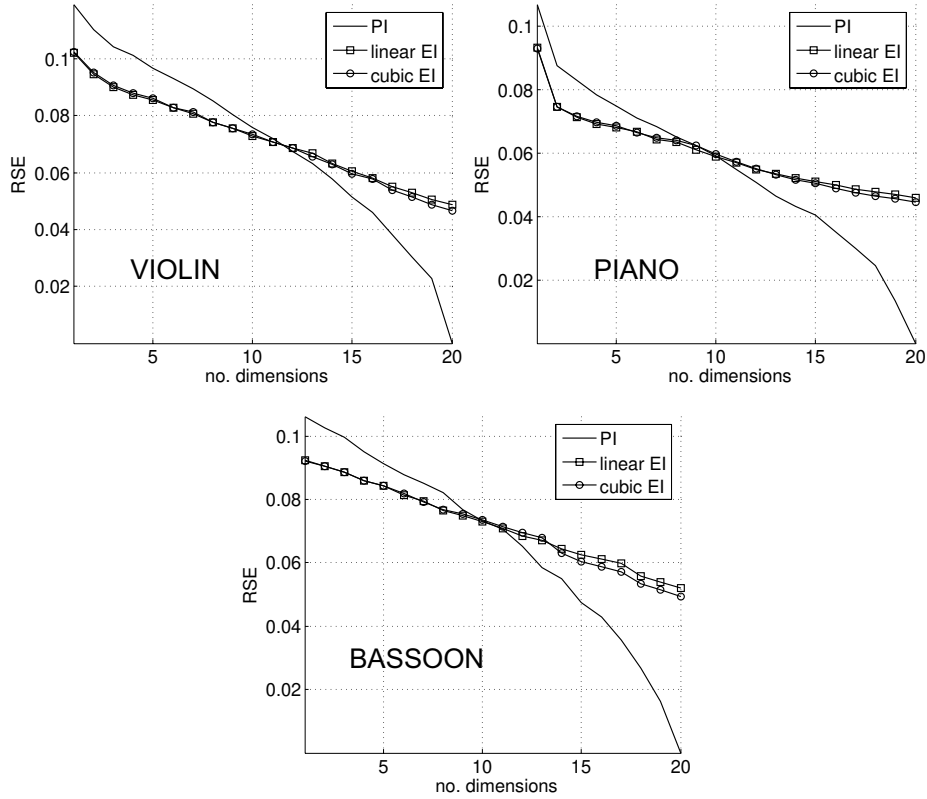


Figure 4.14: Results from experiment 2: Relative Spectral Error.

PCA analysis. If the full dimensionality is employed, like here, PCA is a perfectly invertible transformation, and thus the reconstructed amplitudes have the same values \tilde{A}_{gr} . Keeping a reduced set of bases would result in the reconstructed amplitudes lying on a different position along each frequency grid line. Finally, the reconstructed amplitudes are reinterpolated, producing the envelope shown in red which, as shown by the green points (that correspond to the reinterpolated amplitudes \tilde{A}_{pr}), differs from the original one at the original frequency support. In contrast, with PI the frequency support remains unchanged throughout all processing steps. Therefore, no reinterpolation is needed, and no reconstruction error is present for $D = K$.

The results of this experiment are shown in Fig. 4.14. EI reduces the reconstruction error in the low-dimensionality range. The curves for PI and EI must always cross because of the zero reconstruction error of PI with $D = K$ and of the reinterpolation error of EI. Interestingly, the cross points between both methods occur at around $D = 10$ for all three instruments.

Experiment 3: generality.

If the sets are large enough and representative, a high similarity between the training and testing data clouds in model space implies that the model has managed to cap-

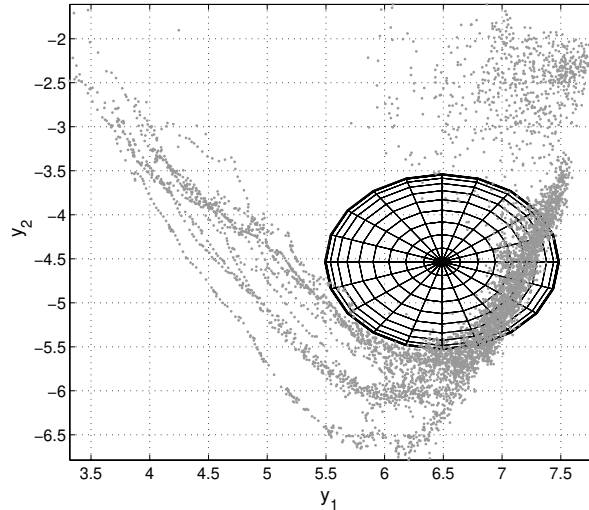


Figure 4.15: First two dimensions in model space of the training data for one octave of an acoustic guitar, and corresponding Gaussian model.

ture general features of the modeled instrument for different pitches and instrument exemplars. Thus, generality can be measured by defining a global distance measure between both data distributions.

Note that this way of measuring training/testing data similarity is solely based on the topology of the data points in feature space, and independent of any training or prototyping approach eventually used in later stages of the method. It is thus appropriate for the present purpose of evaluating the representation stage separately. The training/testing similarity will be implicitly assessed by the results of classification-related applications using the prototype models, whose success will depend on their discriminative power. Such classification tasks will be the subject of Sects. 4.7, 4.8, 5.2.4 and 6.4.

It was observed that most often the projected data clouds do not adopt simple cluster forms. For instance, when observing the scatter plot for one octave of an acoustic guitar (Fig. 4.15) it becomes clear that a single Gaussian density (in this case shown with diagonal covariance matrix) would not be able to yield a reasonable approximation. The same can be said of other models like GMM, which will not appropriately match the underlying data neither, at least not with a moderate number of model parameters. This is due to the non-sustained nature of the guitar notes, in which spectral shape is constantly changing along its decay phase, resulting in a monotonous trajectory in model space.

As a consequence, probabilistic distances that rely on the assumption of a certain probability distribution (like the divergence, the Bhattacharyya distance or the Cross Likelihood Ratio), which will yield inaccurate results for data not matching that distribution, were avoided. Instead, average point-to-point distances were used

because, since they are solely based on point geometry, they will be more reliable in the general case. In particular, the averaged minimum distance between point clouds, normalized by the number of dimensions, was computed:

$$\Delta_D(\omega_1, \omega_2) = \frac{1}{D} \left\{ \frac{1}{n_1} \sum_{i=1}^{n_1} \min_{\mathbf{y}_j \in \omega_2} \{d(\mathbf{y}_i, \mathbf{y}_j)\} + \frac{1}{n_2} \sum_{j=1}^{n_2} \min_{\mathbf{y}_i \in \omega_1} \{d(\mathbf{y}_i, \mathbf{y}_j)\} \right\}, \quad (4.15)$$

where ω_1 and ω_2 denote the two clusters, n_1 and n_2 are the number of points in each cluster, \mathbf{y}_i are the transformed coefficients, and $d(\cdot)$ denotes a given point-to-point distance.

An important point to note is that the distances are being measured in different spaces, each one defined by a different set of bases, one for each preprocessing method. A point-to-point distance susceptible to scale changes (such as the Euclidean distance) will yield erroneous comparisons. It is necessary to use a distance that takes into account the variance of the data in each dimension in order to appropriately weight their contributions. These requirements are met by the Mahalanobis distance:

$$d_M(\mathbf{y}_0, \mathbf{y}_1) = \sqrt{(\mathbf{y}_0 - \mathbf{y}_1)^T \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} (\mathbf{y}_0 - \mathbf{y}_1)}, \quad (4.16)$$

where $\boldsymbol{\Sigma}_{\mathbf{Y}}$ is the global covariance matrix of the training coefficients⁹. The results of this measurement are shown in Fig. 4.16. In all cases, and for all dimensionalities, EI has managed to reduce the distance between training and test sets in comparison to PI. The instrument with the greatest improvement was the piano.

For the sake of comparison, the same experiment was repeated for the piano, but this time using the *Kullback-Leibler* (KL) divergence, which is a density-dependent similarity measure. When applied to Gaussian distributions, the KL divergence is given by

$$\begin{aligned} \text{KL}(\omega_1, \omega_2) &= \frac{1}{2} \left(\log \left(\frac{\det \boldsymbol{\Sigma}_2}{\det \boldsymbol{\Sigma}_1} \right) + \text{tr}(\boldsymbol{\Sigma}_2^{-1} \boldsymbol{\Sigma}_1) \right. \\ &\quad \left. + (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}_2^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) - N \right). \end{aligned} \quad (4.17)$$

Note that the KL divergence measure relies only on the estimated parameters of the normal distributions $N_1(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $N_2(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ (the covariance matrix used in the Mahalanobis distance of Eq. 4.16 was a global one for whitening purposes, not a cluster-dependent one). The results of this experiment (Fig. 4.17), show that the measurements are misleadingly optimistic in comparison with the distribution-independent results of Fig. 4.16.

Evaluation of the representation stage: summary of conclusions

From the previous experiments it follows that using the Envelope Interpolation method for spectral representation improves compression efficiency, reduces the reconstruction error, and increases the similarity between test and training sets in

⁹The same result can also be obtained by performing an additional whitening step after PCA in order to normalize variances before using the Euclidean distance.

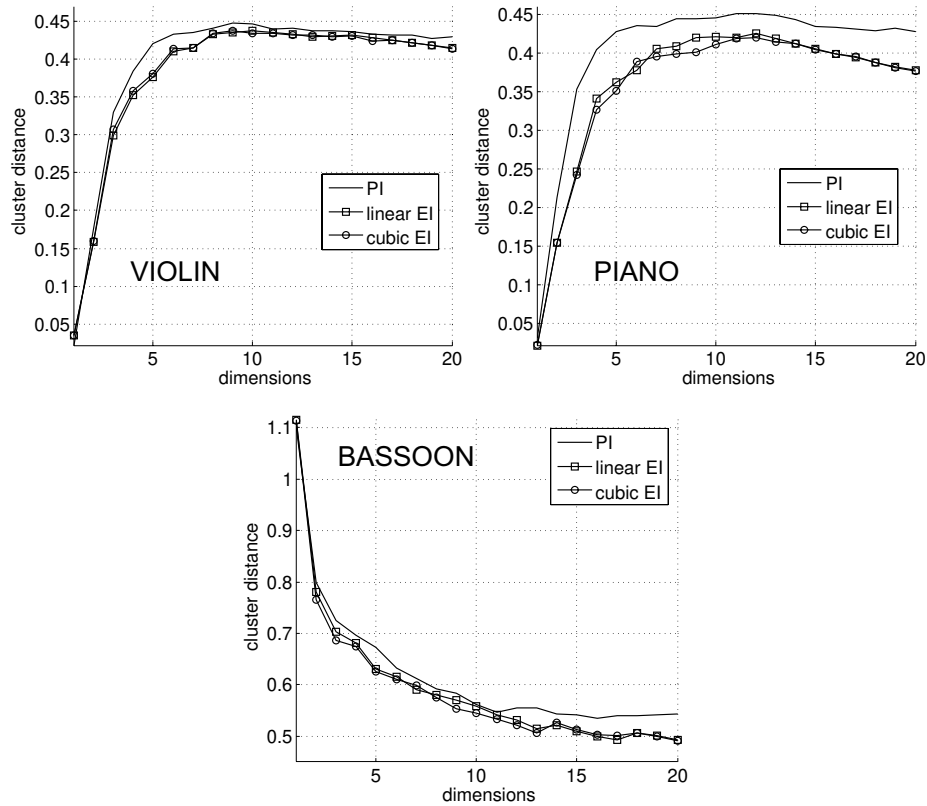


Figure 4.16: Results from experiment 3: training/test cluster distance.

principal component space, for a low to moderate dimensionality. In average, all three measures are improved for 10 or less dimensions, which already correspond to 95% of the variance contained in the original envelope data. In general, cubic and linear interpolation performed very similarly.

4.6 Prototyping stage

In model space, the projected coefficients must be grouped into a set of generic models representing the classes. Common methods from the field of MIR include GMMs and HMMs. Both are based on clustering the transformed coefficients into a set of densities, either static (GMM) or linked by transition probabilities (HMM). The exact variation of the envelope in time is either completely ignored in the former case, or approximated as a sequence of states in the latter.

For the sake of accuracy, however, the time variation of the envelope should be modeled in a more accurate manner, since as mentioned it plays an equally important role as the envelope shape when characterizing timbre. Therefore, the choice here was to always keep the sequential ordering of the coefficients, and to represent each class as a trajectory rather than as a cluster. For each class, all training trajectories

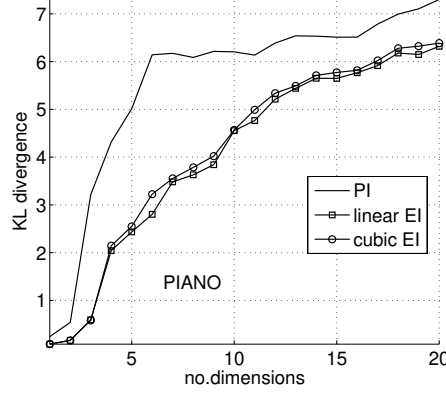


Figure 4.17: Training/test cluster distance measured by KL divergence.

are to be collapsed into a single *prototype curve* representing that instrument.

To that end, the following steps are taken. Let \mathcal{Y}_{si} denote the coefficient trajectory in model space corresponding to training sample s (with $s = 1, \dots, S_i$) belonging to instrument i (with $i = 1, \dots, I$), of length R_{si} frames:

$$\mathcal{Y}_{si} = (\mathbf{y}_{si1}, \mathbf{y}_{si2}, \dots, \mathbf{y}_{siR_{si}}). \quad (4.18)$$

First, all trajectories are interpolated in time using the underlying time scales in order to obtain the same number of points (in this case, cubic interpolation was used). In particular, the longest trajectory, of length R_{max} is selected and all other ones are interpolated so that they have that length.

$$\tilde{\mathcal{Y}}_{si} = \text{interp}_{R_{max}}\{\mathcal{Y}_{si}\} = (\tilde{\mathbf{y}}_{si1}, \tilde{\mathbf{y}}_{si2}, \dots, \tilde{\mathbf{y}}_{siR_{max}}), \quad \forall s, i. \quad (4.19)$$

Then, each point in the resulting prototype curve for instrument i , of length R_{max} , denoted by

$$\mathcal{C}_i = (\mathbf{p}_{i1}, \mathbf{p}_{i2}, \dots, \mathbf{p}_{iR_{max}}), \quad (4.20)$$

is considered to be a D -dimensional Gaussian random variable $\mathbf{p}_{ir} \sim N(\boldsymbol{\mu}_{ir}, \boldsymbol{\Sigma}_{ir})$ with empirical mean

$$\boldsymbol{\mu}_{ir} = \frac{1}{S_i} \sum_{s=1}^{S_i} \tilde{\mathbf{y}}_{sir} \quad (4.21)$$

and empirical covariance matrix $\boldsymbol{\Sigma}_{ir}$, which for simplicity will be assumed diagonal, where $\boldsymbol{\sigma}_{ir}^2 = \text{diag}(\boldsymbol{\Sigma}_{ir})$ is given by

$$\boldsymbol{\sigma}_{ir}^2 = \frac{1}{S_i - 1} \sum_{s=1}^{S_i} (\tilde{\mathbf{y}}_{sir} - \boldsymbol{\mu}_{ir})^2. \quad (4.22)$$

A prototype curve can be thus interpreted as a D -dimensional, non-stationary *Gaussian Process* (GP) with time-varying means and covariances parametrized by the frame index r :

$$\mathcal{C}_i \sim GP(\boldsymbol{\mu}_i(r), \boldsymbol{\Sigma}_i(r)). \quad (4.23)$$

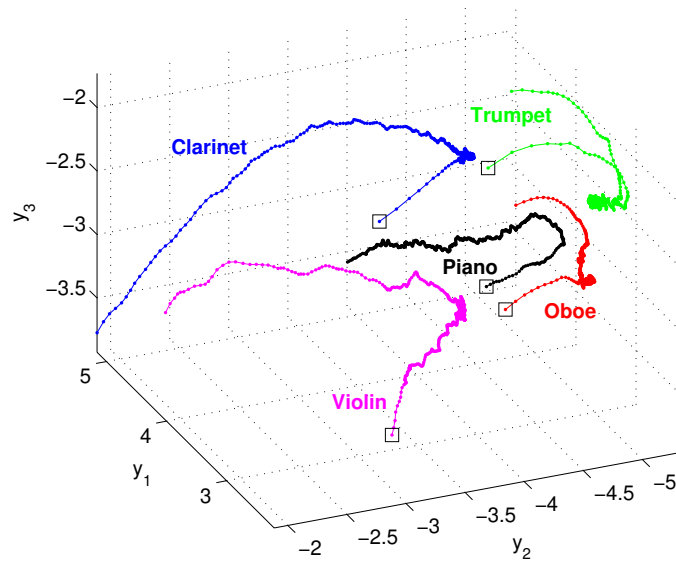


Figure 4.18: Prototype curves in the first 3 dimensions of model space corresponding to a 5-class training database of 423 sound samples, preprocessed using linear envelope interpolation. The starting points are denoted by squares.

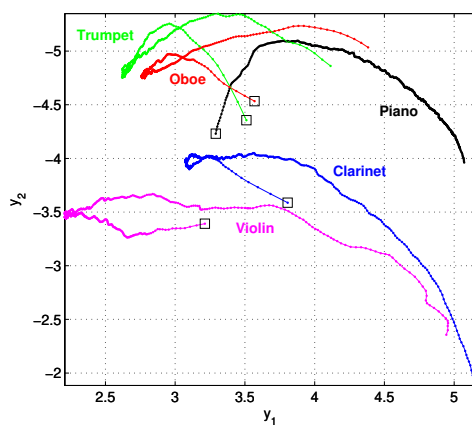
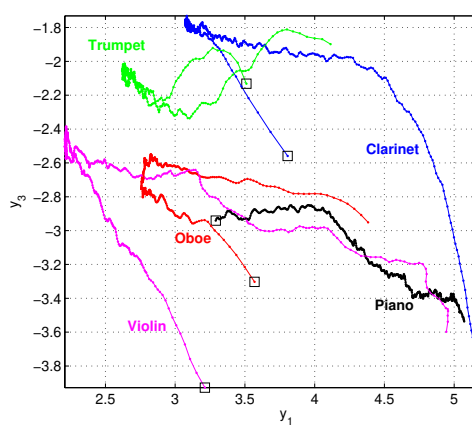
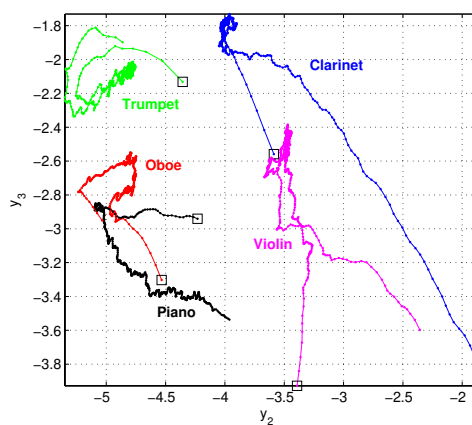
Figure 4.18 shows an example set of mean prototype curves corresponding to a training set of 5 classes: piano, clarinet, oboe, violin and trumpet, in the first three dimensions of a common PCA model space. The database consists of all dynamic levels (piano, mezzoforte and forte) of two or three exemplars of each instrument type, with normal playing style, covering a range of one octave between C4 and B4. This makes a total of 423 sound files. Note that under the projection in which the space is represented on the figure, the mean curves are perfectly separable. This is a remarkable fact, since PCA is not optimized for separability, such as other linear transformations like *Linear Discriminant Analysis* (LDA), but for compactness. In this case, however, it manages to attain a high degree of separation.

Note that, for the graphical representation of the probabilistic prototype curves, only the mean curves formed by the values μ_{ir} were plotted. It is worth emphasizing, however, that each curve has an “influence area” around it as determined by their time-varying covariances.

Figure 4.19 depicts the same trained prototype curves under the three orthogonal projections parallel to the axes, so that the influence and significance of each principal component can be more clearly assessed. The y_2y_3 projection (Fig. 4.19(c)) is clearly the one that attains the highest curve separability.

Note lengths do not affect the length or the shape of the training trajectories. Short notes and long notes share the same curve in space as long as they have the same timbral evolution, the former having a smaller density of points on the curve than the latter.

When projected back to the time–frequency domain, each prototype trajectory

(a) Projection to the y_1y_2 plane(b) Projection to the y_1y_3 plane(c) Projection to the y_2y_3 plane**Figure 4.19:** Orthogonal projections of the timbre space of Fig. 4.18.

will correspond to a *prototype envelope* consisting of a mean surface and a variance surface, which will be denoted by $\mathbf{M}_i(g, r)$ and $\mathbf{V}_i(g, r)$, respectively, where $g = 1, \dots, G$ denotes the regular frequency grid of Eq. 4.12 and $r = 1, \dots, R_{max}$ for all the models. Each D -dimensional mean point $\boldsymbol{\mu}_{ir}$ in model space will correspond to a G -dimensional vector of mean amplitudes constituting a time frame of the reconstructed spectral envelope. From the properties of the Gaussian distribution, it is known that a linear transformation of the form $\mathbf{A}\mathbf{x} + \mathbf{c}$ applied to the variable \mathbf{x} with distribution $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ results in the distribution $N(\mathbf{A}\boldsymbol{\mu} + \mathbf{c}, \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^T)$. Thus, undoing the effects of whitening and centering, the reconstructed vector of means is given by

$$\hat{\boldsymbol{\mu}}_{ir} = \mathbf{P}_\rho \boldsymbol{\Lambda}_\rho^{1/2} \boldsymbol{\mu}_{ir} + E\{\mathbf{X}\} \quad (4.24)$$

and the corresponding variance vector

$$\hat{\boldsymbol{\sigma}}_{ir}^2 = \text{diag} \left(\mathbf{P}_\rho \boldsymbol{\Lambda}_\rho^{1/2} \boldsymbol{\Sigma}_{ir} (\mathbf{P}_\rho \boldsymbol{\Lambda}_\rho^{1/2})^T \right), \quad (4.25)$$

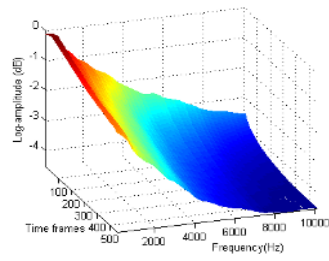
both of G dimensions, which form the columns of $\mathbf{M}_i(g, r)$ and $\mathbf{V}_i(g, r)$, respectively. Again, for representation purposes only the mean surfaces $\mathbf{M}_i(g, r)$ will be used, but variance surfaces are always implicit in the model as well.

Figure 4.20 shows the prototype envelopes corresponding to the prototype curves of Figs. 4.18 and 4.19. For each envelope, a time–frequency, three-dimensional view is shown, together with the projections from the frequency axes, which show more clearly the overall shape characteristics of the spectral envelope. Note the different formant-related features in the mid-low frequency areas. The coloring reflects the logarithmic amplitudes.

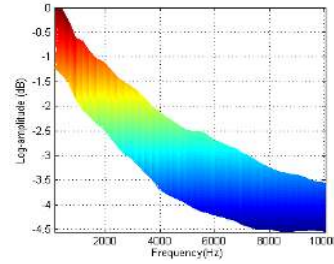
Analogously as in model space, a prototype envelope can be interpreted as a Gaussian Process, but in a slightly different sense. Instead of being multidimensional, the GP is unidimensional (in amplitude), but parametrized with means and variances varying in the 2-dimensional time–frequency plane. Such prototype envelopes are intended to be used as time–frequency templates that can be interpolated at any desired time–frequency point. Thus, the probabilistic parametrization can be considered continuous, and therefore the indices t and f will be used, instead of their discrete counterparts r and k . The prototype envelopes can then be denoted by

$$\mathcal{E}_i \sim GP(\mu_i(t, f), \sigma_i^2(t, f)). \quad (4.26)$$

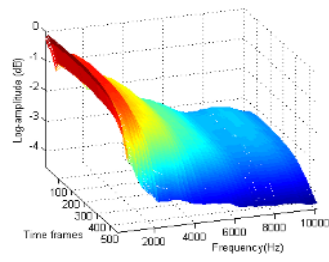
Depending on the application, it can be more convenient to perform further processing on the reduced-dimensional PCA space or back in the time–frequency domain. When classifying individual notes, such as introduced in the next section, a distance measure between unknown trajectories and the prototype curves in PCA space has proven a successful approach. However, in applications where the signals to be analyzed are mixtures of notes, such as polyphonic instrument recognition (Sect. 4.8) or source separation (Chapter 5), the envelopes to be compared to the models can contain regions of unresolved overlapping partials or outliers, which can introduce important interpolation errors when adapted to the frequency grid needed for projection onto the bases. In those cases, working in the time–frequency domain will be more convenient.



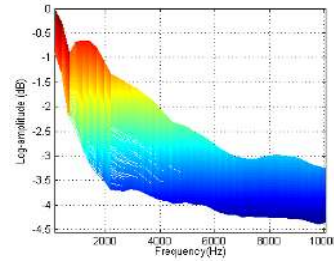
(a) Piano



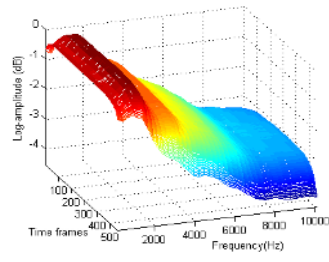
(b) Piano (frequency profile)



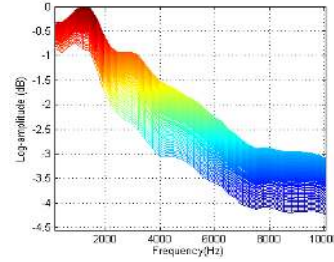
(c) Clarinet



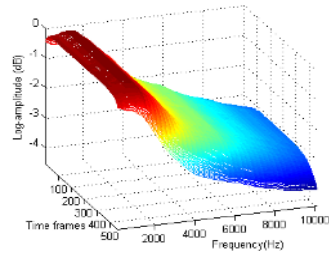
(d) Clarinet (freq. profile)



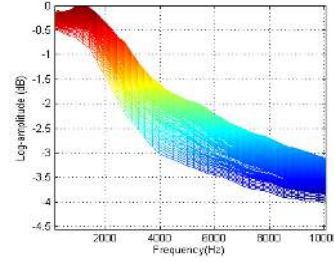
(e) Oboe



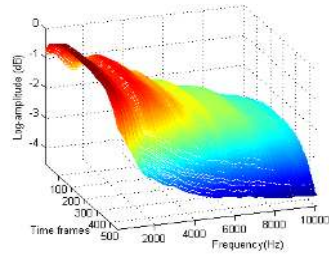
(f) Oboe (frequency profile)



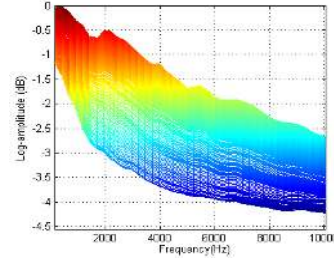
(g) Trumpet



(h) Trumpet (freq. profile)



(i) Violin



(j) Violin (frequency profile)

Figure 4.20: Prototype envelopes corresponding to the curves on Fig. 4.18.

Remarks about the observed formants

On the frequency profile representations of Fig. 4.20, several prominent formants are clearly visible, constituting the characteristic averaged spectral shapes of the respective instruments. A number of acoustic studies with the purpose of analyzing the formants of musical instruments have been reported in the literature, and their observations are consistent with the average resonances found with the modeling procedure that has been presented here.

As an example, the frequency profile of the clarinet (Fig. 4.20(d)) shows a spectral hill that corresponds to the first measured formant, which, as reported by Backus [10], has its maximum between 1500 Hz and 1700 Hz.

In the work by Meyer [110], the first formant of the oboe is reported to start around 1100 Hz (roughly corresponding to the voice formant of the vocal “a”), and the second to be centered around 2700 Hz (between vocals “e” and “i”). The corresponding numbers given by Backus [10] are, respectively, 1400 Hz and 3000 Hz. The corresponding profile of Fig. 4.20(f) shows a very prominent formant in the first region, and a more attenuated hump-like resonance in the second.

In the case of the trumpet, a single formant is visible on Fig. 4.20(h), corresponding to its first acoustically measured formant, observed by Meyer between 1200 Hz and 1500 Hz and by Backus between 1200 Hz and 1400 Hz. Finally, the clear bump around 2000 Hz on the violin profile (Fig. 4.20(j)) can be identified as the “bridge hill” observed by several authors (see, e.g., [63]) in that frequency area, produced by resonances of the bridge.

On the interpretation of the timbre axes

To gain further insight into the meaning of the timbre axes, the spectral envelope will be evaluated and plotted at different points of the timbre space. In benefit of clarity, a two-dimensional projection of the space onto the first two dimensions is performed, and several evaluation locations were chosen as indicated on Fig. 4.21. The used database is the same as in the previous figures. Two kinds of graphical examples are provided. The first was intended to explicitly illustrate the variance of the envelopes in the frequency domain as defined in Eq. 4.25. To that end, three points on three different prototype curves were selected, as indicated by the red dots on the figure. Point A corresponds to the first prototype frame of the clarinet, point B to a mid-length prototype frame of the trumpet, and point C to one of the last frames of the piano. Figure 4.22 represents the corresponding mean envelopes as the thick line, enclosed by two thinner lines representing the variance.

Figure 4.23 represents the evolution of the spectral envelope alongside the straight traces defined on Fig. 4.21, sampled uniformly at 10 different points. The thicker envelopes correspond to the starting points on the traces, which are then followed in the direction marked by the arrows. Traces 1 to 4 are parallel to the axes, thus showing the latter’s individual influence on the envelope.

From traces 1 and 3 it can be asserted that the first dimension (axis y_1) mostly (but not only) affects the overall amount of decreasing slope of the spectral envelope. Such overall slope can be approximated as the slope of the straight line one

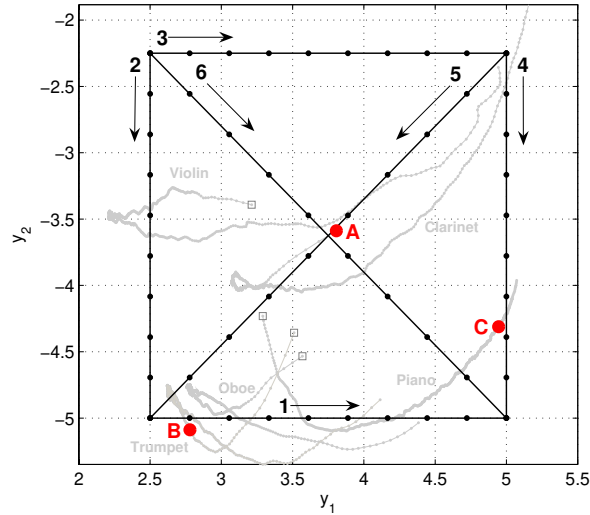


Figure 4.21: Envelope evaluation points and traces for Figs. 4.22 and 4.23.

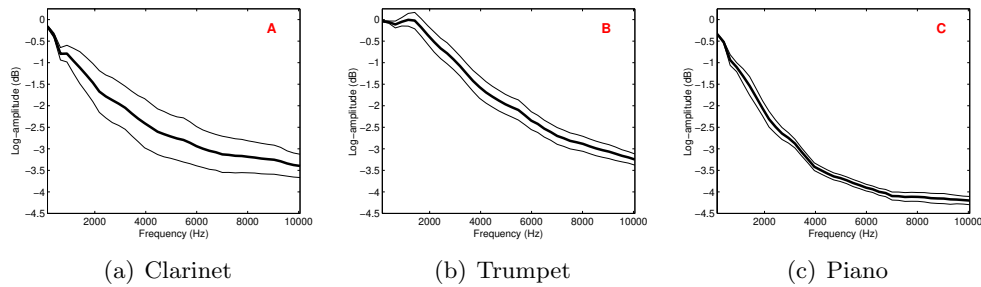


Figure 4.22: Envelope mean and variances at points A,B and C on Fig. 4.21.

would obtain performing linear regression on the spectral envelope. Along traces 2 and 4 (axis y_2), the envelope has the clear behavior of changing the ratio between low-frequency and high-frequency spectral content. For decreasing values of y_2 , high-frequency contents decreases and low-frequency contents increases, producing a rotation of the spectral shape around a pivoting point at approximately 4000 Hz. It follows that one of the most affected features is in this case the spectral centroid (center of gravity), or its perceptual counterpart, the “brightness”. In contrast to the first dimension, the slope of the linear regression would not significantly change. The centroid or brightness has often been identified as one of the 2 or 3 most important perceptual dimensions in classic and recent psychoacoustical studies based on listening tests, such as in Wessel [179], McAdams *et al.* [106] and Lakatos [96]. Traces 5 and 6 travel alongside the diagonals and represent thus a combination of both behaviors.

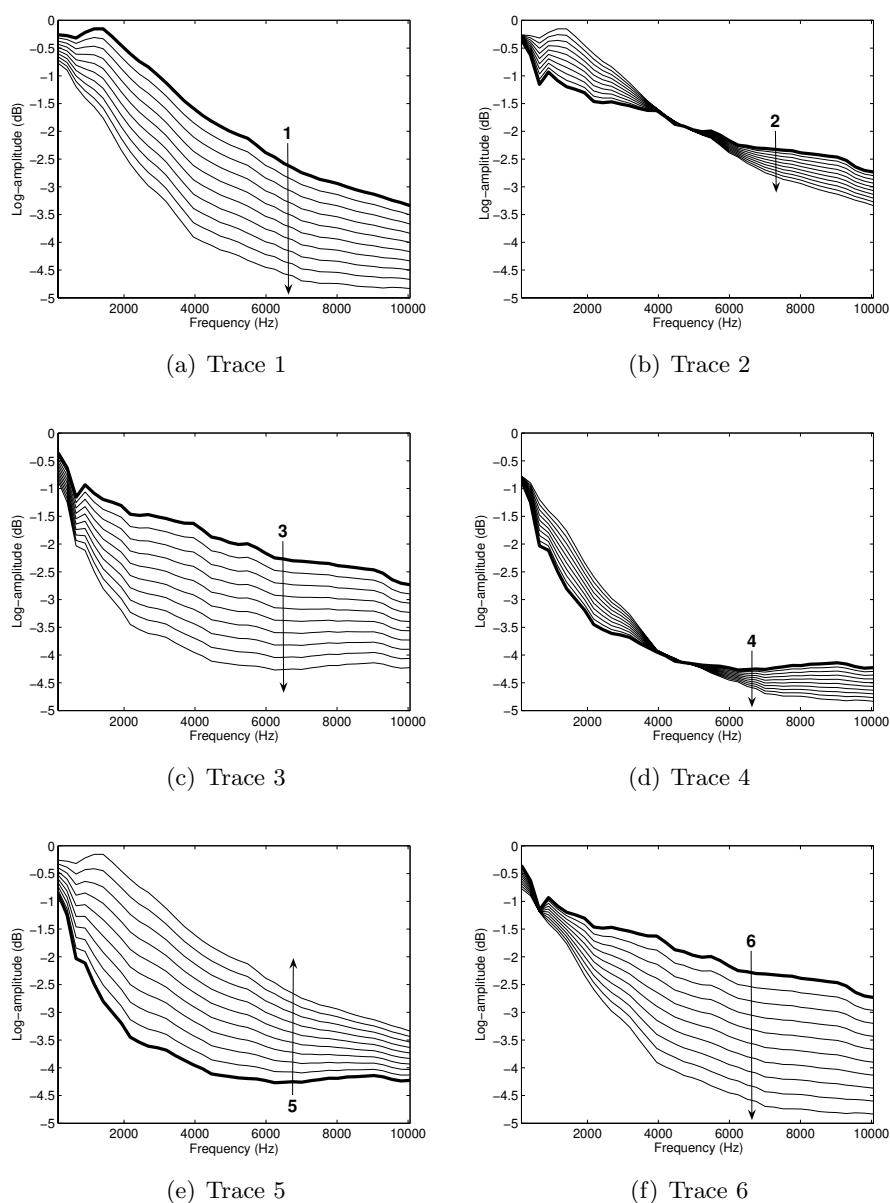


Figure 4.23: Evolution of the spectral envelope alongside the traces on Fig. 4.21.

4.7 Application to musical instrument classification

In the previous sections it has been shown that the proposed modeling approach is successful in capturing the timbral content of individual instruments. For most applications, however, dissimilarity between different models is desired. Therefore, it is desirable to evaluate the performance of the model within a classification context involving solo instrumental samples. Such a classification task is a popular MIR

application, aimed at the efficient managing and searching of sample databases. A comprehensive overview of isolated instrumental sample classification can be found in the work by Herrera, Peeters and Dubnov [73]. In that work, the reported accuracies for classification problems containing few instrumental classes (less than 10) reach percentages higher than 90% with a variety of techniques from the literature.

One possibility to perform such a classification task using the present model is to extract a common basis for the whole training set, compute one prototype curve for each class and measure the distance between an input curve and each prototype curve. Like for prototyping, the curves must have the same number of points, and thus the input curve must be interpolated with the number of points of the densest prototype curve, of length R_{max} . The distance between an interpolated unknown curve \tilde{U} and the i -th prototype curve \mathcal{C}_i is defined here as the average Euclidean distance between their mean points:

$$d(\tilde{U}, \mathcal{C}_i) = \frac{1}{R_{max}} \sum_{r=1}^{R_{max}} \|\tilde{\mathbf{u}}_r - \boldsymbol{\mu}_{ir}\| = \frac{1}{R_{max}} \sum_{r=1}^{R_{max}} \sqrt{\sum_{k=1}^D (\tilde{u}_{rk} - \mu_{irk})^2}. \quad (4.27)$$

The class corresponding to the lowest distance is chosen.

For the experiments, a set of 5 classes was defined (piano, clarinet, oboe, violin and trumpet), again from the RWC database [66], each containing all notes present in the database for a range of two octaves (C4 to B5), in all different dynamics (forte, mezzoforte and piano) and normal playing style. This makes a total of 1098 individual note files, all sampled at 44,1 kHz. For each method and each number of dimensions, the experiments were iterated using 10-fold cross-validation. The same parameters as in the representation stage evaluations were used: $P = 20$ partials for PI, and a frequency grid of $G = 40$ points.

The obtained classification accuracy curves are shown on Fig. 4.24. Note that each data point is the result of averaging the 10 folds of cross-validation. The experiments were iterated up to a dimensionality of $D = 20$, which is the full dimensionality in the PI case. The best classification results are given in Table 4.1. With PI, a maximal accuracy of 74.86% was obtained. This was outperformed by around 20% when using the EI approach, obtaining 94.86% for linear interpolation and 94.59% for cubic interpolation. As in the representation stage experiments, performance does not significantly differ between linear and cubic interpolation. As has been seen, the obtained accuracies are comparable to those of state-of-the-art systems.

Note that the curves rise quickly in the low-dimensionality area, and very strongly tend to stabilize starting from around $D = 8$ or $D = 10$. This is another confirmation of the fact that the first few dimensions of model space capture the most informative timbral features.

4.7.1 Comparison with MFCC

The widely used *Mel Frequency Cepstral Coefficients* (MFCC), originally proposed by Davis and Mermelstein [50], are comparable to the proposed model inasmuch

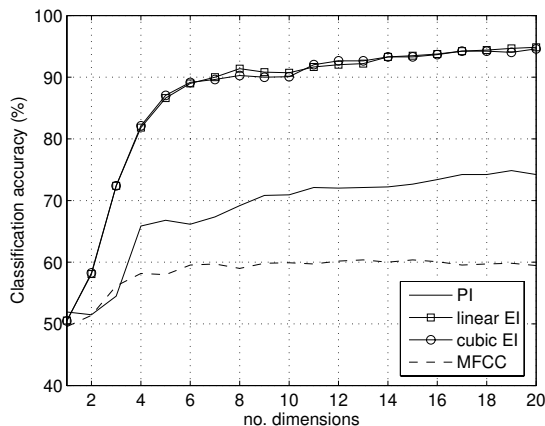


Figure 4.24: Classification results: averaged classification accuracy.

Representation	Accuracy	STD
PI	74.86 %	$\pm 2.84\%$
Linear EI	94.86 %	$\pm 2.13\%$
Cubic EI	94.59 %	$\pm 2.72\%$
MFCC	60.37 %	$\pm 4.10\%$

Table 4.1: Classification results: maximum averaged classification accuracy and standard deviation (STD) using 10-fold cross-validation.

as they aim at a compact description of spectral shape. However, as anticipated before, they lack accuracy in the description of the spectral envelope. To compare the performances of both approaches, the experiments were repeated with exactly the same set of database partitions, substituting the representation stage with a standard computation of MFCCs¹⁰. Before presenting the results, the extraction process [51] will be briefly addressed.

The extraction of MFCCs relies on a modification of the original cepstrum definition of Eq. 4.2 in order to simulate the non-linear human hearing mechanism, which has been shown to further improve the effectiveness of cepstral analysis. In particular, the frequency axis is warped to approximate the nonlinear perception of pitch which, as was introduced in Sect. 3.1, is approximated by the mel scale (Eq. 3.23).

Such a warping is implemented by filtering the original DFT spectral content with a bank of L bandpass filters with center frequencies $f_i = k_i \frac{f_s}{N}$, where N is the DFT size, spaced by the mel scale and with a bandwidth given by some approximation to critical bandwidth. In the context of cepstral analysis, this original spectral content is $\log(|X(k)|)$ and the filtering results in a set of *log total energies* in the

¹⁰The implementation used here was the one contained in Slaney's Auditory Toolbox [141].

critical bands, given by

$$Y(i) = \sum_{k=0}^{N/2} \log(|X(k)|) H_i \left(k \frac{2\pi}{N} \right), \quad (4.28)$$

where H_i is the frequency response of the filter (usually assumed triangular). Note that the values $Y(i)$ must lie on the center frequencies f_i , and the rest of the bins must be set to zero:

$$Y(k) = \begin{cases} Y(i) & , \quad k = k_i \\ 0 & , \quad k \neq k_i \end{cases}. \quad (4.29)$$

After the mel-warping, the final MFCCs, (the *mel-cepstrum*) are obtained by taking the DCT:

$$c_{mel}(q) = \sum_{k=0}^{N-1} Y(k) \cos \left[\frac{\pi}{N} \left(k + \frac{1}{2} \right) q \right]. \quad (4.30)$$

Taking the DCT has the effect of concentrating the energy in the first few coefficients (this concentration is however less efficient than the one obtained with PCA, which as was shown in Sect. 2.3.4, is optimal both in the variance and in the MSE sense).

The steps to extract the MFCCs can then be summarized as follows (in the case of short-time processing, all of them must be performed for each windowed frame):

1. Compute the DFT $X(k)$ from the input signal $x(t)$.
2. Take the logarithm from the amplitude, $\log(|X(k)|)$.
3. Apply mel-scale warping using critical band averaging (Eq. 4.29).
4. Take the DCT thereof.

The coefficients obtained in this way were subjected to the same prototyping approach outlined in Sect. 4.6, and a set of MFCC prototype curves was thus created. Again, classification based on average point-to-point Euclidean distance was performed with the same database, under exactly the same cross-validation partitions. The results are shown on Fig. 4.24 and Table 4.1. The highest achieved classification rate was of 60,37 % (with 13 coefficients), i.e., around 34 % lower than obtained with EI.

4.8 Application to polyphonic instrument recognition

A more demanding classification-related task in which the presented modeling approach can be used is that of polyphonic, multi-timbral instrument recognition, i.e., the detection of the instruments that are present in a monaural mixture. The problem is more difficult than the previous isolated-note classification, for the same reasons that complicate source separation: the spectral components of the sources overlap in the time–frequency domain. Approaches for polyphonic instrument recognition follow either one of two possible paths, corresponding to either

the *understanding-without-separation* or the *separation-for-understanding* paradigm, both introduced in Sect. 1.1. Examples of the first class, which avoid separation and try to detect the instruments from the whole mixture, include the works by Essid, Richard and David [60] and Livshin and Rodet [100]. The second kind of approaches perform a partial source separation, not necessarily oriented to quality, but enough to allow further, quasi-isolated processing (see, e.g., the work by Kashino and Murase [85]).

The timbre modeling approach presented in the current chapter was tested within a recognition framework of the second type, consisting on a first separation-oriented block and a second timbre-matching block. The first block yields a set of time–frequency clusters, each one ideally corresponding to a single note. These are then passed on to the matching block, in which the clusters are compared to each one of the timbre models, and the highest match is selected as the instrument for that note.

The separation block uses sinusoidal modeling as front-end and is based on the *Normalized Cut* (Ncut) criterion, which originated from the domain of video and image segmentation [139]. The goal of the Ncut algorithm is to partition a graph following solely topological criteria. It was first proposed for audio applications by Lagrange and Tzanetakis [95], where the nodes of the graph correspond to the amplitudes of the partial peaks obtained by additive analysis. A detailed description of the algorithm can be found in the previous references.

For the present discussion, it suffices to know that the Ncut stage provides a set of clusters of partial frequencies and amplitudes that approximately correspond to each individual note present in the mixture. A particular cluster of R_j frames will be represented here as an ordered set of amplitude and frequency time frames $\mathbf{A}_j = (\mathbf{a}_1, \dots, \mathbf{a}_{R_j})$ and $\mathbf{F}_j = (\mathbf{f}_1, \dots, \mathbf{f}_{R_j})$, each one with possibly a different number of partials P_1, \dots, P_{R_j} .

In this particular application scenario, working in the reconstructed time–frequency domain instead of in model space (i.e., using prototype envelopes rather than prototype curves) is more convenient, to avoid the already mentioned large interpolation errors produced by unresolved partials, undetected time–frequency areas or outliers that can be produced by the separation stage.

Timbre matching

Each one of the clusters obtained by the Ncut stage is matched against each one of the mean prototype envelopes $\mathbf{M}_i(g, r)$. The proposed approach consists of evaluating the prototype envelope of model i at the frequency support of the input cluster j . This operation will be denoted by

$$\tilde{\mathbf{M}}_{ij} = \mathbf{M}_i(\mathbf{F}_j). \quad (4.31)$$

To that end, the time scales of both input and model are first normalized. Then, the model frames closest to each one of the input frames in the normalized time scale are selected. Finally, each new amplitude value \tilde{m}_{pr}^{ij} is linearly interpolated from the neighboring amplitude values of the selected model frame.

Classified as	True instruments					
	p	o	c	t	v	s
p	100	0	0	0	0	0
o	0	100	8	8	0	0
c	0	0	67	0	33	0
t	0	0	0	92	0	8
v	0	0	0	0	58	8
s	0	0	25	0	8	83

Table 4.2: Confusion matrix (detection accuracies in %) for single-note instrument classification. The labels denote: piano (**p**), oboe (**o**), clarinet (**c**), trumpet (**t**), violin (**v**) and alto sax (**s**).

The distance between the j -th cluster and the i -th interpolated prototype envelope is then defined as

$$d(\mathbf{A}_j, \tilde{\mathbf{M}}_{ij}) = \frac{1}{R_j} \sum_{r=1}^{R_j} \sqrt{\sum_{p=1}^{P_{R_j}} (A_{pr}^j - \tilde{m}_{pr}^{ij})^2}, \quad (4.32)$$

i.e., the average of the Euclidean distances between frames of the input clusters and interpolated prototype envelopes at the normalized time scale. This is the time–frequency counterpart of the timbre-space curve distance of Eq. 4.27. The model $\tilde{\mathbf{M}}^{ij}$ minimizing this distance is chosen as the predicted instrument for classification.

Results for isolated notes

As a baseline for comparison with the multi-note case, the timbre matching stage was tested for the task of classification of isolated notes, such as in the previous section. The matching and decision process, however, takes now place in the time–frequency domain using envelopes, rather than in model space as before.

A dataset of 72 notes within the range C4 to B4 belonging to 6 instruments (piano, oboe, clarinet, trumpet, violin and alto saxophone) was again extracted from the RWC Musical Instrument Sound Database [66]. The results are shown on the confusion matrix of Table 4.2. The overall classification rate was 83.3%. Violin and clarinet turned out to be the most difficult instruments to classify.

Results for mixed notes

More interesting are the results for multiple notes. A total of 54 synthetic mixtures were created, each one containing 2, 3 or 4 simultaneous notes belonging to different instruments, all with synchronous onsets. The first evaluation is based on the *true positive* (TP) and *false positive* (FP) values. In this context, true positives are the number of separated clusters correctly classified as an instrument present in the mixture. False positives are the number of instrument detections not present in the mixture. Based on these quantities, the standard Information Retrieval measures of *Recall* (RCL):

$$\text{RCL} = \frac{\text{TP}}{\text{COUNT}}, \quad (4.33)$$

	2-note			3-note			4-note			total		
	RCL	PRC	F1	RCL	PRC	F1	RCL	PRC	F1	RCL	PRC	F1
p	83	100	91	22	100	36	0	0	0	23	100	38
o	133	75	96	100	46	63	67	40	50	86	50	63
c	33	100	50	33	100	50	40	86	55	36	93	52
t	89	100	94	58	100	74	58	64	61	67	85	75
v	67	67	67	83	45	59	83	36	50	80	43	56
s	100	43	60	78	60	63	60	75	67	67	62	64
total	75	79	77	56	64	59	46	56	50	56	64	60

Table 4.3: Recall (RCL), precision (PRC) and F-Measure (F1) values for instrument identification in multiple-note mixtures.

where COUNT is the total number of notes of a given instrument present in the mixture database, *Precision* (PRC):

$$\text{PRC} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (4.34)$$

and *F-Measure* (F1):

$$\text{F1} = \frac{2 \cdot \text{RCL} \cdot \text{PRC}}{\text{RCL} + \text{PRC}} \quad (4.35)$$

were computed. The results are shown in Table 4.3. It can be seen that the system detected correctly 75% of the instrument occurrences in the 2-note case, 56% in the 3-note case and 46% in the 4-note case. The corresponding average precisions were 79%, 64% and 56%, respectively. The most demanding identification was posed by the piano in the 4-note case, where no note was correctly detected as being produced by it.

The second evaluation criterion used is a more demanding one: it not only requires, as before, that the present instruments are correctly detected, but that each individual note detection actually corresponds to the detected instruments. For example, a violin+piano mixture would have been correctly detected with the previous measures even if the first note was wrongly detected as a piano and the second as a violin. To avoid this, the separated clusters were subjected to simple fundamental frequency estimation, so that they could be compared with the correct training notes that were labeled according to pitch in the database. Pitch detection consisted on computing a frequency histogram of each cluster, yielding the strongest partials, followed by a second histogram of differences between those strongest partial frequencies, yielding the fundamental frequency estimate. This simple approach was sufficient in such an isolated-note scenario.

With the pitch information obtained in this way, the results in Table 4.4 were obtained. The table shows the note-by-note detection accuracy, i.e., the percentage of individually, correctly detected notes. The average performances were 65%, 50% and 33% for the 2-, 3- and 4-note cases, respectively.

4.9 Conclusions

The task of developing a computational model representing the timbral characteristics of musical instruments has been addressed in the present chapter. The develop-

	Instrument detection accuracy			
	2-note	3-note	4-note	overall
p	67	67	0	55
o	100	86	60	81
c	33	29	19	26
t	75	33	22	43
v	67	100	50	75
s	75	36	42	44
total	65	50	33	47

Table 4.4: Instrument classification performance (detection accuracy in %) for 2-, 3- and 4-note mixtures.

ment criteria were chosen and combined so that such models can be used not only as a source model for source separation, but also in a wide range of MIR applications. To that end, techniques aiming at compactness (PCA), accuracy of the envelope description (sinusoidal modeling) and model generalization (training and prototyping) were combined into a single framework. The obtained features were modeled as prototype curves in the reduced-dimension space, which can be projected back into the time–frequency domain to yield a set of prototype envelopes.

A particular point of interest was the evaluation of the frequency misalignment effects that occur when notes of different pitches are used in the same training database. In order to handle that, a representation strategy based on frequency interpolation was proposed as an alternative to applying data reduction directly to the partial parameters. This *Envelope Interpolation* (EI) technique improved objective measures of explained variance, reconstruction error and training/test cluster similarity for low and moderate dimensionalities of up to around 1/4 of the full dimensionality, which already corresponds to around 95% of the total variance. It also improves prototype-curve-based classification of isolated instrumental samples by 20% in comparison to using plain partial indexing and by 34% in comparison to using MFCCs as the features. It follows that the interpolation error introduced by EI is compensated by the gain in correlation in the training data. It can also be concluded that f_0 -invariant features play a more important role in such a PCA-based model, and thus their frequency alignment must be favored.

In a first content analysis context, the models were employed in a classification task involving isolated sound samples. Class decisions were based on average distances between the prototype curves and the unknown trajectories in PCA space. The method attained 94.86% classification accuracy with 5 classes. In comparison, the accuracy using MFCCs as the representation stage was of 60.37%.

The models were also successfully employed in the time–frequency timbre matching stage of a single-channel polyphonic instrument recognition system based on a previous cluster extraction using the Ncut criterion. Obtained note-by-note accuracies range from 65% for 2-voice mixtures to 33% for 4-voice mixtures for a database of 6 instruments.

The modeling approach proposed here can be extended and refined in many different ways, and it can be considered for other applications such as transcription or realistic sound transformations. A detailed discussion of possibilities for future

developments will be given in Sect. 7.2.

The next chapter will be devoted to the usage of the developed timbre models as a source of a priori information within a monaural source separation context. Chapter 6 will extend the approach to the stereo case.

5

Monaural separation based on timbre models

The timbre models discussed in the previous chapter were developed considering their primary application as time–frequency templates guiding the separation of the partials present in a mixture according to the source they belong to. This was the main reason why an accurate representation of the spectral envelope based on sinusoidal modeling and interpolation was chosen, in contrast with other, general-purpose timbre models using only rough descriptions of the spectral shape, such as MFCCs or MPEG-7’s Audio Spectrum Envelope feature.

The present chapter describes the development of a novel monaural source separation approach based on those models. Monaural (single-channel) separation is the most underdetermined situation, in which only one mixture or sensor is observed:

$$x(t) = \sum_{n=1}^N a_n s_n(t), \quad (5.1)$$

and thus no spatial cues can be used to search for the mixing parameters, as was the case in the methods presented in Sects. 2.6 and 3.4. Therefore, the separation success relies solely on the capacity of the models to group the transformed coefficients into sources, and their representation accuracy will directly be reflected on the separation quality. A method that combines both the spectral information provided by the models and spatial separation cues will be the subject of the next chapter.

In general, the single-channel separation problem calls for the use of sophisticated signal models, either pre-trained or assuming a certain level of structural configuration of the signals. The next section will present a brief review of previous works that rely on such kind of modeling. The proposed system will be introduced in Sect. 5.2, following a sequential description of the different processing steps involved. Its experimental evaluation will be addressed in Sect. 5.3, and a summary of conclusions in Sect. 5.4. Parts of the present chapter were previously published in [32] and [33].

5.1 Monaural music separation based on advanced source models

Source separation from a single channel is a demanding problem that requires either strong assumptions about the nature of the sources, a fair amount of a priori infor-

mation, or a combination of both. Since no spatial information can be exploited, basic signal models such as the STFT or the DWT are not sufficient, and more elaborate descriptions are needed.

Systems of this type can be classified into unsupervised and supervised ones. The first do not rely on a previous training and generate the models in a data-driven fashion. Examples include methods based on adaptive basis decomposition (Sect. 2.3) and on sinusoidal modeling (Sect. 4.2), which can be interpreted as adaptive models whose parameters are estimated from the signals. They will be reviewed in the next two subsections. Supervised methods, in contrast, employ a training database of source examples to estimate the model before the actual separation takes place, and will be introduced in Sect. 5.1.3. An overview of methods aiming at monaural music separation can be found in the work by Siamantas, Every and Szymanski [140]. An overview of unsupervised methods can be found in the work by Virtanen [171].

Note that all approaches reviewed here perform separation from a single channel. There exist hybrid systems that combine both the use of advanced source models with the exploitation of spatial cues for the stereo or multichannel case. They will be introduced in the next chapter.

5.1.1 Unsupervised methods based on adaptive basis decomposition

Data-driven basis decomposition applied to spectra has already appeared in the present work (Sect. 4.5.1) in the context of PCA-based dimensionality reduction to obtain appropriate representation spaces for the timbral descriptions. As again suggested by the representation/separation analogy (Sect. 2.4), the same principle can be applied to the mixture signals in order to obtain the separated sources as the result of the decomposition.

Usually, the obtained expansion functions do not directly correspond to the sources; each source is rather formed by the sum of a certain subset of bases. This requires an additional clustering step after the basis decomposition, so that each subset of bases is grouped into a source stream. The clustering step has turned out to be the most demanding one in many proposed systems, and must sometimes be performed manually [1, 176].

An example of this kind of methods is the application of ICA (Sect. 2.6.1) to a time–frequency representation of the input signal, as proposed by Casey [39]. The ICA requirement that there must be the same number of sources than sensors is overcome by performing analysis in the transformed domain, and decomposing the mixture spectrogram into a set of statistically independent subspaces. This approach is called *Independent Subspace Analysis* (ISA). Clustering of the bases into sources is performed by partitioning a matrix, which they called *ixegram*, whose components are the symmetric Kullback-Leibler divergences between each pair of independent components.

A related approach is based on the constraint that the obtained spectral bases and coefficients must be non-negative, which is meaningful when working with amplitude or power spectra. Such is the case of the application of *Nonnegative Matrix*

Factorization (NMF) [97] to time–frequency representations, such as in the work by Wang and Plumbley [176], which yields a set of spectral masks used for the unmixing. The clustering of the corresponding bases must be performed manually. Durrieu *et al.* [54] combine NMF with a source–filter model of the singing voice for its separation from the accompaniment.

A related set of approaches are based on *Nonnegative Sparse Coding* (NSC) [77], which combines the criterion of non-negativity with that of sparsity. An example is the transcription-related approach proposed by Abdallah and Plumbley [1], where the original NMF algorithm is extended assuming a sparse generative model with multiplicative noise. Virtanen [169] further extends NSC with an additional criterion of temporal continuity, formulated as a cost function measuring the absolute value of the overall amplitude difference between spectral frames. Here, each temporal expansion function is assumed to correspond to a source, and thus no clustering step is needed. This however implies that each source is supposed to be generated by a constant spectral envelope (given by the corresponding spectral basis) multiplied by a time-varying gain. This is a too inaccurate approximation for most real-world, non-stationary music signals.

A possibility to improve temporal accuracy was later proposed by Virtanen [170] as a further extension of NSC. The generative model for each source consists of the convolution of spectrograms (thus, full time–frequency representations conveying the dynamic variation of the spectral envelope) with a vector of onsets. This approach was thus named *Convolutional Sparse Coding*.

5.1.2 Unsupervised methods based on sinusoidal modeling

A different family of unsupervised approaches is based on sinusoidal modeling (Sect. 4.2), which is also a highly sparse model and allows a detailed handling of overlapping partials. They are based on grouping the extracted partials according to ASA cues (see Sect. 2.8), such as temporal and spectral smoothness, harmonic concordance, common onsets and offsets or common modulations, and can be thus interpreted as data-driven CASA implementations.

In an early 1990 approach by Maher [103], the goal is to separate two sources after a preliminary multipitch estimation step. Then, four different methods for resolving the overlapping partials are compared: the solution of a set of linear equations, the analysis of beating components (which takes advantage of the fact that the interference of two sinusoids with close frequencies results in an amplitude modulation at the rate of the frequency difference), linear interpolation from neighbouring partials, and the use of a set of fixed spectral envelopes as templates. The latter approach is thus supervised, but was discarded for the final evaluations because of its lack of robustness.

In [173], the first of a series of works by Virtanen dealing with separation based on sinusoidal modeling, CASA-like cues are implemented by a perceptual distance that measures synchronous changes and harmonic concordance. The individual trajectories are grouped into sources by selecting the combination that minimizes that distance. The system is limited in that it cannot handle notes with the same pitch

or the same onset. Also, according to the author, the results were not reliable for a large number of sources.

To overcome these problems, an iterative approach was later proposed [174]. After a first, rough estimation of the sinusoidal frequencies by means of a multipitch estimation stage, the parameters are refined in a least squares optimization based on a linear model for the amplitudes and imposing harmonicity constraints for the frequencies. An intermediate processing step of amplitude smoothing in a critical-band frequency scale is introduced to simulate the smoothness of the spectral envelope. This procedure was extended in [172] with a more refined model of spectral smoothness and in [168] with a similar model for temporal smoothness. In both cases, the smoothness is modeled by means of basis decomposition of the harmonic structures and their evolution in time. In the spectral case, the expansion functions are bandpass-filtered harmonic combs at frequency locations defined by several warpings, such as a critical-band scale and a mel warping (Sect. 3.1).

In the previous approaches, once the individual partials have been assigned to sources, their parameters are used to resynthesize them by means of additive analysis (Sect. 4.2). An alternative approach has been presented by Every and Szymanski [62], where spectral filtering techniques are used to resolve overlapping sinusoids, which are detected based on harmonicity relations. The amplitudes of the overlapping partials are linearly interpolated from the nearest neighbors. Then, a set of especially designed notch filters is applied to separate each pair of overlapping peaks. The method can be interpreted as an adaptive time–frequency masking process driven by the sinusoidal parameters. Its advantage over additive synthesis is that it allows a more accurate estimation of the noise residual for further processing. This system is non-blind, since it requires an explicit a priori knowledge about the pitches in form of a MIDI score, or alternatively, a robust multipitch pre-processing stage.

An explicit separation of the noise residual part is addressed by Every in [61]. One of the methods proposed is based on the assumed amplitude correlation between the spectral envelope of the sinusoids and the spectral envelope of the noise floor.

5.1.3 Supervised methods

The above methods are unsupervised, without a training stage, and are based on generic source models. To further improve separation, statistical models of the sources can be trained beforehand on a database of isolated source samples, at the cost of reducing the general applicability of the system to unknown sound or instrument types.

A possibility of using a priori trained models arises in the context of Wiener-filtering-based source separation, which was briefly introduced in Sect. 2.7.3. In particular, Benaroya and Bimbot [18] use a database of isolated samples in order to train alternatively GMMs or HMMs as the source priors needed for Wiener filtering in the STFT domain. The approach is tested upon a 2-source single-channel mixture. A similar approach was used by Ozerov *et al.* [122] to separate singing voice from accompaniment. However, instead of using a separated training database, the

priors are in this case extracted online from the processed mixture, which must have been previously segmented manually into the vocal and non-vocal parts. A further Wiener-based extension by Benaroya *et al.* [19] used a more sophisticated NSC model instead of GMMs or Gaussian-state HMMs.

Vincent and Plumbley [166] propose a Bayesian framework to estimate the frequency and amplitude parameters of the harmonic components (a set of harmonically related sinusoids spanning several frames), as well as the noise residual, constituting a monaural mixture, under an MAP criterion. The spectral envelope of each component is assumed to be fixed and multiplied by a time-varying gain. The priors of the model are learnt on a database of isolated sounds. A better performance than NMF is reported. Automatic clustering of the components into sources is however not discussed. Instead, clustering is based on maximizing the separation performance, given the sources are known beforehand.

Meron and Hirose [109] address the single-channel separation of singing and piano accompaniment, based on the prior knowledge of the musical score of the piano part and on sinusoidal modeling. A model of the piano is trained on isolated samples. It consists of a parametric description of each partial's ADSR envelope, whose frequency support is assumed to be constant.

Kashino *et al.* [86] combine a prediction-driven CASA architecture (see Sect. 2.8) with a range of a priori musical knowledge sources, including a chord transition dictionary, a chord-note relation database, a set of chord naming rules, ASA grouping rules (harmonicity and common onset), a set of sample spectral envelopes (which they call *tone memory*) and a set of timbre models which in this case are simple Gaussian models of a set of dimensionality-reduced (via PCA) features, such as onset gradients and frequency modulations. This architecture is called OPTIMA (Organized Processing Toward Intelligent Music Scene Analysis). This framework was extended to handle overlapping partials by Kinoshita *et al.* [87].

Bay and Beauchamp [14] base their separation system on Sinusoidal Modeling and on a preliminary multipitch estimation step that assumes harmonicity. A library of stored spectra is created by clustering a set of training, isolated-note spectra for each fundamental frequency via the k-Means algorithm and selecting their centroids. The library consists thus on a set of averaged static spectral shapes. The non-overlapping sinusoids, predicted from the f_0 information and the harmonicity constraint, are matched with the library after a nearest-neighbor criterion, and the overlapping sinusoids are retrieved from the best-matching template.

5.2 Proposed system

Many of the modeling steps discussed in the previous chapter (envelope estimation based on sinusoidal modeling, envelope interpolation in the frequency grid, evaluation of the reconstruction errors) were motivated by an increase of representation accuracy, having as ultimate goal their deployment within a separation context. The system that will be presented below was devised to evaluate and demonstrate the ability of the timbre models as templates guiding the peak selection and unmixing

for separation. For that reason, single-channel separation was addressed first, so that no spatial cues can be exploited and the success of separation will directly depend on the models.

Most approaches based on sinusoidal modeling rely either on a previous multipitch estimation stage [14, 172, 168] or on the knowledge of the MIDI or music score of the mixture [61, 62, 109]. In contrast, as will be seen, for the proposed approach no previous multipitch estimation or any kind of a priori pitch-related score is needed. Instead, separation is solely based on common-onset properties of the partials, and on the analysis of the evolution in time of the spectral envelope they define.

Another novelty is that no assumptions on harmonicity are made, unlike all previous sinusoidal-modeling-based approaches [61, 62, 103, 172, 169, 173, 174]. Instead, separation is based on the CASA-like cues of common fate and good continuation of the sinusoidal amplitudes, and on their comparison with the pre-stored time–frequency templates which, it should be kept in mind, are interpolated surfaces that cover a continuous range of frequencies. This allows separating highly inharmonic sounds and separating chords played by a single instrument. The knowledge of the number and names of the instruments is not mandatory, but will obviously increase the performance.

System overview

Figure 5.1 shows an overview of the proposed separation system. It will be briefly introduced here; a detailed presentation of the processing steps and its evaluation will constitute the remainder of the chapter.

First, the mixture signal is subjected to sinusoidal modeling, obtaining a set of sinusoidal tracks. Next, an onset detection stage follows (Sect. 5.2.2), based on identifying synchronously starting tracks. In the track grouping module (Sect. 5.2.3), tracks corresponding to the same onset are grouped together, and overlapping tracks are detected. The core of the separation system is formed by the timbre matching (Sect. 5.2.4) and track retrieval (Sect. 5.2.5) modules, both based on the trained timbre models described in the previous chapter. The timbre matching module assigns an instrument to each track group, eliminating the need for a post-separation clustering. Since this module also outputs onset/offset information and the instrument each note belongs to, it can also be used for segmentation or polyphonic instrument recognition. The track extension module retrieves the missing segments of the partial tracks (either due to overlapping or for not having been detected), as well as entirely overlapping tracks, from the timbre models. Finally, the separated tracks are resynthesized using additive synthesis.

Requirements on sinusoidal modeling

As will be seen in Sects. 5.2.2 and 5.2.5, the method requires analyzed partial tracks to be split if there is a quick change in amplitude of moderate proportions, and thus a sinusoidal extraction with a high sensitivity to amplitude changes must be performed. On the other hand, a too high sensitivity would result on tracks being cut

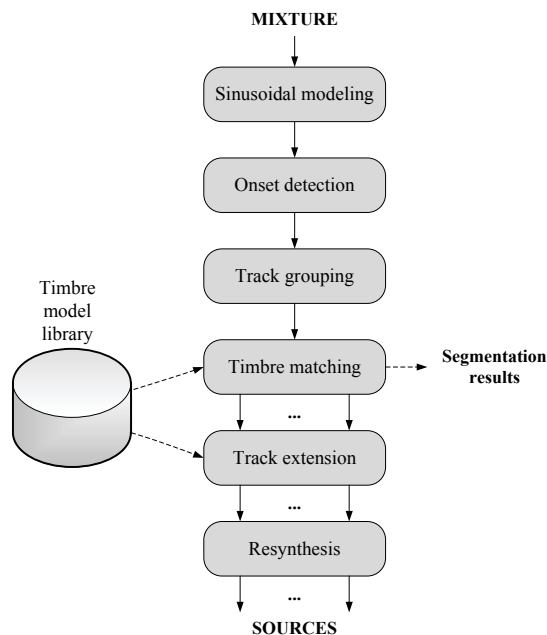


Figure 5.1: Monaural source separation system overview.

at intra-note amplitude modulations. An appropriate amplitude sensitivity balance must be thus set carefully.

A further point to note when setting the extraction parameters is that peak picking and partial tracking must be performed in inharmonic mode, since the signals to be analyzed are mixtures of unknown pitches and harmonicities. Thus, the use of a thresholding procedure to prevent the noise floor from being detected, like the one detailed in Sect. 4.2 and illustrated on Fig. 4.2, is crucial.

5.2.1 Experimental setup

Throughout this chapter, the separation system developed will be tested with a large set of experimental databases, each one consisting of a collection of mixtures with distinctive characteristics that determine the level of complexity in question. Not only the separation performance, but also the instrument classification accuracy will be evaluated. The degree of difficulty will be determined by the polyphony, the knowledge or absence of knowledge of the number and types of instruments, the length of the mixture, the melodic and harmonic relationships between notes, and the spectral characteristics of the involved instruments. The reader is again referred to Tables 2.1 and 2.2 for a contextual overview of such experimental demands.

The experimental criteria have been organized according to Table 5.1. The group of basic experiments (EXP 1 to EXP 3k¹) corresponds to the mixing conditions most

¹The “k” suffix denotes the fact that the instruments are assumed to be known a priori. It is only used in case there is another instance of the same experiment set with unknown instruments.

Type	Name	Source content	Harmony	Instruments	Polyphony
Basic	EXP 1	Individual notes	Consonant	Unknown	2,3,4
	EXP 2	Individual notes	Dissonant	Unknown	2,3,4
	EXP 3	Sequence of notes	Cons., Diss.	Unknown	2,3
	EXP 3k	Sequence of notes	Cons., Diss.	Known	2,3
Extended	EXP 4	One chord	Consonant	Unknown	2,3
	EXP 5	One cluster	Dissonant	Unknown	2,3
	EXP 6	Sequence with chords	Cons., Diss.	Known	2,3
	EXP 7	Inharmonic notes	-	Known	2

Table 5.1: Table of experimental setups for the monaural separation system.

commonly tested with monaural separation systems: each source is a sequence of one or more individual notes. The extended experiments (EXP 4 to EXP 7) demonstrate advanced capabilities of the proposed method: the separation of sources containing same-instrument chords and inharmonic sounds. Further implications and demands of each experiment will be detailed in the course of the chapter.

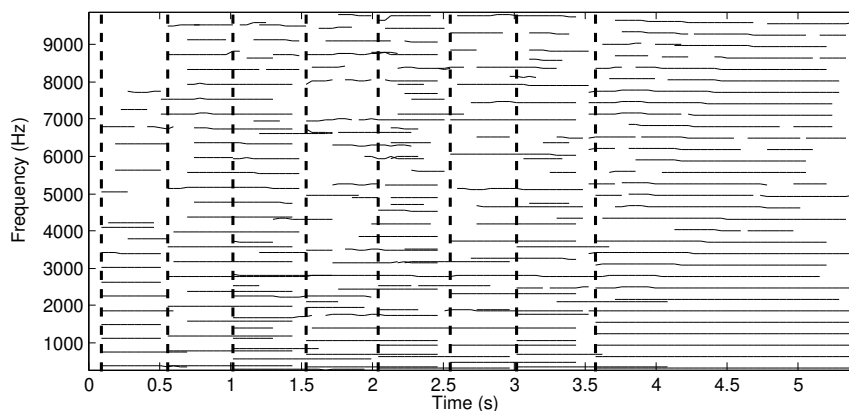
The instrument model library used for setups EXP 1 to EXP 6 was the same used for the evaluation of the model design in the previous chapter, consisting of 5 prototypes, namely piano, clarinet, oboe, trumpet, and violin. EXP 7 additionally uses a trained model of inharmonic bell sounds. Each model was learnt using the procedure detailed in Fig. 4.3 and throughout the previous chapter. They were trained with individual note samples ($f_s = 44.1$ kHz) from the RWC Musical Instrument Sound database [66] corresponding to the fourth octave (C4 to B4) and including all three different dynamic levels: forte, mezzoforte and piano. All piano, clarinet, oboe and trumpet samples correspond to the “normal” playing style, and violin was played without vibrato. The training parameters used were $G = 40$ frequency grid points, $D = 10$ PCA dimensions, and linear frequency interpolation for the time–frequency training data matrix.

In all experiments, cross-validation was ensured by testing with one instrument exemplar and training with the remaining exemplars of the RWC database. Piano, clarinet and violin are represented by 3 instrument exemplars; oboe and trumpet by 2. The total size of the database is of 414 files. Each experimental setup contains a collection of 10 mixtures for each degree of polyphony, except the sequence experiments EXP 3, EXP 3k and EXP 6, containing each 20 mixtures for each polyphony. This makes a total of 170 individual separation experiments.

5.2.2 Onset detection

Sinusoidal extraction is followed by a simple onset detection stage, based on counting the number of new tracks at any given time. The procedure will be illustrated by an example mixture consisting of a sequence of 8 notes played alternatively by a piano and by an oboe. Figure 5.2(a) shows the frequency support of the partial tracks resulting from the additive analysis of such a mixture.

Let $b(r)$ denote the function giving the number of tracks born at frame r . This function, normalized by its maximum, is plotted as the dashed line on Fig. 5.2(b). Taking this function as the onset detection function might work for simple mix-



(a) Sinusoidal tracks and detected onsets

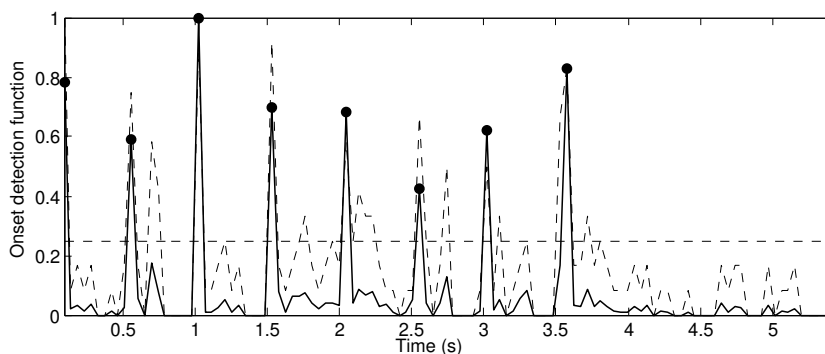
(b) Onset detection functions (see text): $b(r)$ (dashed line) and $o(r)$ (solid line)

Figure 5.2: Sinusoidal modeling and onset detection for an 8-note sequence of alternating piano and oboe notes.

tures where a robust sinusoidal analysis is possible. Very often, however, many high-frequency partials are only detected several frames after the real onset of the corresponding note. This is due to their more unstable nature (both in frequency and in energy) compared to the lower partials, particularly shortly after the attack. If several of such higher partials happen to appear for the first time at the same post-onset frame, they will produce a false peak on $b(r)$. This is especially noticeable in the second and sixth notes on the figure.

A preliminarily tested procedure to overcome this consisted of taking a moving average of $b(r)$ and taking the peaks of the resulting smoothed function as the position of the onsets (this was the method used in [32]). A more robust solution is to weight the contribution of each partial to the onset detection function by its frequency, so that the lower partials, that are supposed to be more stable, have a greater effect on establishing the detection peaks. In particular, the onset detection function $o(r)$ used was defined as

$$o(r) = \sum_{p \in \mathcal{N}_r} \frac{1}{\hat{f}_{pr}}, \quad (5.2)$$

where \hat{f}_{pr} is the estimated frequency of partial p at frame r and \mathcal{N}_r is the set of indices of the partials born at frame r . This function is shown as a solid line on Fig. 5.2(b), together with an appropriate peak-picking threshold. A threshold suitable for most of the experiments performed with the developed system was of 0.25 times the maximum value of $o(r)$. The peaks are then declared as the onset positions L_o^{on} for $o = 1, \dots, O$ (given in frames), which are denoted by the dashed lines on Fig. 5.2(a).

If a note is followed by another note of different pitch, new partial tracks will appear and contribute to the above onset function. The contribution will be greater the less partials of the new note overlap with the partials of the old note (i.e., the more dissonant their interval is). For highly consonant intervals, and even with unisons, the high amplitude sensitivity of the sinusoidal extraction will have already ensured that the overlapping partials have been split at the onset, and thus they will also contribute to $o(r)$.

An important point to note is that the time of the detected onsets will always be quantized with a resolution determined by the size of the analysis frames. In other words, even with an optimal onset detection, there can be a time error of up to one hop size. This is because the sinusoidal analysis method delivers frame-wise time-stamps, assigned to all partials deemed stable in the course of the corresponding analysis window. Since the preference is to have a relatively high frequency resolution for detecting close partials, such time quantization can be quite coarse (e.g., with the working sampling rate of 44.1 kHz, an FFT size of 8192 and a hop size of 2048 samples were used, which corresponds to a time resolution of 46.4 ms). Since the same time-stamp is used for resynthesis, the onset quantization can have a noticeable effect on the qualitative and quantitative separation results.

In spite of its simplicity, the presented onset detection approach was sufficient for the desired purposes. It was out of the scope of this work to evaluate in depth the onset detection quality, or to further improve it. It must be noted, however, that since the module is completely independent from the rest of the system (it only delivers a vector of frame positions), any other onset detection method can be used. A comprehensive review of onset detection methods can be found in [17].

5.2.3 Track grouping and labeling

To account for the above mentioned partial instability during the attack phase, all tracks \mathbf{t}_i having its first frame within the interval $[L_o^{\text{on}} - Q, L_o^{\text{on}} + Q]$ for a given onset location L_o^{on} are grouped into the set \mathcal{T}_o , where o is the onset index. A value of $Q = 2$ was chosen. A track belonging to this set can be of one of the following types:

1. *Nonoverlapping*: if it corresponds to a new partial not present in the previous track group \mathcal{T}_{o-1} , and does not overlap with any other partial belonging to the

same track group \mathcal{T}_o .

2. *Overlapping with previous track*: if its mean frequency is close, within a narrow margin, to the mean frequency of a partial from the previous track group \mathcal{T}_{o-1} .
3. *Overlapping with synchronous track*: if it corresponds to a new partial not present in the previous track group \mathcal{T}_{o-1} , and coincides in frequency, within a narrow margin, with a track belonging to the same track group \mathcal{T}_o .

Tracks of type 2 are detected, and correspondingly labeled, by searching the set \mathcal{T}_{o-1} for a track fulfilling the narrow frequency margin condition. More specifically, a track is labeled as overlapping with a previous-onset track if their mean frequencies differ by less than 40 cents².

Tracks of type 2 and 3 can be furthermore classified as resulting from overlaps between partials belonging to the same or different instruments. Whether a track of type 2 corresponds to the same or to different instruments is irrelevant for the present purposes since the corresponding notes will be segmented and separated anyway. On the other hand, tracks of type 3 belonging to the same instrument will be left intact without separation in order to detect same-instrument chords as belonging to a single source, allowing them to be separated as a single entity. Note that this separation goal differs from that of transcription or multipitch estimation, which would require to detect each and every constituent note of the chord.

The information available at this point of the system does not allow detecting if tracks of type 3 belong to the same or to different instruments, and in fact to distinguish between nonoverlapping tracks (type 1) and overlapping tracks of type 3. Preliminary tests were performed that consisted in matching individual tracks to the timbre models, but the results showed no sufficient robustness. This gave rise to the simplification of considering tracks of types 1 and 3 as belonging to the same source, which implies the current limitation of the system not supporting the separation of notes and chords from different instruments if they start within the same analysis frame. Such an onset separability constraint has to be assumed occasionally as a trade-off for separation quality in systems relying on onset detection [61, 173].

For each track set \mathcal{T}_o , a reduced set $\mathcal{T}_o^{\text{NOV}}$ was created by eliminating all the overlapping tracks of type 2 (NOV stands for “nonoverlapping”). This subset will be used for the timbre matching stage, to ensure that each onset-wise track group contains as many nonoverlapping tracks as possible in order to facilitate the matching with the time–frequency templates. The full set \mathcal{T}_o will however be stored and used afterwards in the track extension and resynthesis stages.

As a final step within the track grouping module, the offset L_o^{off} corresponding to a given onset L_o^{on} is declared as the last frame of the lowest-frequency partial of group \mathcal{T}_o . This decision is again based on the supposition that the lowest partials are the ones with the smoothest and most stable behavior.

²A *cent* is one hundredth of a semitone in a logarithmic scale. Taking the octave as basis, a cent corresponds to a frequency factor of $^{1200}\sqrt{2}$.

5.2.4 Timbre matching

The timbre detection stage compares each one of the onset-related, non-overlapping track groups $\mathcal{T}_o^{\text{NOV}}$ to each one of the prototype envelopes derived from the timbre models, and selects the instrument corresponding to the highest match. The overlapping tracks are discarded since their dynamic behaviour can significantly differ from that of the stored time–frequency templates. The output of this module is a set of labels, one for each onset, with the name of the instrument that has produced that note or chord. Although it is not the main goal of the present work, this means that running the system up to the timbre matching module can be used for polyphonic instrument recognition and segmentation (not however for multipitch estimation or transcription, since simultaneously playing notes by the same instrument will remain together as chords). In the context of source separation, this has the crucial consequence that a post-separation clustering of separated components or notes to sources is not necessary.

The core problem in this module is to design an appropriate distance measure between the track groups and the models. A similar situation appeared in Sect. 4.8, where the aim was to match partial clusters already separated by an independent separation method for the final purpose of polyphonic instrument recognition. In that case, an averaged Euclidean distance between the clusters and the time–frequency prototypes was used (Eq. 4.32). Here, that basic idea is further developed, enhanced and adapted to the proposed separation system.

The first measure tested, similar to Eq. 4.32, was the total Euclidean distance between the amplitude of each time–frequency bin belonging to a nonoverlapping track group $\mathcal{T}_o^{\text{NOV}}$ and the mean surface of the prototype envelope of instrument i evaluated at the frequency support of $\mathcal{T}_o^{\text{NOV}}$, denoted by $\tilde{\mathbf{M}}_{io}$. Such a distance can be rewritten here as

$$d(\mathcal{T}_o^{\text{NOV}}, \tilde{\mathbf{M}}_{io}) = \sum_{t \in \mathcal{T}_o^{\text{NOV}}} \sum_{r=1}^{R_t} |A_{tr} - \mathbf{M}_i(f_{tr})|, \quad (5.3)$$

where R_t is the number of frames in track $\mathbf{t}_t \in \mathcal{T}_o$ and A_{tr} and f_{tr} are the amplitude and frequency, respectively, on the r -th frame of that track. In order to obtain the evaluation at the frequency support $\tilde{\mathbf{M}}_{io} = \mathbf{M}_i(\mathbf{F}_o)$, for each data point the model frames closest in time to the input frames are chosen, and the corresponding values for the mean surface are linearly interpolated from neighboring data points.

A probabilistic reformulation of the matching distance allows taking into account not only the metric distance to the mean surfaces \mathbf{M}_i , but also the spread of their distribution, which was modeled as the variance surface \mathbf{V}_i . To this end, the distance-minimization problem was redefined as a likelihood-maximization or, in other words, a Maximum Likelihood decision. In particular, as measure of timbre similarity between $\mathcal{T}_o^{\text{NOV}}$ and the instrument model formed by parameters $\boldsymbol{\theta}_i = (\mathbf{M}_i, \mathbf{V}_i)$, the following likelihood function is used:

$$L(\mathcal{T}_o^{\text{NOV}} | \boldsymbol{\theta}_i) = \prod_{t \in \mathcal{T}_o^{\text{NOV}}} \prod_{r=1}^{R_t} p(A_{tr} | \mathbf{M}_i(f_{tr}), \mathbf{V}_i(f_{tr})), \quad (5.4)$$

where $p(x)$ denotes a unidimensional Gaussian distribution. The evaluation of the variance surface at the frequency support $\tilde{\mathbf{V}}_{io} = \mathbf{V}_i(\mathbf{F}_o)$ is performed in the same way as with the mean surface.

A requirement on both the metric and the probabilistic formulations in order for them to be generally applicable is that they should not be affected by the overall gain and by the length of the note or chord being classified. Gain invariance is required because the models were trained with different dynamic levels, and thus they are supposed to represent spectral shapes in a generalized way with respect to dynamic levels. In order to guarantee a correct optimization of the matching measures, a two-dimensional parameter search must be performed, with one parameter controlling the amplitude scaling and one controlling the time extent. Amplitude scaling is introduced by the additive parameter α and time scaling is performed by jointly, linearly stretching the partial tracks towards the offset. Then, the Euclidean-based measure becomes the optimization problem

$$d(\mathcal{T}_o^{\text{NOV}}, \tilde{\mathbf{M}}_{io}) = \min_{\alpha, N} \left\{ \sum_{t \in \mathcal{T}_o^{\text{NOV}}} \sum_{r=1}^{R_t} |A_{tr}^N + \alpha - \mathbf{M}_i(f_{tr}^N)| \right\}, \quad (5.5)$$

and the likelihood-based problem is

$$L(\mathcal{T}_o^{\text{NOV}} | \theta_i) = \max_{\alpha, N} \left\{ \prod_{t \in \mathcal{T}_o^{\text{NOV}}} \prod_{r=1}^{R_t} p(A_{tr}^N + \alpha | \mathbf{M}_i(f_{tr}^N), \mathbf{V}_i(f_{tr}^N)) \right\}, \quad (5.6)$$

where A_{tr}^N and f_{tr}^N denote the amplitude and frequency values for a track belonging to a group that has been stretched so that its last frame is N . To avoid rounding errors, the likelihood was computed in the logarithmic domain.

A further modification subjected to evaluation was a track-wise weighting such that lower-frequency and longer tracks have a greater impact of the matching measure than higher-frequency and shorter tracks. Such a weighted likelihood takes the form

$$L_w(\mathcal{T}_o^{\text{NOV}} | \theta_i) = \max_{\alpha, N} \left\{ \prod_{t \in \mathcal{T}_o^{\text{NOV}}} w_t \prod_{r=1}^{R_t} p(A_{tr}^N + \alpha | \mathbf{M}_i(f_{tr}^N), \mathbf{V}_i(f_{tr}^N)) \right\}, \quad (5.7)$$

where w_t is the track-dependent weight, which according to the above was defined as

$$w_t = e^{R_t / \bar{f}_t}, \quad (5.8)$$

where \bar{f}_t is the mean frequency of the track. A disadvantage of such a weighting is that pitch affects the likelihood, which will be higher for lower notes.

The three matching measures defined in Eqs. 5.5, 5.6 and 5.7 will be subjected to performance evaluation in the next subsection. But first, a simple example will serve to illustrate the timbre matching process. Figure 5.3(a) shows a good match between a track group belonging to a piano note (solid black lines) and a segment of the piano prototype envelope, with given amplitude scaling and time stretching parameters.

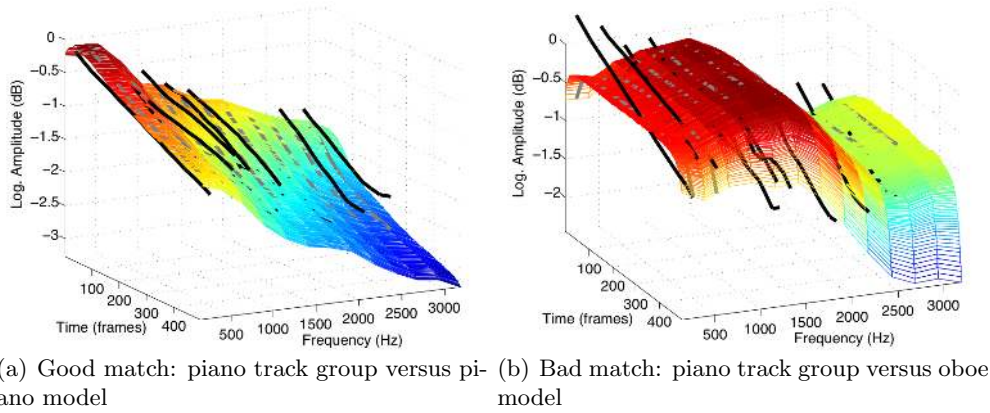


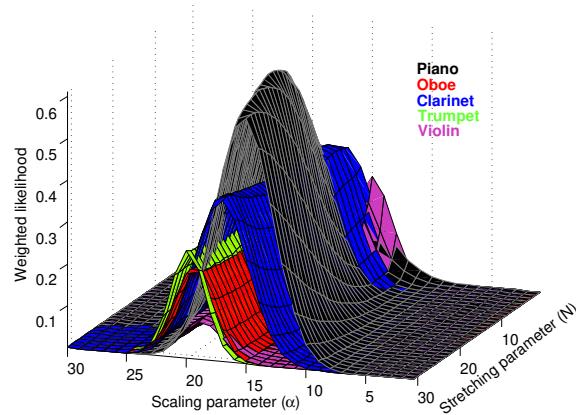
Figure 5.3: Examples of matches between track groups (solid black lines) and prototype envelopes.

The gray lines on the envelope surface correspond to the evaluation points of the envelope at the frequency support of the track group. It can be seen that the tracks have an overall strong similarity in both their frequency-dependent amplitude distribution and dynamic variation, in this case corresponding to an unsustained sound. In contrast, Fig. 5.3(b) is an example of weak match between the same piano track group and the oboe model. Both spectral shape and dynamic behavior differ significantly.

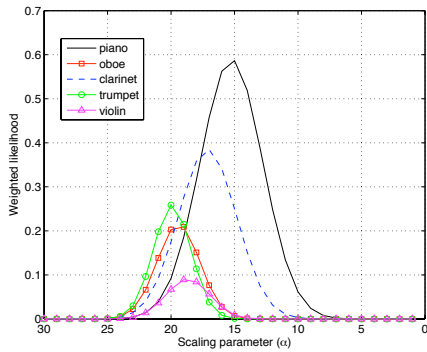
Figure 5.4(a) shows the matching surfaces produced by the exhaustive, two-dimensional parameter search (α, N) for the same piano note, compared with the previously library of 5 instrument models: piano, oboe, clarinet, trumpet and violin (non vibrato). The prototype curves corresponding to this database were shown in Figs. 4.18 and 4.19, and the time–frequency prototype envelopes in Fig. 4.20. In this example, the weighted likelihood of Eq. 5.7 was used as optimization measure. The class corresponding to the global maximum of each such collection of probability surfaces is assigned to the analyzed note. Figures 5.4(b) and 5.4(c) show representative projection profiles of the surfaces with fixed stretching and scaling parameters, respectively.

Timbre matching evaluation

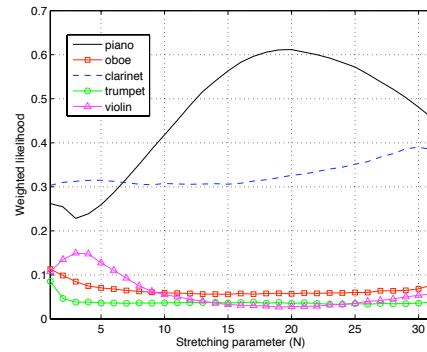
The separation performance of the system will obviously depend on its instrument classification performance. A wrongly classified note will be assigned to the wrong output source. Moreover, as it will be seen in the next section, its overlapping and incomplete tracks will be retrieved from the wrong model, which will artificially alter its timbre. From a different point of view, it is also interesting to test the timbre matching module as a polyphonic instrument detector by itself. An evaluation of the classification accuracy will thus help to assess these issues, and to select the most appropriate matching measure.



(a) Weighted likelihood optimization surfaces



(b) Amplitude scaling profile



(c) Time stretching profile

Figure 5.4: Examples of likelihood optimization results for a piano note.

The evaluations were based on the databases corresponding to the *basic* experimental setups EXP 1 to EXP 3, as defined in Table 5.1. Mixtures in EXP 1 and EXP 2 contain one single note played by each instrument, with different onsets. They differ in the harmonic relationships between them: in EXP 1, the used pairwise intervals are the ones that are considered more consonant (perfect fifth, major and minor thirds and major and minor sixths), and in EXP 2, the intervals are mostly dissonant (major and minor seconds, augmented fourths and major and minor sevenths). As was argued in the introduction of Chapter 2, predominantly dissonant mixtures are expected to be easier to separate than predominantly consonant ones, because of the higher degree of partial overlaps in the latter case. In all timbre matching experiments, the number and types of instruments out of the library of 5 models were unknown.

The sources making up EXP 3 contain sequences of more than one note per instrument, again at different onsets. Such mixtures are more demanding, since in order for the separation to be correct, each and every note of a given source must

Polyphony	Consonant (EXP 1)				Dissonant (EXP 2)			
	2	3	4	Av.	2	3	4	Av.
Euclidean distance	63.14	34.71	40.23	46.03	73.81	69.79	42.33	61.98
Likelihood	66.48	53.57	51.95	57.33	79.81	57.55	56.40	64.59
Weighted likelihood	76.95	43.21	40.50	53.55	79.81	77.79	61.40	73.00

Table 5.2: Instrument detection accuracy (%) for simple mixtures of one note per instrument.

Polyphony	Sequences (EXP 3)		
	2	3	Av.
Euclidean distance	64.66	50.64	57.65
Likelihood	63.68	56.40	60.04
Weighted likelihood	65.16	54.35	59.76

Table 5.3: Instrument detection accuracy (%) for mixtures of sequences containing several notes.

be correctly classified. Furthermore, the sequences were produced in such a way that the first of two consecutive notes played by the same instrument gets cut at the point where the second note starts, if the first did not yet finish. In such cases, the matching of the track group with the models will decrease in robustness.

The classification measure chosen for the experiments was the note-by-note accuracy, which is the percentage of individual notes correctly assigned to their instrument (this was the measure used in Table 4.4 of Sect. 4.8). Table 5.2 shows the results for the individual note experiments EXP 1 and EXP 2 using either the average Euclidean distance (Eq. 5.5), the Gaussian likelihood (Eq. 5.6) or the weighted likelihood (Eq. 5.7) as matching measures. The likelihood approach worked better than the distance in all cases, showing the advantage of taking into account the model variances. Using the track-wise length and frequency weighting in the likelihood clearly improves performance in the dissonant case. That is not the case, however, for high, consonant polyphonies. This can be explained by the fact that, in consonant intervals, it is very likely that the lowest-frequency partials of one of the notes are overlapping, and thus ignored for the matching, cancelling their proportionally more important contribution to the weighted likelihood as compared to the unweighted likelihood. In contrast, lowest partials in dissonant intervals are in fact very unlikely to overlap, and the overlapping will more commonly occur in higher frequencies. As expected, performance decreases with increasing polyphony and is better with dissonant than with consonant mixtures. The best performances were of 79.71% with 2 voices, 77.79% with 3 voices, and 61.40% for 4 voices.

Table 5.3 contains the results for the sequence experiments (EXP 3). Again, the likelihood approach outperforms the Euclidean distance. The improvement is however less important, and the difference in average accuracy between the weighted and non-weighted likelihoods is statistically negligible.

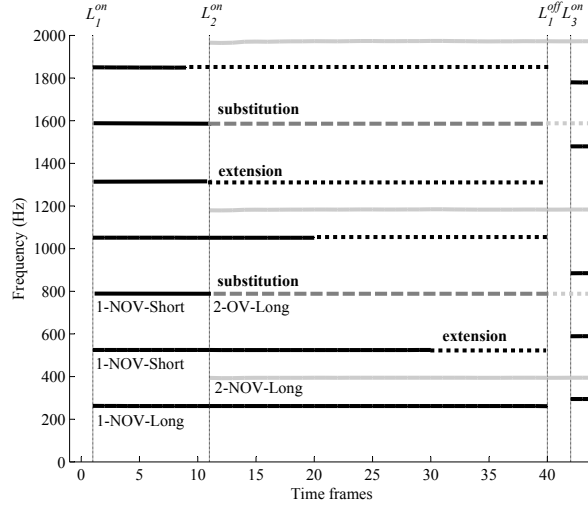


Figure 5.5: Illustration of track types and track extension/substitution for two notes separated by a perfect fifth.

5.2.5 Track retrieval

Once a non-overlapping track group $\mathcal{T}_o^{\text{NOV}}$ with onset L_o^{on} and offset L_o^{off} has been declared as produced by instrument i , the corresponding prototype envelope means \mathbf{M}_i are used for performing the two following operations on the corresponding full track group \mathcal{T}_o :

1. **Extension.** Tracks of type 1 or 3 that are shorter than the current note are extended forwards towards the offset (post-extension) or backwards towards the onset (pre-extension) by selecting the appropriate frames from \mathbf{M}_i and linearly interpolating the amplitudes at the mean frequency of the remainder of the track. Furthermore, the amplitudes retrieved from the model are scaled so that the amplitude transition between original and extended sections of the partial is smooth.

Tracks shorter than the note can result from either:

- a partial amplitude approximating the noise threshold in the region towards the offset and thus remaining undetected in the sinusoidal analysis stage (subjected to post-extension),
- the imminent appearance of a partial from the next onset group overlapping with it (subjected to post-extension), or
- a partial not being correctly detected during the first frames after the onset because of the instability of the attack phase (subjected to pre-extension).

2. **Substitution.** Overlapping tracks of type 2 are retrieved from the model in their entirety by interpolating the model at the frequency support of the

track. If the track is shorter than the note, it is again extended using the same procedure as above.

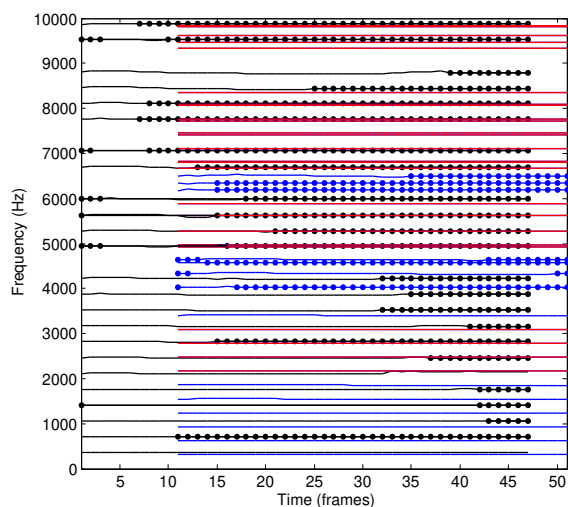
Figure 5.5 shows a schematic example of the results of the track extension block on the frequency support for each type of partial track. The example illustrates two partially overlapping notes separated by a perfect fifth. A representative selection of the partials are labeled according to type. The OV labels mean overlapping tracks of type 2 and NOV means nonoverlapping tracks (type 1). Short, nonoverlapping partials are extended towards the offset (marked by *extension*) and overlapping tracks of the second track group are marked by *substitution*. All the extensions shown in the example are post-extensions (which are by far more common than pre-extensions). Note that any region marked as *substitution* additionally implies a post-extension of the nonoverlapping tracks from the previous onset.

Figure 5.6 shows a more realistic application context of the extension and substitution module. Figure 5.6(a) shows the frequency support and Fig. 5.6(b) the time–frequency projection of a mixture of two notes. The first note (black tracks) is a clarinet playing an F4 and the second is an oboe playing an Eb4, which forms the highly dissonant interval of a major second. Pre- and post extensions corresponding to the clarinet tracks are marked by the black dots superimposed to the corresponding track segments. Non-overlapping tracks of the oboe note are denoted by blue solid lines, and their corresponding pre- and post-extension sections by blue dots. Overlapping tracks of the oboe (closer than 40 cents to a clarinet track) are displayed as solid red lines.

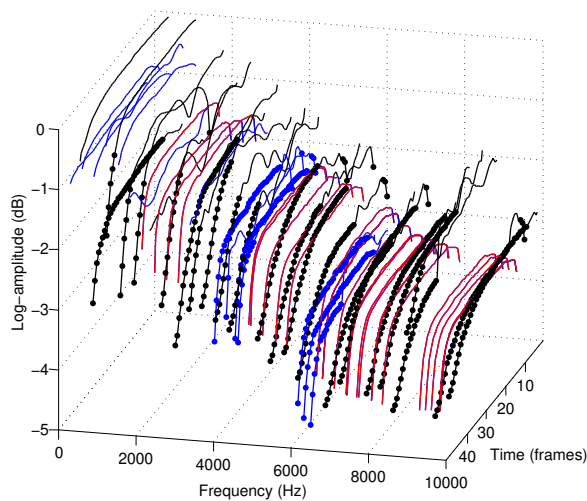
Because of the high degree of dissonance, most of the lowest-frequency partials do not overlap (the first overlap occurs at the 7th partial of the oboe). As can be seen in Fig. 5.6(b), the dynamic profiles of the overlapping tracks (red lines), entirely retrieved from the prototype envelope, show a smoother behavior, which corresponds to the averaged spectral envelopes obtained in the training. The same applies to the extended sections. As a result, the most overlaps and extensions, the more artificial and “canonical” will sound the separated note when resynthesized.

Resynthesis

At the final stage of the system, all reconstructed track groups belonging to the same instrument, as detected by the timbre matching stage, are concatenated and resynthesized using additive synthesis (see Sect. 4.2) to create the separated source corresponding to that instrument. The extended and substituted tracks do not preserve the phases, and thus a phaseless resynthesis was performed. This was however judged as being perceptually irrelevant. The already mentioned quantization of the onsets results in the separated notes being time-shifted with respect to the original notes by up to one hop size (46.4 ms) using the above mentioned analysis parameters.



(a) Frequency support



(b) Time-frequency view

Figure 5.6: Application example of track extension/substitution for a mixture of 2 notes.

5.3 Evaluation of separation performance

The separation performance of the system depends on the individual performances of the subsequent modules, most importantly from the onset detection and timbre matching stages. An undetected onset is especially costly, since the entire corresponding note will be absent from the separated signal and thus will highly degrade objective measures such as Signal to Error Ratios. False-positive onsets (i.e., the detection of an onset that does not correspond to a note start) are much less harmful, since their track groups will be mixed with the correctly detected part of the note

at the appropriate amplitude level, as long as they have been correctly assigned to their instrument. As for the timbre matching stage, wrong classifications have two harmful effects: the false assignment of a note to a separated track, which decreases the performance in the same order of magnitude than undetected onsets, and the extension and/or substitution of tracks with a false timbre model, which is far less perceptually and objectively relevant.

As was introduced in Sect. 3.5.2, a frequently used objective measure of the separation quality is the time-domain *Signal to Error Ratio* (SER), defined in Eq. 3.44. When separation is based on additive resynthesis, the time-domain subtraction to obtain the error signal is only valid if the algorithm preserves the phases [14]. The proposed system does not provide phase information in the track segments that have been retrieved from the time–frequency models, and thus a direct application of SER will produce misleading results. Instead, a *Spectral Signal to Error Ratio* (SSER) will therefore be used, defined as the SER between the magnitude of the STFT of the original source $S(r, k)$ and the separated source $\hat{S}(r, k)$:

$$\text{SSER} = 10 \log_{10} \frac{\sum_{r,k} |S(r, k)|^2}{\sum_{r,k} (|S(r, k)| - |\hat{S}(r, k)|)^2}. \quad (5.9)$$

Another motivation for performing the comparison in the frequency domain arises from the already mentioned onset time quantization of up to one hop size. Such onset indeterminacy is not critical from the point of view of perceptually or aurally assessing the separation success, but can have a noticeable impact on the sample-by-sample, time-domain error measure (note that the indeterminacy occurs at the most energetic part of the signals). Using the spectral measure SSER with the same STFT parameters that were used for the sinusoidal modeling ($f_s = 44.1$ kHz, window and FFT sizes of 8192 samples or 185.7 ms and hop size of 2048 samples or 46.4 ms) avoids this by performing an averaged error computation along each analysis frame. Note also that, for the same reasons, the other quality measures introduced in Sect. 3.5.2 are not directly usable within this context since they are defined in the time domain.

The separation performance was evaluated with all the mixture types listed as experimental setups in Table 5.1. Each degree of polyphony in each experimental setup was tested with a collection of 10 mixtures, giving a total number of 170 separation experiments. The final performance measures appearing in all tables in the remainder of the chapter correspond to the averaged SSER values across all separated sources of each particular setup’s mixtures whose onsets have been correctly detected.

The following four subsections address the results for mixtures consisting of one note per instrument (Sect. 5.3.1, EXP 1 to EXP 2), of a sequence of notes per instrument (Sect. 5.3.2, EXP 3k), for mixture containing chords or clusters (Sect. 5.3.3, EXP 4 to EXP 6) and for mixtures containing inharmonic sounds (Sect. 5.3.4, EXP 7).

5.3.1 Experiments with individual notes

For the first and simplest evaluation test, 60 mixtures of single notes from 2, 3 or 4 different instruments were considered, 30 of them composed by consonant intervals (EXP 1), and the other 30 by dissonant intervals (EXP 2), the same that were used in the timbre matching evaluation. Each one of such mixtures makes up a multi-timbral arpeggio³, an example of which is shown in Fig. 5.7(a). For this experimental setup, the instruments contained in the mixture were considered unknown and belonging to the 5-class instrument library specified in Sect. 5.2.1.

Figure 5.7 illustrates the separation results for an instance of this experimental setup. Figures 5.7(a) and 5.7(b) show the input mixture in musical notation and as a waveform, respectively. It should be noted that all musical scores shown in the remainder of the chapter were created a posteriori for illustration purposes and to allow a rapid overview of the involved pitches and the demands of the experiment in question. The note durations should be thus considered approximate with respect to those of the real original sources. Figure 5.7(c) shows the segmentation chart resulting from the timbre matching module, indicating the frame ranges in which the different instruments are present. Black frames denote the detected onsets. The width of those onset frames serve as an indication of the degree of quantization of the resynthesized onsets. The number of rows in the segmentation chart corresponds to the number of instruments in the timbre template database. In this particular example, the oboe has been misclassified as a trumpet. Finally, Fig. 5.7(d) shows the waveforms of the separated and resynthesized sources.

The first two rows in Table 5.4 show the results of averaging all SSER values for each detected source and for all mixtures, for the consonant and dissonant cases, respectively. The numerical values obtained obey the expected behavior: separation quality decreases with increasing polyphony and increases with dissonance. The difference between consonance and dissonance is higher with lower polyphony. This shows that polyphony has a considerably stronger effect on increasing the degree of overlapping, and thus separation difficulty, than the nature of the harmonic relationships. It can be expected that with polyphonies of 5 notes or higher, consonant and dissonant mixtures will be equally difficult to separate.

5.3.2 Experiments with note sequences

For the next set of experiments, 40 mixtures of 2 and 3 instruments playing short melodic sequences of individual notes were used. This time, the instruments were known a priori (EXP 3k). As noted before, this situation is more demanding, mainly because all notes from each instrument need to be correctly detected and classified in order to be clustered into the same separated track. Figure 5.8 shows an example of the kind of mixture this set of experiments is dealing with.

The averaged SSER results are in the third row of Table 5.4. It is surprising that, while similar, the results with 3-note polyphony are slightly better than with 2-note

³An *arpeggio* is a *broken chord*, i.e., a chord in which the constituent notes are played in succession rather than simultaneously.

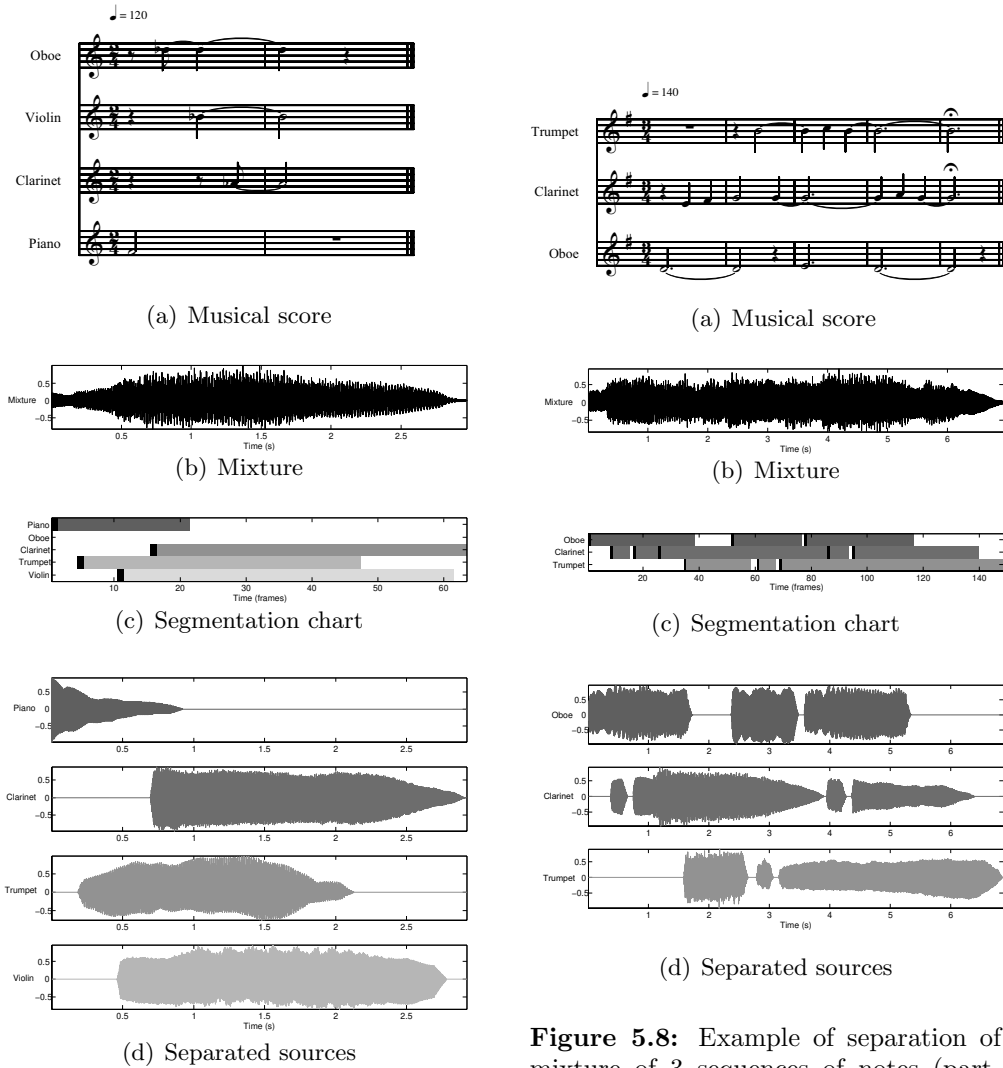


Figure 5.7: Example of separation of an individual-note, four voice mixture (part of EXP 1).

Figure 5.8: Example of separation of a mixture of 3 sequences of notes (part of EXP 3k).

polyphony. This is most probably a consequence of the disparity of the created test mixtures with respect to length, note intervals, tempo and rhythmic relationships, which partially hinders the evaluation of polyphony as an independent parameter. In contrast, EXP 1 and EXP 2 contained similar mixtures with well-defined constraints and characteristics, making their averaged evaluation more statistically significant.

5.3.3 Experiments with chords and clusters

A novelty of the presented approach compared to previous separation systems based on sinusoidal modeling is that it is able to separate groups of simultaneously sound-

Source type	Polyphony		
	2	3	4
Individual notes, consonant (EXP 1)	6.93 dB	5.82 dB	5.35 dB
Individual notes, dissonant (EXP 2)	9.38 dB	8.36 dB	5.95 dB
Sequences of notes (EXP 3k)	6.97 dB	7.34 dB	-

Table 5.4: Results (averaged SSER) for the basic experiments.

ing notes (i.e., chords⁴) produced by a single instrument. As has already been noted, this is because both timbre matching and track retrieval are based on tracks grouped solely following common-onset and common-dynamic-behavior criteria. No harmonic or quasi-harmonic relationships are required for a track to be grouped into a common-onset, same-instrument separated entity.

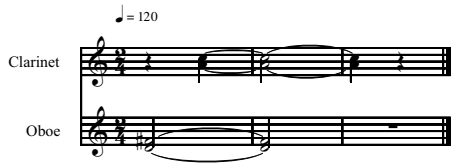
To illustrate and evaluate this capability, the “extended” experimental setups EXP 4 to EXP 6 were defined. As counterparts of, respectively, EXP 1 and EXP 2, both EXP 4 and EXP 5 include short mixtures of only one chord instance per instrument. The difference between both mixture types pertains again to the harmonic nature, either consonant or dissonant. When dealing with chords however, an additional consideration must be taken into account. In this case, there is a distinction between intra-class harmony (i.e., the harmonic relationships between the constituent notes of a chord), and inter-class harmony (between different chords played by different instruments).

Inter-class dissonance, as in the individual-note case, will result in less overlaps and better separation. On the contrary, as it will be seen, the effect of intra-class dissonance is exactly the opposite. If a chord contains several notes in highly dissonant mutual relations, its corresponding track group will contain many non-overlapping tracks. These will cover the frequency range more tightly and make collisions with the next chord’s tracks more probable, hindering separation. Consonant chords, in contrast, will have many tracks overlapping in the high-energy area, and will thus leave empty frequency gaps to be filled by tracks of the upcoming chord.

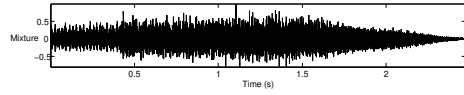
To test these new harmonic implications, EXP 4 was based on 20 mixtures of 2 and 3 instruments playing chords with mostly consonant intra-class intervals (such as major and minor triads, and seventh chords, see Fig. 5.9(a) for a two-note, two-chord example), and EXP 5 on 20 mixtures with the most internally-dissonant chords possible: clusters⁵. Figure 5.10 shows the separation of a mixture of a trumpet chromatic cluster comprising all notes between A4 and C5, and of a diatonic 3-note cluster played by the piano. The averaged results shown on the first two rows of Table 5.5 confirm the higher difficulty of separating consecutive clusters. These experiments were performed with the instruments unknown, as can be observed from the empty rows on the segmentation charts.

⁴The usual musical definition of *chord* is a group of three or more simultaneously sounding notes. A group of two simultaneous notes is more appropriately termed a *dyad*. For simplicity, “chord” will be used here in either case.

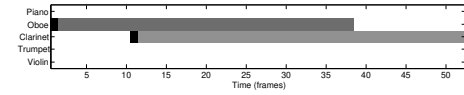
⁵A musical *cluster* is a chord containing notes that are consecutive on a given scale. A chromatic cluster is a special case in which the notes are adjacent semitones, and a diatonic cluster is the



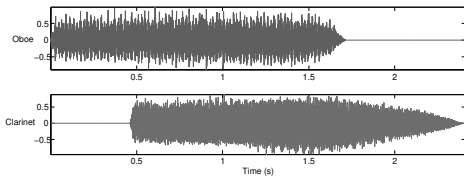
(a) Musical score



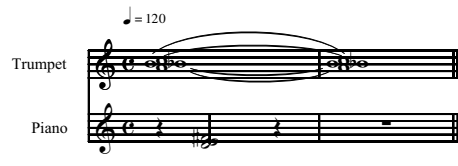
(b) Mixture



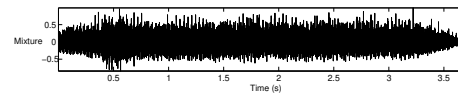
(c) Segmentation chart



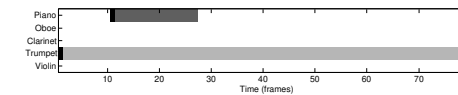
(d) Separated sources



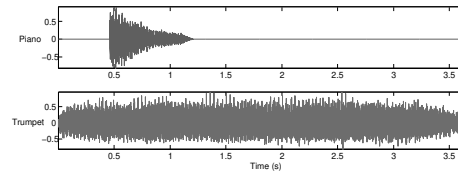
(a) Musical score



(b) Mixture



(c) Segmentation chart



(d) Separated sources

Figure 5.9: Ex. of separation of a mixture of two consonant chords (part of EXP 4).

Figure 5.10: Example of separation of a mixture of two clusters (part of EXP 5).

Source type	No. Instruments	
	2	3
One chord (EXP 4)	7.12 dB	6.74 dB
One cluster (EXP 5)	4.81 dB	4.77 dB
Sequences with chords and clusters (EXP 6)	4.99 dB	6.29 dB
Inharmonic notes (EXP 7)	7.84 dB	-

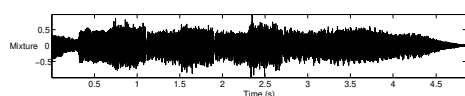
Table 5.5: Results (averaged SSER) for the extended experiments.

As a counterpart to EXP 3, a set of 20 mixtures of sequences, this time including chords, was generated and evaluated as EXP 6, with the instruments known. Test mixtures included chord-only sequences (such as the one in Fig. 5.11) and hybrid sequences containing both chords and individual notes (such as in Fig. 5.12). The averaged SSER results included on the table accentuate the mixture disparity problem mentioned in the last section.

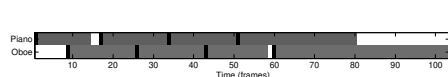
equivalent with whole-tones.



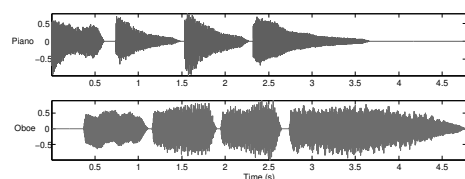
(a) Musical score



(b) Mixture



(c) Segmentation chart

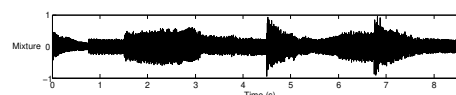


(d) Separated sources

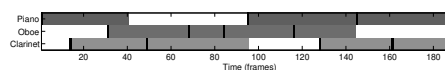
Figure 5.11: Example of separation of a mixture of two chord sequences (part of EXP 6).



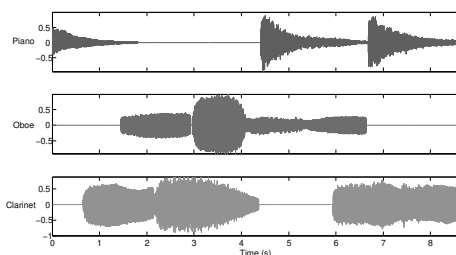
(a) Musical score



(b) Mixture



(c) Segmentation chart



(d) Separated sources

Figure 5.12: Example of separation of a mixture of sequences including chords (part of EXP 6).

5.3.4 Experiments with inharmonic sounds

The same working principle that allows the separation of same-instrument chords as single entities, namely the fact that grouping is only based on the amplitude behavior of the partials, and not on their frequency characteristics, allows dealing with sounds containing partials that do not follow a harmonic or even quasi-harmonic frequency positioning. The only difference to be taken into account is that the additive analysis stage needed to perform the training corresponding to such an instrument must be performed in inharmonic mode (see Sect. 4.2), without an f_0 -based prediction of the partial frequencies.

To the aim of demonstrating this capability, a new timbre model was trained using a collection of 36 tubular bell samples from the RWC database [66], with predominant pitches C4 to B4. Then, for the final set of experiments (EXP 7), 10 mixtures of individual notes played by the bells and one of the other 5 instruments were created. The quantitative averaged result shown on the last row of Table

5.5 shows a quality halfway between the individual-note, consonant case and the individual-note, dissonant case.

5.4 Conclusions

The discussions and results that have been presented in this chapter show that an accurate description of the spectral envelope and its dynamic behaviour, based on the models presented in Chapter 4, is sufficient to make the demanding task of single-channel source separation possible. No explicit assumptions about the frequency contents (pitch, harmonicity) were made; the system solely relies on the amplitude characteristics of the partial tracks, exploiting the CASA-like grouping principles of good continuation and common fate, together with their group-wise matching with a set of stored probabilistic templates. Experiments using mixtures of up to 4 notes from up to 5 instruments, including mixtures with single-instrument chords, have been shown to demonstrate the viability of the method.

The main features of the proposed separation system can be summarized as follows:

- **No pitch information required.** In contrast to many previous approaches, the system does not require a priori knowledge about the pitches present in the mixture [61, 62, 109], nor a preliminary multipitch estimation stage [14, 168, 172], which is often a source of unrobustness.
- **No harmonicity assumed.** Instead, track grouping and reconstruction is only based on common-onset and amplitude continuity cues. This allows separating highly inharmonic sounds and detecting single-instrument chords as single entities. These results were not possible with the wide range of previous systems that assume perfect or approximate harmonicity [61, 62, 103, 169, 172, 173, 174].
- **Accurate spectral modeling.** The timbre models used describe in detail the temporal evolution of the spectral envelope. This contrasts with simpler models that assume a set of fixed spectral shapes as the timbre library [14], or a multiplication of static spectral envelopes by a time-varying gain [166, 169].
- **Source identification.** Due to the timbre matching stage, the system outputs the name of the musical instrument each source belongs to. Thus, it can be used for polyphonic instrument recognition. The maximum accuracies were of 79.81% correctly detected notes for a 2-voice polyphony, of 77.79% for 3 voices, and of 61.40% for 4 voices.
- **Provides segmentation data.** Each detected common-onset group is processed and reconstructed separately, so that the system can output the start and ending points of each played note, making it appropriate as a pre-processing step for polyphonic transcription.

- **No note-to-source clustering needed.** Due to the fact that each note is assigned to an instrument, clustering is implicitly accomplished by the timbre matching stage, and thus not needed as a separate final stage, unlike most methods relying on basis decomposition [1, 39, 166, 176].

As a trade-off for not using any harmonicity cue, the main limitation of the system is that it is not able to handle common-onset separation if the corresponding notes have been played by different instruments (if they are played by the same instrument, they are considered a chord and thus they are left mixed). A first direction towards relaxing such an onset separability constraint [61, 173] was to match the tracks to the timbre models individually, rather than in common-onset groups, and declaring an onset group as a mixture of two instruments if the individual track classification result was spread across the corresponding classes. Although some success has been obtained using this approach, it showed little robustness in preliminary experiments. Another possibility would be to train joint models of the combination of several timbres and add them to the library for model matching.

The modular architecture of the system has the drawback that the overall success highly depends on the robustness of the consecutive processing stages, mainly of onset detection and timbre classification, and on the quality of the pre-trained timbre models. On the other hand, performance can be further improved by using more sophisticated onset detection methods or by exploring new timbre similarity measures. These and other possibilities for future improvements will be further discussed in Sect. 7.2.

In the next chapter, the present system will be extended to stereo mixtures, so that the additionally available spatial information will be able to further facilitate and generalize separation.

6

Extension to stereo mixtures

The highly underdetermined nature of the monaural separation problem, together with the generality imposed by not assuming harmonicity cues, resulted in some constraints concerning the applicability of the separation system presented in the previous chapter. An important issue was the onset separability constraint: two notes from different instruments cannot be detected separately if they start within the same analysis window. Also, both the robustness of the system, and its separation quality, depend on the accumulated success of each individual stage of onset detection, timbre matching and track retrieval. For example, clustering of notes into sources is performed by classification in the timbre matching stage, and its failure will have an important negative effect on the separation quality.

Such problems can be greatly reduced if the separation process can take into account spatial cues resulting from a multichannel mixture. The purpose of this chapter is to combine the blind stereo separation system presented in Chapter 3, which was solely based on sparsity and spatial information, with the ideas developed in Chapters 4 and 5 concerning supervised detection and separation of monaural signals according to common-onset and timbral cues. In particular, two different methods for developing such a hybrid system will be proposed, evaluated and discussed. Note that, in this context, the word “hybrid” denotes the combination of elaborate source modeling techniques with the exploitation of spatial cues [164].

The first, simpler, method consists in applying the monaural separation system of Chapter 5 to each one of the output channels¹ of the stereo BSS system of Chapter 3, with minor modifications concerning classification decisions. This approach will be introduced in Sect. 6.2. The second makes use of sinusoidal subtraction to perform a more elaborate refinement of the BSS output, and will be addressed in Sect. 6.3. The classification (timbre matching) stage, common to both systems, will be evaluated in Sect. 6.4. Separation experiments and corresponding evaluations will be detailed, for each of the methods separately, in Sect. 6.5. But before, some hybrid approaches from the literature will be briefly presented.

6.1 Hybrid source separation systems

After having reviewed previous work proposing basic blind approaches to source separation (Sects. 2.6 and 2.7), and monaural systems based in more advanced models

¹It should be noted that “channel” will be used to denote the output signals of the BSS stage, and “track” will always denote sinusoidal tracks defined by the partials.

or a priori knowledge (Sect. 5.1), this section will introduce systems extending the latter group to the stereo ($M = 2$) and multichannel ($M > 2$) cases. In comparison with the monaural case, few works have addressed an extension of advanced generative models to the multichannel scenario.

Vincent and Rodet [167] develop a generative source model defined as a three-layer Bayesian Network that is combined with spatial diversity information provided by the IIDs for MAP estimation. From bottom (low-level) to top (high-level), the three layers are: spectral layer, which uses an ISA spectral basis decomposition model (see Sect. 5.1); descriptor layer, which collects the time-varying weights of the ISA decomposition together with an energy factor, all assumed to be Gaussian; and state layer, which models the presence or absence of a note at a particular instant, following a HMM. The model parameters are learnt on a database of single-channel solo excerpts. The results show that combining spatial information with the generative source models improve average separation quality by 2.7 dB over using only spatial cues, and of 9.7 dB over using the source models alone. This approach was extended in [161] by considering delayed mixtures and by using a more sophisticated model for the state layer called *segmental model*, which adds a temporal persistence prior. According to the authors, this was the first system capable of separating mixtures with long reverberation.

Viste and Evangelista [175] present a system based on sinusoidal modeling that exploits spatial cues and similarity of the sinusoidal amplitude tracks belonging to the same note. Non-overlapping partials are detected assuming harmonicity and are used as models for the amplitude envelope of the overlapping partials. Then, for each bandpass frequency region containing a set of overlapping partials, a spatial unmixing matrix is searched so that it maximizes the similarity of the unmixed partial tracks with the corresponding partial models.

Nakatani [114] introduces source localization cues into a CASA architecture aimed at speech segregation, most other examples of which are based on monaural mixtures. The mixture is assumed to be delayed, and both IID and IPD are exploited for obtaining a measure of spatial diversity that is considered in the hypothesis scoring.

Sakuraba [133] uses IID and IPD cues to group the partials according to single notes, together with an harmonicity assumption. To perform the note-to-source clustering (which is called *sequential grouping* in the cited work), spatial proximity is again a criterion, which is furthermore combined with a measure of timbre similarity based on a set of temporal and spectral features given as input to a *Support Vector Machine* (SVM).

6.2 Stereo separation based on track retrieval

As has been mentioned throughout this work, underdetermined blind separation methods, without a priori knowledge about the sources, can attain reasonable performances with well-spaced instantaneous multichannel mixtures. The basic idea behind the methods proposed in this chapter is to perform BSS as a first, coarse separation stage, followed by a channel-wise refinement using the methods intro-

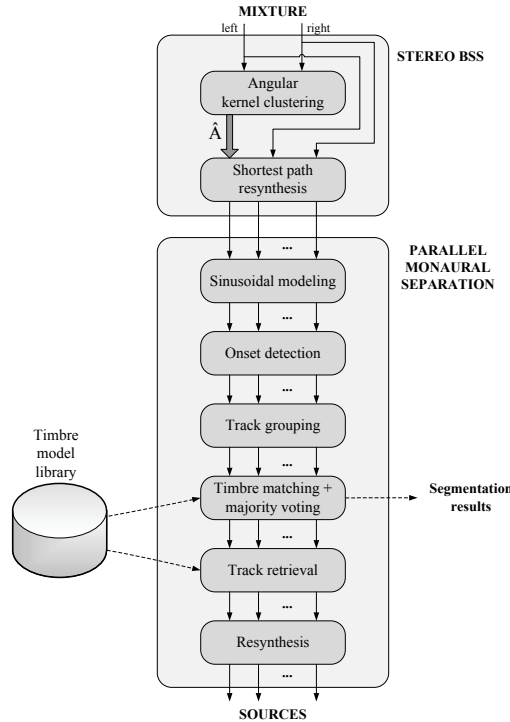


Figure 6.1: Overview of stereo separation based on track retrieval.

duced in the previous two chapters, including sinusoidal modeling, timbre matching and common-onset grouping.

The Bofill and Zibulevsky BSS approach [23, 24] used for the warped-frequency separation experiments in Chapter 3 showed an overall satisfying and robust performance concerning source detection (see Table 3.4) and separation (Table 3.5). It consists of the successive stages of sparse transformation, mixing matrix estimation using kernel-based angular clustering, shortest-path resynthesis based on ℓ_1 norm minimization and inverse sparse transformation. After processing with that system, separated channels still contain artifacts and interferences from other sources, depending mostly on the degree of polyphony and on inter-channel closeness in the stereo field. The goal of applying the mentioned advanced modeling techniques is to diminish the effect of such residual components.

The first evaluated method to that aim consists of applying in parallel the monaural processing chain of Fig. 5.1, with some slight modifications, to each channel preliminarily separated by the BSS stage. The resulting scheme is shown in Fig. 6.1, where $\hat{\mathbf{A}}$ denotes the estimated mixing matrix. Note that, for clarity, the direct/inverse sparse transformation stages before and after BSS have been omitted on the figure. It should be noted however, that the BSS processing stages take place in the sparse domain.

In spite of the probably important interference residual still present after BSS,

this scheme dramatically increases the robustness and capabilities of the system compared to the monaural case. The reasons for this are the following:

- Sinusoidal modeling is now applied on partially separated channels, rather than on a full mixture. This allows finding better additive analysis parameters for a smoother and more robust detection of the sinusoidal tracks. The same applies to onset detection, which will expectedly increase in robustness. Nevertheless, it was decided to keep the inharmonic analysis mode in order to still be able to handle same-instrument chords and inharmonic sounds and to avoid a previous fundamental frequency analysis.
- Overlapping partials have been already partially separated in the BSS stage by an amplitude subtraction weighted by the coefficients of the reduced unmixing matrix \mathbf{A}_p^{-1} , as defined in Eq. 3.43. A crucial consequence of this is that in the timbre matching and track retrieval stages, all onset-synchronous predominant partials can be assumed to belong to the same instrument and thus separation of different-instrument notes starting at the same onset will be possible. In other words, and using the track typology of Sect. 5.2.3, sinusoidal tracks of type 3 are resolved by the BSS stage. Also, this is expected to increase the note-by-note timbre matching accuracy, since the overall dynamic behavior of the track groups will more closely resemble the prototype envelope.
- Assuming each source contains notes played only by one instrument, the note-to-source clustering is now performed by the BSS stage following spatial criteria, and not by timbre matching. As long as the mixture is instantaneous with sufficiently spaced sources, such a BSS-based clustering is extremely reliable. Assignment of an instrument label to a separated source can be thus performed by majority voting among all note-wise classifications of that channel.

Such an approach will be evaluated in Sects. 6.4 and 6.5 for respectively, classification and separation performance. A disadvantage of the method is that, like in the monaural case, separated sources are constituted only by the sinusoidal part, and the contribution of the noise part is ignored. Also, the onset/offset uncertainty mentioned in Sect. 5.2.2 is still present.

6.3 Stereo separation based on sinusoidal subtraction

The track retrieval approach, a direct extension of the previous chapter's monaural separation method to stereo mixtures, is able to suppress interferences after BSS, but has the drawback that, since the separated sources are resynthesized from the retrieved partial parameters, they lack the noise part, which contributes to the perceived quality of the separated signals. This was also a disadvantage of the monaural separation method.

As was seen on the literature review section of the previous chapter (Sect. 5.1), most single-channel separation approaches are solely based on sinusoidal resynthesis. An exception is the system proposed by Every and Szymanski [62], which is based on

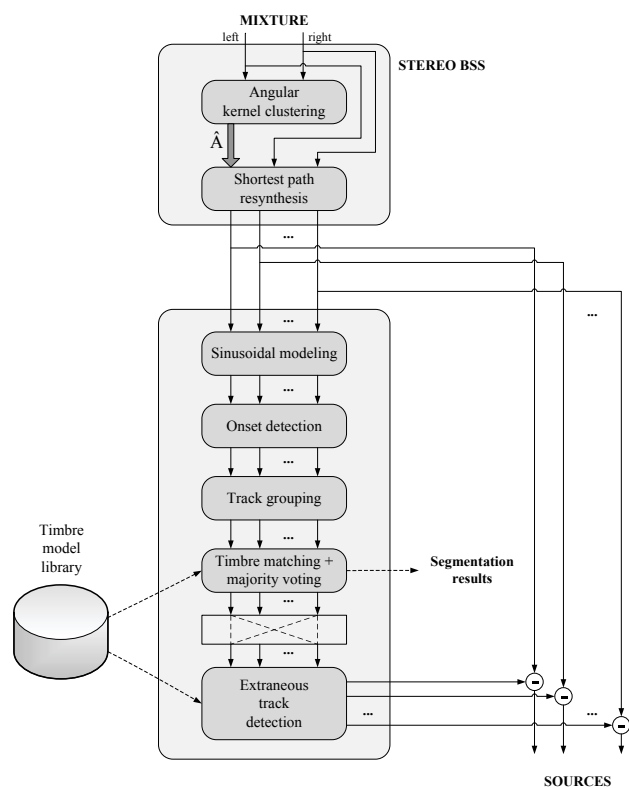


Figure 6.2: Overview of stereo separation based on sinusoidal subtraction.

subtracting interfering sinusoids from the spectrum of the mixture, thus leaving the noise regions between sinusoids unaltered. This means that the resulting sources have separated sinusoidal parts, but a common, non-separated noise part. If the noise content of the original sources is moderate, the result of such a separation is nevertheless more perceptually satisfactory than a plain additive resynthesis. In another work by the same authors [61], several methods to separate the noise part are proposed, based on transient detection and on an assumed strong correlation between the spectral envelopes of the noise floor and of the sinusoidal peaks.

In the stereo setup proposed here, it is possible to take advantage of the fact that the BSS stage partially separates the input mixture as a whole, both its sinusoidal and noise parts. The idea is to use sinusoidal subtraction techniques applied to the partially BSS-separated channels. This, again, will not change the noise floor between sinusoidal main lobes, but in contrast to the monaural case, the noise part is already partially unmixed, and the noise part of a given separated source is assumed to have a higher energy than the noise parts of the interfering sources.

More specifically, for a given BSS partially separated channel, the aim is to detect and eliminate extraneous sinusoidal tracks caused by the remaining interferences of the other channels. To that end, the monaural processing modules of

sinusoidal modeling, onset detection, track grouping and timbre matching/majority voting, are again applied in parallel to detect the instrument that is playing on each channel (see Fig. 6.2). However, instead of using the detected models for extending and substituting overlapping tracks, they are used to detect tracks that have probably been produced by another instrument, which will then be labeled as extraneous. Furthermore, one can use the other partially-separated channels to make more robust decisions about the interfering tracks, as will be detailed below. This inter-channel dependency is denoted in the chart figure by the “switch-matrix”-type symbol between the timbre matching and the extraneous track detection modules. The detection of extraneous tracks will be addressed in more detail in Sect. 6.3.1. Finally, each set of extraneous sinusoidal tracks is subtracted from the partially-separated sources², thus reducing interferences and improving separation quality.

Apart from the benefits of the track-retrieval-based method compared to the monaural case (more robust sinusoidal modeling and onset detection, same-onset separation possible, better classification performance), the sinusoidal subtraction method has the following additional advantages:

- The (partially separated) noise content is kept in the separated sources.
- Possible bad performance of the onset detection and timbre matching modules has a less costly effect on the separation quality. For instance, a false-positive onset can result in failing to detect several extraneous tracks that will not be subtracted. In contrast, in both monaural and stereo versions of the track retrieval system, false-positive onsets can generate full notes inexistent in the original mixture. A wrong classification will again lead to wrong labelling of a few extraneous tracks, but in track retrieval it led to the generation of wrong spectral shapes in the resynthesis.
- The onset/offset uncertainty of additive resynthesis does not apply to separated notes, whose timing is not altered at all. It will only result in short fragments of extraneous tracks not being effectively subtracted.
- The phases are kept all along the algorithm, allowing the usage of the well-established time-domain separation measures of *Source to Distortion Ratio* (SDR), *Source to Interference Ratio* (SIR) and *Source to Artifacts Ratio* (SAR), introduced in Sect. 3.5.2.

6.3.1 Extraneous track detection

Crucial to the success of the subtraction approach is a reliable detection of the sinusoidal tracks that are supposed to originate from interfering sources. The decision is based on three criteria: temporal, timbral and comparison between BSS-separated channels. They will be introduced in the following subsections.

²The subtraction was performed with IRCAM’s pm2 software, as was the additive analysis.

Temporal criterion

The first criterion to decide if a track is to be considered extraneous is its temporal location with respect to the detected onsets and offsets. Correct (non-extraneous) tracks are assumed to start and end within one of the onset/offset intervals of its channel, with a tolerance of a few analysis frames. Every track that does not fulfill this, i.e., that starts considerably before an onset and ends considerably after it, is automatically labeled as extraneous, independently from the two subsequent criteria.

Timbral criterion

The second criterion is timbre similarity with respect to the timbre model assigned to the current channel by the timbre matching and majority voting stages. All tracks with a similarity lower than an appropriate threshold will be marked as extraneous. The similarity measure chosen, as in the timbre matching stage, is based on a joint Gaussian likelihood considered along all points of the track, and with amplitudes linearly interpolated from the corresponding timbre model at the track's frequency support. Such a likelihood was the basis of the timbre similarity measures introduced in Eqs. 5.6 and 5.7. However, tracks must now be matched individually, rather than by common-onset track groups. This requires several additional considerations.

In the timbre matching stage, tracks were grouped according to onsets and matched together with the models. In other words, the used similarity measures were combined or averaged across all bins corresponding to all tracks of a given group. In that context, it was a reasonable option to assign weights to individual tracks within a group, such that longer and lower-frequency tracks have a greater impact on that particular group-wise likelihood value. This was the case for the weighted likelihood defined in Eq. 5.7. On the contrary, when tracks are matched individually, they cannot use this kind of weighting. As long as it is located between a given onset/offset pair, the length of a track is not a reliable indication of its extraneousness, since it will depend on how well could it be detected by the Sinusoidal Modeling stage, which in turn will depend on both the original amplitude of the track in the mixture, and on the unmixing factor detected in the mixing matrix estimation stage of the BSS block.

In fact, the length-dependency of the new track-wise likelihood must be now explicitly cancelled by taking the geometric mean:

$$L(\mathbf{t}_t|\boldsymbol{\theta}_i) = \left[\prod_{r=1}^{R_t} p(A_{tr}|\mathbf{M}_i(f_{tr}), \mathbf{V}_i(f_{tr})) \right]^{\frac{1}{R_t}}, \quad (6.1)$$

where \mathbf{t}_t is the track defined by the set of sinusoidal parameter pairs

$$\mathbf{t}_t = \{(A_{tr}, f_{tr})|r = 1, \dots, R_t\}, \quad (6.2)$$

$p(x)$ denotes a unidimensional Gaussian distribution and $\boldsymbol{\theta}_i = (\mathbf{M}_i, \mathbf{V}_i)$ is the parameter vector of the assigned musical instrument i , containing the mean and variance time–frequency surfaces. Such a cancellation was not necessary in timbre matching,

since the Maximum Likelihood decision of comparing a track group with different prototype envelopes was unaffected by the common factor arising from the total number of time–frequency points in the group.

Another point to note is that optimization according to time and amplitude scales, which was realized in the timbre matching stage by time-stretching the tracks and sliding the surfaces in amplitude, is no longer applicable here. An optimized match has already been found to assign a prototype envelope to each onset/offset interval, with given optimal time-stretching and amplitude-scaling parameters (α, N) . The track-wise frequency support evaluation operations $\mathbf{M}_i(f_{tr})$ and $\mathbf{V}_i(f_{tr})$ must be performed with those parameters.

Inter-channel comparison

The third and last criterion for extraneousness concerns a comparison between the BSS output channels. A track that is correctly considered extraneous in a given channel will certainly appear as a correct track on another. From another point of view, a given sinusoidal track in the original stereo mixture can originate several tracks in several post-BSS channels. If the track was non-overlapping in the original mixture, it will result in one correct track in the correct source, and, if leaked to adjacent channels after BSS, in one or more extraneous tracks in different channels, with lower energy but the same frequency support and amplitude shape. If the track was overlapping in the original mixture, it will be partially unmixed by the BSS stage, and will appear as two or more valid (non-extraneous) tracks in two or more BSS output channels.

These considerations, together with the fact that a set of partially separated channels are available in parallel after BSS, allow defining the following algorithm as the final check for track extraneousness:

1. For each track in the current post-BSS channel (which will be called the “considered track”), search for tracks on all other channels whose mean frequency is very close to the mean frequency of the considered track. As in the track grouping module (Sect. 5.2.3), frequency distance is measured in cents, and a threshold of 40 cents is set to declare frequency matches.
2. For each of the found tracks in step 1, select the one whose location in time is closest to that of the considered track. The time span of the found tracks is allowed to be equal or longer than the time span of the considered track, but not shorter (in other words, it is assumed that extraneous tracks will be equal or shorter than the original tracks they originated from).
3. Check if the mean amplitude of the found track is considerably larger than the mean amplitude of the considered track. If yes, label the considered track as extraneous.

The amplitude threshold needed in step 3 to quantify what “considerably larger” actually means must be chosen very carefully. Candidate tracks that differ only

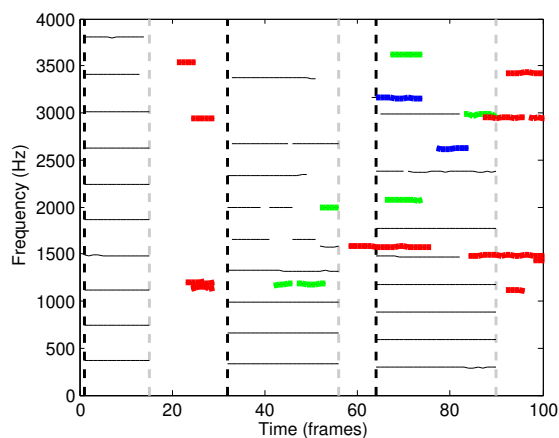


Figure 6.3: Application example of extraneous track detection. The colors denote the criteria used to declare extraneousness: red denotes the temporal criterion, blue the timbral criterion and green the inter-channel criterion.

by a moderate amplitude are probably both valid, and none should be declared as extraneous. On the other hand, tracks with a large amplitude difference could in fact be both valid, if they are overlapping and one of them belongs to a note played very softly in the original mixture. Assuming the mixture is reasonably well balanced, the latter is however more unlikely to happen.

It is important to note that none of the three criteria for extraneousness assumes harmonic relationships between partials. As repeatedly argued in the previous chapter, the non-assumption of harmonicity was chosen in order to allow the separation of same-instrument chords and of sounds with inharmonic partials.

A practical example of detection of extraneous tracks is shown in Fig. 6.3. The example corresponds to a three-note melody fragment played by a piano, that has been separated from a 3-voice mixture additionally containing an oboe and a trumpet. The figure shows the frequency support of the detected sinusoidal tracks after the BSS stage. The vertical dashed lines indicate the detected onsets and offsets. The colored frequency trajectories correspond to the detected extraneous tracks. Red tracks were detected because they do not fulfill the temporal criterion, blue tracks do not meet the timbral criterion and green tracks do not meet the inter-channel criterion. For this particular example, after the removal of the extraneous tracks separation quality improved from 9.16 dB to 14.94 dB in the case of SDR, from 26.77 dB to 38.41 dB in the case of SIR and from 9.25 dB to 14.96 dB in the case of SAR.

6.4 Evaluation of classification performance

The modules prior to timbre matching are common to both presented stereo separation approaches, and thus their classification and segmentation performance is

identical. To allow a direct comparison with the monaural case, and an assessment of the improvement introduced by the spatial cues, the “basic” databases used in Sect. 5.2.4 of the previous chapter for the classification experiments will be used here again. Those databases consisted of monaural mixtures containing individual notes in consonant intervals (EXP 1), individual notes in dissonant intervals (EXP 2), and sequences of notes (EXP 3), with a polyphony of 2 to 4 instruments. Each degree of polyphony in EXP 1 and EXP 2 is represented by 10 mixtures, and in EXP 3 by 20 mixtures. To generate the instantaneous, stereo counterpart for those experimental setups (which will be called, respectively, EXP 1s, EXP 2s and EXP3s), the sources were equally distributed across the stereo field, as was done in Sect. 3.4.2.

For both the classification and the separation experiments, the STFT was used as sparse transformation in the BSS stage. This was mainly decided on the grounds of computational simplicity. Chapter 3 showed that frequency warping can improve results, but, as argued in Sect. 3.1, at the cost of heavily increasing computational demands, due to the direct filter bank implementation that was used. The practical usage of frequency warping in an application scenario such as the one proposed in this chapter should preferably be based on more efficient implementations, such as the mentioned use of chains of all-pass filters [70].

For classification, the used evaluation measure is again the note-by-note accuracy. Also, the different timbre similarity measures introduced in Sect. 5.2.4 are compared. These were: averaged Euclidean distance (Eq. 5.5), Gaussian likelihood (Eq. 5.6) and Gaussian likelihood weighted proportionally to the length of the constituent tracks, and inversely-proportional to the mean track frequency (Eq. 5.7). The timbre library is the same as the one used in the two previous chapters: a set of 5 trained prototype envelopes of piano, clarinet, oboe, trumpet and non-vibrato violin. For more details about the training parameters, see Sect. 4.7.

The results are shown in Table 6.1 for EXP 1s and EXP 2s, and in Table 6.2 for EXP 3s. As could be expected, classification accuracy is significantly better than in the monaural case, attaining 87.67% with 2 channels (compared to 79.81% in mono), 86.43% with 3 channels (compared to 77.79%) and 82.38% for 4 channels (compared to 61.40%). For the sequences, the best average accuracy is 71.08%, compared to 60.04% in the monaural setup.

Apart from the improvement in accuracy, two notable aspects differ from the monaural case. The first is that the accuracy is much more independent from the degree of polyphony. In fact, there are even cases where the accuracy is better with higher polyphonies, such as with 3-channel polyphony compared to 2-channel polyphony in EXP 2s. This indicates that the separation quality delivered by the BSS stage is good enough to consider that the classification has been performed on isolated notes. In other words, the remaining interferences have too little a weight to considerably worsen the timbre matching procedure.

Secondly, the unweighted likelihood is on average clearly the better measure, in contrast to the monaural case, in which the weighted likelihood was preferred for dissonant intervals, and the unweighted for consonant intervals. An explanation for this is the more robust sinusoidal detection after BSS separation, which results in individual tracks being often correctly, steadily detected during the whole duration

Polyphony	Consonant (EXP 1s)				Dissonant (EXP 2s)			
	2	3	4	Av.	2	3	4	Av.
Euclidean distance	63.33	77.14	76.57	72.35	60.95	86.43	78.00	75.13
Likelihood	86.67	84.29	82.38	84.45	81.90	81.95	81.33	81.73
Weighted likelihood	70.00	70.95	66.38	69.11	78.10	78.62	74.67	77.13

Table 6.1: Instrument detection accuracy (%) for simple stereo mixtures of one note per instrument.

Polyphony	Sequences (EXP 3s)		
	2	3	Av.
Euclidean distance	64.71	59.31	62.01
Likelihood	67.71	74.44	71.08
Weighted likelihood	69.34	58.34	63.84

Table 6.2: Instrument detection accuracy (%) for stereo mixtures of sequences containing several notes.

of the note, and thus making the track-length weighting uninformative. This was not the case in the mono experiments, where interrupted tracks due to overlaps were far more common.

The reasons for the worse performance of the sequence experiments are again the artificial cut of same-channel, consecutive notes, and the higher probability of overlapping in time. When regarding track-wise instrument detection, however, the effect of long sequences is the opposite than in the monaural case. In the mono system, track creation and note-to-source clustering was entirely determined by the timbre matching results, and thus longer sequences meant a higher probability of error. In the stereo case, source clustering is robustly performed by the BSS stage, and the global assignments of separated sources to instruments can be made, as mentioned, by majority voting across the detected and individually classified notes. Thus, as long as the note-by-note classification accuracy is higher than random, longer sequences will lead to more robust majority decisions.

6.5 Evaluation of separation performance

The next two subsections are devoted to the separation quality evaluation of the two proposed systems for stereo separation, based respectively on track retrieval and sinusoidal subtraction. Like in the previous section, part of the evaluation experiments were performed on a subset of the experiments used in the previous chapter, so that the benefits of adding spatial information become apparent. Again, the stereo versions of the basic experiments (EXP 1s, EXP 2s and EXP 3s) were used, all containing note sequences without simultaneous onsets (except for same-instrument chords). The results obtained for those experiments, with both systems, will be reported in Sect. 6.5.1.

Additionally, a new collection of mixtures were generated to exploit the additional capabilities of the stereo systems, most importantly the ability to separate same-onset notes from different instruments. Again, a differentiation will be made

Type	Name	Source content	Harmony	Instr.	Polyph.
Basic (stereo)	EXP 1s	Individual notes	Consonant	Unknown	3,4
	EXP 2s	Individual notes	Dissonant	Unknown	3,4
	EXP 3s	Sequence of notes	Cons., Diss.	Unknown	3
Common onsets	EXP 8s	Individual notes	Consonant	Unknown	3,4
	EXP 9s	Individual notes	Dissonant	Unknown	3,4
	EXP 10s	Sequence with chords	Cons., Diss.	Unknown	3,4

Table 6.3: Table of experimental setups for the stereo separation systems.

between consonant individual-note mixtures (i.e., multi-instrument chords), denoted by EXP 8s, dissonant individual-note mixtures (EXP 9s) and sequences (EXP 10s), which in this case can contain same-instrument chords and clusters. Table 6.3 summarizes all experimental setups for the stereo case. Note that EXP 1s to EXP 3s keep their numbering from the mono versions, since they are a mere upmix of the same notes. Also, in contrast to the mono experiments (see Table 5.1), the instruments are always assumed unknown. The results corresponding to this new set of experiments will be presented in Sect. 6.5.2.

Each degree of polyphony is again represented by 10 mixtures, making a total of 110 stereo separation experiment instances. Only the mixture for which the correct number of sources was detected by the BSS stage were considered. Insight into the source detection performance of the angular-kernel-based mixing matrix estimation was given in Table 3.4.

The performance of the track-retrieval-based system will be measured by the *Spectral Signal to Error Ratio* (SSER), as defined in Eq. 5.9, since, like the monaural system, it suffers from onset uncertainty and phase removal. The sinusoidal-subtraction-based system, as has been discussed, allows the full usage of the common time-domain measures of SDR, SIR and SAR (see Sect. 3.5.2). However, the SSER was additionally computed in the latter case, so that the change in performance between both systems can be readily appreciated.

6.5.1 Stereo version of monaural experiments

The averaged SSER, SDR, SIR and SAR values for the stereo-upmixed version of the basic monaural experiments are shown in Table 6.4. Note that the case of 2-instrument polyphony has been ignored. In that case, the instantaneous stereo separation problem would be even-determined and thus trivial, and the BSS stage alone will already yield near-perfect separation. For example, SSER, SDR and SAR values for the $M = 2$ case were typically higher than 80 dB, and SIR values higher than 130 dB.

Stereo track retrieval outperforms monaural track retrieval by around 5 to 7 dB (compare with Table 5.4). In turn, sinusoidal subtraction significantly outperforms stereo track retrieval in terms of SSER, the difference ranging between around 5 dB and 10 dB. Other expected behaviors hold here as well (dissonances are easier to separate than consonances, higher polyphonies are more difficult), however in a less pronounced manner than in the monaural case. In the presence of large interferences,

Source type	Polyph.	Track retrieval	Sinusoidal subtraction			
		SSER	SSER	SDR	SIR	SAR
Individual notes, cons. (EXP 1s)	3	13.92	21.13	20.70	43.77	20.77
	4	12.10	17.13	16.78	40.83	16.83
Individual notes, diss. (EXP 2s)	3	14.37	24.20	23.63	47.01	23.72
	4	12.06	21.33	20.76	43.74	20.81
Sequences of notes (EXP 3s)	3	12.52	22.00	21.48	44.79	21.53

Table 6.4: Results for the stereo version of the basic experiments of Chapter 5 using track retrieval and sinusoidal subtraction.

Source type	Polyph.	Track retrieval	Sinusoidal subtraction			
		SSER	SSER	SDR	SIR	SAR
Individual notes, cons. (EXP 8s)	3	13.36	18.26	17.35	40.48	17.39
	4	14.88	15.31	14.96	36.25	15.06
Individual notes, diss. (EXP 9s)	3	11.88	21.72	20.91	44.56	21.03
	4	15.10	18.93	18.24	40.36	18.30
Sequences with chords (EXP 10s)	3	11.21	17.95	17.17	32.30	17.44
	4	10.57	12.16	11.18	26.26	11.51

Table 6.5: Results for the simultaneous-note experiments using track retrieval and sinusoidal subtraction.

the sources separated by means of spectral subtraction typically improve averaged performance measures by 2 to 4 dB in the case of SDR and SAR and by 3 to 6 dB in the case of SIR when compared to the output channels of the BSS stage.

6.5.2 Experiments with simultaneous notes

The performances obtained for the new experiments involving common-onset notes (Table 6.5) were lower on average, but not significantly so (the average difference is of around 1 to 2 dB). This again confirms the effectiveness of the BSS stage in providing a good partially-separated basis for the remaining refinements. The degree of improvement of sinusoidal subtraction over track retrieval is less homogeneous than with the basic experiments, this time ranging between approximately 1 dB SSER for the 4-channel EXP 8s and 10 dB SSER for the 3-voice EXP 9s.

6.6 Conclusions

The goal of this chapter was to combine several of the ideas developed throughout the present work into a hybrid framework for supervised separation of stereo instantaneous mixtures. The sparsity-based blind separation method tested in Chapter 3 was used to exploit spatial information and provide a set of preliminarily separated channels. Sinusoidal modeling techniques (Chapter 4) were used to channel-wise refine that separation, according to common onset and good continuation cues, assisted by timbral similarity measures computed by matching with the stored timbre models.

More specifically, two approaches were presented. The first is a direct extension of the track-retrieval system proposed in Chapter 5 to the stereo case: it basically applies the monaural separation system to each output channel of the BSS stage. Classification and separation performance is considerably higher than in the monaural case, but such a method ignores the noise part of the signal altogether.

The second approach consisted in the removal of remaining interferences by detecting extraneous tracks that have been leaked from the original mixture to one or more of the wrong post-BSS channels. Such a detection is again guided by common-onset and timbral criteria, as well as by a mutual comparison between the channels. The detected extraneous tracks are then subtracted from the partially separated channels. This improves performance compared to using the BSS stage alone, and compared to the track retrieval approach.

7

Conclusions and outlook

The developments reported in this work were motivated by the need for improving content-based analysis and processing of complex musical signals. The role of source separation in such a context is to allow a *separation-for-understanding* or *Significance Oriented* (SO) paradigm in which feature extraction of the (at least partially) separated components is easier and more robust than feature extraction on the mixture as a whole. The specific scenario addressed was the separation of monaural and stereo instantaneous musical mixtures. To that aim, several new methods were proposed, all relying on source models of different levels of complexity and applicability constraints: starting from unsupervised, blind separation based on sparsity-optimized representations, and subsequently developing towards a more sophisticated supervised method making use of models capturing timbre and its temporal evolution.

A characteristic of the present work are the numerous connections with MCA and MIR techniques, such as timbre learning, instrument classification, dimensionality reduction, onset detection, etc. On the one hand, they have been used both as a means of helping (Chapter 6), or allowing (Chapter 5) separation. On the other, as by-products of the design of the separation systems, several modules have been evaluated in non-separation tasks. In particular, the proposed timbre modeling methodology has been evaluated as a statistical model for the classification of isolated musical sounds and for polyphonic instrument detection.

7.1 Summary of results and contributions

What follows is a list of results and contributions in the order they were developed and reported in the present dissertation. For a more detailed summary, the reader is referred to the conclusion sections at the end of each chapter.

Evaluation of sparsity and disjointness of music and speech mixtures

The starting point for the design of the source models was a preliminary study on sparsity (Sect. 3.2) and disjointness (Sect. 3.3) properties of speech and music signals. Sparsity was measured by normalized kurtosis and disjointness (i.e., the degree of non-overlapping) was measured by approximate *W-Disjoint Orthogonality* (WDO). One of the main results was that speech needs a balanced trade-off between

time and frequency resolutions for optimal sparsity and disjointness, whereas for music signals frequency resolution must be favored.

Comparison of sparsity, disjointness and source separation performance of frequency-warped representations

Next, the focus was shifted to music signals, and several non-uniform time–frequency representations were compared not only in terms of intrinsic sparsity and disjointness properties, but also in terms of source detection and separation performance. To that end, a set of frequency-warped filter banks (a constant-Q and three auditory frequency filter banks: Mel, Equal Rectangular Bandwidth and Bark) were implemented as the representation front-end of a sparsity-based stereo source separation system, and compared to the use of the common STFT-based spectrogram. Evaluation experiments showed that all warping schemes improved sparsity and disjointness. In the case of disjointness, the improvement is higher the more the sources overlap. Source detection and separation performance was improved as well, most significantly in terms of *Signal to Distortion Ratio* (SDR) and of *Signal to Artifacts Ratio* (SAR). The improvement in *Source to Interference Ratio* (SIR) is however lower on average, which points to the fact that more sophisticated assumptions need to be taken to handle the overlapping parts of the spectrum, a topic that was developed in the remainder of the thesis. For a detailed summary of these quantitative results, the reader is referred to Sect. 3.6.

Detailed and compact timbre modeling

The previous observations led to the exploration of means to provide a priori information to guide and help separation in the spectral domain. The novel timbre modeling approach proposed in Chapter 4 is based on a compact representation of the spectral envelope. A notable feature is its detailed characterization of temporal timbre evolution. The models are built by first extracting the spectral envelope of a training set of isolated note samples by means of sinusoidal modeling and spectral interpolation. The partial parameters are then arranged into a matrix upon which PCA is performed, to attain compactness and extraction of salient spectral shapes. The proposed method to arrange the partials into the PCA data matrix aims at preserving formant structures and is based on reinterpolating the spectral envelope at an equally-spaced frequency grid. Spectral interpolation followed by PCA was called in this context the *representation stage*. Such stage was thoroughly evaluated with respect to compactness, accuracy and generality, obtaining that the envelope interpolation method outperforms, in all three criteria, the more common method of arranging the data matrix with the indices of the partials.

This is followed by a *prototyping stage* in which the coefficients projected to PCA space, forming timbral trajectories for each training sound sample, are used to train *prototype curves* that can be interpreted as multidimensional *Gaussian Processes* (GP). This results in the construction of a timbre space where it is possible to visually inspect timbre similarities, or perform classification by analyzing geometric

relationships. Also, it is possible to transform back to a less compact version of the models in the time–frequency domain, which originates the *prototype envelopes*, consisting of a mean and a variance surface, alternatively interpreted as a time- and frequency-variant GP.

Use of the timbre models for monophonic and polyphonic instrument classification

The developed timbre modeling approach was additionally used for non-separation applications. First, a musical instrument classification experiment was conducted (Sect. 4.7), consisting in projecting the spectral envelope of unknown samples on the PCA space and comparing an average distance between the resulting trajectory and each one of the prototype curves. This approach reached a classification accuracy of 94.86% with a database of 5 classes, and outperformed using MFCCs for the representation stage by 34%.

A second content-analysis application for the models arose in the context of detection of instruments in polyphonic monaural mixtures (Sect. 4.8). An existing sound source formation system based on the *Normalized Cut* (Ncut) criterion was extended with a timbre matching module in which the separated partial clusters were compared to a set of prototype envelopes derived from the above models. This is a clear example of the *separation-for-understanding* paradigm. Obtained accuracies range from 65% for 2-voice mixtures to 33% for 4-voice mixtures for a database of 6 instruments.

Finally, the separation systems developed in the last chapters are (partially) based on timbre matching with the models (they follow the *understanding-for-separation* paradigm), and can thus also be used for polyphonic instrument detection. In the single-channel version (Sect. 5.2.4), accuracies up to 79.81% for 2 voices, 77.79% for 3 voices and 61.40% for 4 voices were obtained with a 5-class instrument database. The performance in the stereo case (Sect. 6.4) was obviously higher, ranging from 86.67% with 2 voices to 82.38% with 4 voices.

Novel approach for separation of monaural mixtures, based on matching with the timbre models and without harmonicity constraints

The a priori information conveyed by the designed source models was combined with sinusoidal modeling to implement a source separation system (Sect. 5.2). First, the mixtures were constrained to a single channel. This scenario is far more demanding than the stereo case, where it is possible to exploit spatial cues, but was devised as a way to evaluate the ability to detect and separate constituent notes following only spectral cues. Broadly, separation is based on applying grouping principles, as derived from *Computational Auditory Scene Analysis* (CASA), to detected partial tracks on the mixture. More specifically, partial tracks are clustered into notes according to common onset and good continuation principles. To resolve overlapping partials, or overlapping sections of partials, timbre matching is performed with the timbre model database, and the missing parts are retrieved from the model with the

highest likelihood by means of linear interpolation at the desired frequency support. This approach has been called *track retrieval*. The system has been tested with up to 4-voice polyphony.

An important feature of the proposed separation system is that no assumptions on the harmonicity (or quasi-harmonicity) of the sources were taken. One of the consequences of this is that, in contrast to many previous approaches, the system does not require pitch-related a priori information, nor a preliminary multipitch estimation stage. Instead, grouping and separation solely relies on the dynamic behavior of partial amplitudes. This also allows separating highly inharmonic sounds (such as bells) and detecting single-instrument chords as single entities. A limitation of such a method is the inability to separate notes from different instruments that start during the same analysis frame. A large set of experiments were conducted to demonstrate the separation performance with different kinds of input mixtures: single arpeggios, note sequences, consonant and dissonant intervals, tonal chords and chord clusters and inharmonic sounds.

Novel hybrid approaches for separation of stereo mixtures

A sparsity-based approach for exploiting spatial cues, and the previously mentioned sinusoidal and timbral methods for exploiting spectral cues, were finally combined for the proposal of two alternative methodologies for the separation of stereo mixtures. The first approach (Sect. 6.2), again based on track retrieval, was a simple extension of the monaural system with a preliminary BSS stage for delivering partially separated tracks. The BSS stage, which was also used in Chapter 3, follows a staged architecture and consists of kernel-based angular clustering for the estimation of the mixing matrix and on ℓ_1 norm minimization for resynthesizing the sources. It basically applies the monaural processing chain detailed above to each partially separated channel. This eliminates interferences, but still has the main disadvantage that the noise part of the signals is not preserved.

Thus, a proposed alternative (Sect. 6.3) was to use the spectral and timbral cues not to resynthesize partials, but to detect and eliminate (by sinusoidal subtraction) extraneous tracks. The detection of extraneous tracks was based on temporal, timbral and inter-channel-comparison criteria. In this way, separation quality was significantly improved compared to the track retrieval method and to the BSS stage.

7.2 Outlook

The reported developments have covered a wide range of topics that can be further studied either for improving source separation, or for their application within different content analysis contexts. This is a partial list of possible relevant research directions.

Separation-for-understanding applications

Since source separation has been regarded in this work from the point of view of improving content analysis via the separation-for-understanding paradigm, an obvious direction for future research is the evaluation of the proposed separation methods within well-developed MCA systems, and a comparison of performance with whole-mixture feature extraction. This can be of especial interest in the case of polyphonic pitch detection or transcription, but could be also usable in other contexts such as music genre, mood, structure or harmonic analysis.

Refinement of the timbre models

The timbre modeling approach can be further refined in various ways, corresponding to the individual processing steps involved in the process. In the representation step, other spectral basis decomposition methods can be evaluated instead of PCA. For example, *Linear Discriminant Analysis* (LDA), which maximizes class separability and therefore is expected to improve applications aimed at classification or instrument detection. In the classification context, it can be of interest to test the performance difference of performing timbre matching in a single timbre space (as has been done here), with that of performing Maximum Likelihood decisions between a set of multiple spaces, one for each trained instrument. Also, frequency-warped interpolation methods could be used to substitute the regular frequency grid, following the ideas developed in Chapter 3.

The prototyping stage opens some other interesting research directions. The used Gaussian Process approach can be refined to obtain more informative, parametrized curve shapes by using methods such as nonlinear regression, Principal Curves [22] (which generalize PCA to the nonlinear case), or neural approaches to the modeling of nonlinear systems [130]. Another possibility is to separate prototype curves into components corresponding to the temporal segments of the ADSR envelope. This can allow three enhancements: first, different statistical models can be more appropriate to describe different segments of the temporal envelope. For example, clusters can be appropriate for highly stationary parts of the sustain section, and they could be combined with parametrized “tails” arising from them and describing the attack and release parts. Second, such a multi-model description can make possible a more abstract parametrization at a morphological level, turning timbre description into the description of geometrical relationships between objects. For example, a violin sound could be described as a cluster or set of clusters with different covariance matrices, connected with curved tails that output the clusters at a given angle or with a given gradient in timbre space. Note that this still offers a higher level of dynamic detail than temporal modeling based on HMMs. And finally, it would allow treating the segments differently when performing time interpolation for the curve averaging, and time stretching for maximum-likelihood timbre matching. This avoids stretching the attack time in the same degree than the sustained part.

Finally, the models could also be extended by studying in detail the influence of the fundamental frequency and of dynamics on the spectral envelope, and by including those two factors as either parameters for retrieval or as additional model

dimensions.

Other applications of the timbre models

Timbre modeling is undoubtedly the module most susceptible of being used for a variety of analysis applications other than source separation. For example, its application for instrument classification and polyphonic instrument detection can be pursued and further optimized by considering alternative similarity measures. When using a single projection space, it can be used as an automatically generated timbre space for timbral and psychoacoustical characterization. Also, the models could be used to generate appropriate spectral envelope shapes to enhance timbre realism in pitch shifting applications.

It is also possible to envision sound-transformation or even synthesis applications involving the generation of dynamic spectral envelope shapes by navigating through the timbre space, either by a given set of deterministic functions or by user interaction via mouse or gestural sensors. If combined with multi-model extensions of the prototyping stage, like the ones mentioned above, this could allow approaches to morphological or object-based sound synthesis.

Improvement of timbre matching

Robustness of the separation systems, and of their classification and segmentation modules, can be improved by studying alternative similarity measures for timbre matching, or more efficient parameter search algorithms for the amplitude scaling and time stretching of partial track groups. An interesting candidate technique to this aim is *Dynamic Time Warping* (DTW). Also, such an increase in robustness could allow single-track classification and detection of same-onset, different-instruments overlapping partials in the monaural case.

Refinement of the separation of the noise residual

Separation of the noise part was not considered in the monaural case, and assumed to be partially realized by the BSS stage in the stereo setup. Extending the systems with an explicit modeling of the noise parts can further improve the quality of the separated sounds. A possible starting point for this research direction is the assumption of correlation between the noise and sinusoidal spectral envelopes, as proposed in [61].

Addition of a feedback loop to the hybrid separation framework

The hybrid separation approaches proposed in Chapter 6 are implemented using a sequential architecture: first, sparsity and spatial diversity are exploited (BSS stage), then the partially separated channels are refined via sinusoidal modeling and matching with the timbre models. A potentially more efficient design could be obtained by allowing the refinement stage pass model-based information back to the BSS stage, and defining an iterative procedure optimizing a given objective

function. For example, passing classification and onset detection results, or assigning probabilities to the extraneous tracks could, starting from the second iteration, help refine the detection of the mixing matrix directions and the avoidance of the artifacts due to spectral zeros.

Separation of more complex signals

Finally, an evident research goal would be to extend the applicability of the proposed separation systems to perform with more realistic signals of higher polyphonies, different mixing model assumptions (e.g., delayed or convolutive models due to reverberation) and real recordings that can contain different levels of between-note articulations (transients), playing modes, special effects, moving sources, artificially-altered timbres, etc. Also, percussive sounds will need a specialized treatment. From the point of view of source modeling, such more demanding separation tasks will require more refined timbral descriptions, and possibly the learning of other mixture aspects, such as typical recording practices or even genre-based timbral and structural characteristics. The latter is a further example of the close and mutually beneficial connections that can exist between sound source separation and Music Content Analysis.



Related publications

Several methods and results presented in this dissertation have been published in the following works, which are listed here chronologically:

- J. J. Burred and T. Sikora. On the use of auditory representations for sparsity-based sound source separation. In *Proc. International Conference on Information, Communications and Signal Processing (ICICS)*, Bangkok, Thailand, December 2005.

This article presented the preliminary results obtained in measuring disjointness of speech and music mixtures in a separation-algorithm-independent framework for the STFT and two auditory scales: Bark and ERB. In the present dissertation, they were reported and extended in Sect. 3.3.

- J. J. Burred and T. Sikora. Comparison of frequency-warped representations for source separation of stereo mixtures. In *Proc. 121st Convention of the Audio Engineering Society*, San Francisco, USA, October 2006.

This work follows the developments presented in the previous one by comparing the actual separation quality of the STFT with that obtained with a Constant-Q and ERB, Bark and Mel auditory warpings, as was presented here in Sects. 3.4 and 3.5.

- J. J. Burred, A. Röbel and X. Rodet. An accurate timbre model for musical instruments and its application to classification. In *Proc. Workshop on Learning the Semantics of Audio Signals (LSAS)*, Athens, Greece, December 2006.

The timbre modeling process discussed in Sects. 4.4 to 4.6 was first introduced in this article through an abridged presentation of its design and development.

- L. G. Martins, J. J. Burred, G. Tzanetakis and M. Lagrange. Polyphonic instrument recognition using spectral clustering. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

The timbre models were integrated into a source formation framework based on the Normalized Cut criterion developed by the above coauthors. The corresponding results were reported on Sect. 4.8.

- J. J. Burred and T. Sikora. Monaural source separation from musical mixtures based on time–frequency timbre models. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

This paper contains an abridged presentation of the monaural separation system described in Sect. 5.2.

- J. J. Burred, M. Haller, S. Jin, A. Samour and T. Sikora. Audio Content Analysis. In Y. Kompatsiaris and P. Hobson (Eds.), *Semantic Multimedia and Ontologies: Theory and Applications*, Springer, January 2008.

This is an introductory chapter to the general topic of Audio Content Analysis. A timbre space consisting of a set of prototype curves, similar to the one depicted on Fig. 4.18, was presented as an example of a feature extraction process for the purpose of music description.

- J. J. Burred, A. Röbel and T. Sikora. Polyphonic musical instrument recognition based on a dynamic model of the spectral Envelope. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Taipei, Taiwan, April 2009.

This paper reports on the experiments in polyphonic musical instrument recognition obtained in the timbre matching stage, as detailed in Sect. 5.2.4.

List of Figures

1.1	Chart of thematic dependencies.	8
2.1	Ideal stereo reproduction setup with azimuth angle.	17
2.2	Instantaneous stereo recording techniques.	18
2.3	Delayed stereo recording techniques.	18
2.4	Convolutional stereo recording techniques.	18
2.5	Comparison of Laplace and normal probability density functions for two different variances.	28
2.6	Example of sparsity properties of basic signal representations.	31
2.7	Example of PCA and whitening of a bivariate normal distribution.	36
2.8	Diagram of a general staged Blind Source Separation system.	39
2.9	Scatter plot in two-channel mixture space for statistically independent and sparse sources.	40
2.10	Application of PCA to an even-determined mixture of two statistically independent and sparse sources.	41
3.1	Normalized frequency responses of 17-band, Hann-window filter banks at a sampling rate of 8 kHz.	56
3.2	Filter bank center frequencies as a function of band number, for 16 kHz sampling rate and 257 bands.	60
3.3	Comparison of 129-band spectrogram and ERB spectral representation of a clarinet melody.	61
3.4	Averaged sparsity, measured by normalized kurtosis, against number of bands for speech and music sources at 8 kHz sampling rate.	64
3.5	Average sparsity, measured by normalized kurtosis, against number of bands for spectrogram (STFT), Bark and ERB representations, at 8 kHz sampling rate.	65
3.6	Disjointness (\overline{WDO}) against number of bands for speech (SP), uncorrelated music (UMEL) and correlated music (CMEL), 3-source stereo mixtures at 8 kHz sampling rate.	69
3.7	Disjointness (\overline{WDO}) against number of bands for ERB, Bark and STFT representations, 3-source stereo mixtures and 8 kHz sampling rate.	70
3.8	Example of 3-source, stereo mixture scatter plots for music signals.	73
3.9	Example of mixing matrix estimation by kernel-based angular clustering.	74
3.10	Shortest path resynthesis.	77
3.11	Evaluation of the source resynthesis stage: SDR, SIR and SAR as a function of number of subbands L , for stereo mixtures of $N = 3$ sources.	79
3.12	Evaluation of the source resynthesis stage: SDR, SIR and SAR as a function of number of subbands L , for stereo mixtures of $N = 4$ sources.	80

4.1	Comparison between linear and cubic interpolation for spectral envelope estimation.	88
4.2	Frequency-dependent thresholding for peak picking.	91
4.3	Overview of the timbre modeling process.	96
4.4	Example of basis decomposition of a spectral envelope by PCA. The data matrix is the product of the coefficient and basis matrices. The figure shows the first 3 bases of a violin note played with vibrato. . .	99
4.5	Interpretation of the basis decomposition of Fig. 4.4 as a projection onto the space spanned by the bases.	99
4.6	Example of envelope reconstruction with only the first PCA basis. .	100
4.7	Explained variance as a function of dimensionality for a single French horn note.	100
4.8	PCA data matrix with Partial Indexing (1 octave of an alto saxophone).	102
4.9	PCA data matrix with Envelope Interpolation (1 octave of an alto saxophone).	102
4.10	Cubic envelope interpolation at a regular frequency grid.	103
4.11	Cross-validation framework for the evaluation of the representation stage.	104
4.12	Results from experiment 1: explained variance.	105
4.13	Reinterpolation error.	106
4.14	Results from experiment 2: Relative Spectral Error.	107
4.15	First two dimensions in model space of the training data for one octave of an acoustic guitar, and corresponding Gaussian model.	108
4.16	Results from experiment 3: training/test cluster distance.	110
4.17	Training/test cluster distance measured by KL divergence.	111
4.18	Prototype curves in the first 3 dimensions of model space corresponding to a 5-class training database of 423 sound samples, preprocessed using linear envelope interpolation. The starting points are denoted by squares.	112
4.19	Orthogonal projections of the timbre space of Fig. 4.18.	113
4.20	Prototype envelopes corresponding to the curves on Fig. 4.18.	115
4.21	Envelope evaluation points and traces for Figs. 4.22 and 4.23.	117
4.22	Envelope mean and variances at points A,B and C on Fig. 4.21. . .	117
4.23	Evolution of the spectral envelope alongside the traces on Fig. 4.21. .	118
4.24	Classification results: averaged classification accuracy.	120
5.1	Monaural source separation system overview.	133
5.2	Sinusoidal modeling and onset detection for an 8-note sequence of alternating piano and oboe notes.	135
5.3	Examples of matches between track groups (solid black lines) and prototype envelopes.	140
5.4	Examples of likelihood optimization results for a piano note.	141
5.5	Illustration of track types and track extension/substitution for two notes separated by a perfect fifth.	143

5.6	Application example of track extension/substitution for a mixture of 2 notes.	145
5.7	Example of separation of an individual-note, four voice mixture (part of EXP 1).	148
5.8	Example of separation of a mixture of 3 sequences of notes (part of EXP 3k).	148
5.9	Ex. of separation of a mixture of two consonant chords (part of EXP 4).	150
5.10	Example of separation of a mixture of two clusters (part of EXP 5).	150
5.11	Example of separation of a mixture of two chord sequences (part of EXP 6).	151
5.12	Example of separation of a mixture of sequences including chords (part of EXP 6).	151
6.1	Overview of stereo separation based on track retrieval.	157
6.2	Overview of stereo separation based on sinusoidal subtraction.	159
6.3	Application example of extraneous track detection. The colors denote the criteria used to declare extraneousness: red denotes the temporal criterion, blue the timbral criterion and green the inter-channel criterion.	163

List of Tables

2.1	Classification of audio source separation tasks according to the nature of the mixtures.	10
2.2	Classification of audio source separation tasks according to available a priori information.	10
2.3	Sparsity measures corresponding to the signals in Fig. 2.6.	32
3.1	Summary of nominal center frequency (f_k) and bandwidth (Δf_k) definitions.	60
3.2	Maximum averaged sparsity, measured by normalized kurtosis, and optimal number of bands, for speech and music data for 8 kHz sampling rate.	64
3.3	Maximum disjointness, measured in % of $\overline{\text{WDO}}$, and optimal number of bands for speech (SP), uncorrelated music (UMEL) and correlated music (CMEL) data for 8 kHz sampling rate.	70
3.4	Evaluation of the mixing matrix estimation stage: averaged source detection rate (DR) and angular error (e_{ang}) in degrees, for stereo mixtures of $N = 3$ (left) and $N = 4$ sources (right).	75
3.5	Evaluation of the source resynthesis stage: maximum achieved SDR, SIR and SAR for stereo mixtures of $N = 3$ (left) and $N = 4$ sources (right).	78
4.1	Classification results: maximum averaged classification accuracy and standard deviation (STD) using 10-fold cross-validation.	120
4.2	Confusion matrix (detection accuracies in %) for single-note instrument classification. The labels denote: piano (p), oboe (o), clarinet (c), trumpet (t), violin (v) and alto sax (s).	123
4.3	Recall (RCL), precision (PRC) and F-Measure (F1) values for instrument identification in multiple-note mixtures.	124
4.4	Instrument classification performance (detection accuracy in %) for 2-, 3- and 4-note mixtures.	125
5.1	Table of experimental setups for the monaural separation system.	134
5.2	Instrument detection accuracy (%) for simple mixtures of one note per instrument.	142
5.3	Instrument detection accuracy (%) for mixtures of sequences containing several notes.	142
5.4	Results (averaged SSER) for the basic experiments.	149
5.5	Results (averaged SSER) for the extended experiments.	150
6.1	Instrument detection accuracy (%) for simple stereo mixtures of one note per instrument.	165
6.2	Instrument detection accuracy (%) for stereo mixtures of sequences containing several notes.	165
6.3	Table of experimental setups for the stereo separation systems.	166

6.4	Results for the stereo version of the basic experiments of Chapter 5 using track retrieval and sinusoidal subtraction.	167
6.5	Results for the simultaneous-note experiments using track retrieval and sinusoidal subtraction.	167

Bibliography

- [1] S. A. Abdallah and M. D. Plumbley. Polyphonic transcription by Nonnegative Sparse Coding of power spectra. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, 2004.
- [2] J. B. Allen. Short term spectral analysis, synthesis, and modification by Discrete Fourier Transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-25 (3):235–238, 1977.
- [3] S.-I. Amari. Natural gradient learning for over- and under-complete bases in ICA. *Neural Computation*, 11 (8):1875–1883, 1999.
- [4] X. Amatriain, J. Bonada, A. Loscos, and X. Serra. Spectral processing. In U. Zölzer, editor, *DAFX - Digital Audio Effects*, pages 373–438. John Wiley & Sons, 2002.
- [5] S. Araki, S. Makino, A. Blin, R. Mukai, and H. Sawada. Underdetermined blind separation for speech in real environments with sparseness and ICA. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Montreal, Canada, 2004.
- [6] D. Arfib, F. Keiler, and U. Zölzer. Source-filter processing. In U. Zölzer, editor, *DAFX - Digital Audio Effects*, pages 299–372. John Wiley & Sons, 2002.
- [7] J.-J. Aucouturier and F. Pachet. Improving timbre similarity: How high is the sky? *Journal of Negative Results in Speech and Audio Sciences*, 1 (1), 2004.
- [8] C. Avendano. Frequency domain techniques for stereo to multichannel upmix. In *AES International Conference on Virtual, Synthetic and Entertainment Audio*, Espoo, Finland, June 2002.
- [9] A. D. Back and A. S. Weigend. A first application of Independent Component Analysis to extracting structure from stock returns. *International Journal on Neural Systems*, 8, No. 4:473–484, 1997.
- [10] J. Backus. *The Acoustical Foundations of Music*. W.W. Norton, 1977.
- [11] M. Baeck and U. Zölzer. Real-time implementation of a source separation algorithm. In *Proc. International Conference on Digital Audio Effects (DAFX)*, London, UK, 2003.
- [12] R. Balan and J. Rosca. Statistical properties of STFT ratios for two channel systems and applications to blind source separation. In *Proc. International Workshop on Independent Component Analysis and Blind Source Separation (ICA)*, Helsinki, Finland, June 2000.
- [13] D. Barry, B. Lawlor, and E. Coyle. Sound source separation: azimuth discrimination and resynthesis. In *Proc. International Conference on Digital Audio Effects (DAFX)*, Naples, Italy, 2004.

-
- [14] M. Bay and J. W. Beauchamp. Harmonic structure separation using pre-stored spectra. In *Proc. International Conference on Independent Component Analysis and Blind Source Separation (ICA)*, Charleston, USA, 2006.
- [15] A.J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.
- [16] A.J. Bell and T.J. Sejnowski. Learning the higher order structure of a natural sound. *Network: Computation in Neural Systems*, 7, 1996.
- [17] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, 13 (5), part 2:1035–1047, September 2005.
- [18] L. Benaroya and F. Bimbot. Wiener based source separation with HMM/GMM using a single sensor. In *Proc. International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, 2003.
- [19] L. Benaroya, L. McDonagh, F. Bimbot, and R. Gribonval. Non negative sparse representation for Wiener based source separation with a single sensor. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, China, 2003.
- [20] A.J. Berkhout, D. de Vries, and P. Vogel. Acoustic control by Wave Field Synthesis. *Journal of the Acoustical Society of America*, 93(5):2764–2778, 1993.
- [21] A. Blin, A. Araki, and S. Makino. Blind source separation when speech signals outnumber sensors using a sparseness-mixing matrix estimation (SMME). In *Proc. International Workshop on Acoustic Echo and Noise Control (IWAENC)*, Kyoto, Japan, September 2003.
- [22] A. F. Bobick and A. D. Wilson. A state-based approach to the representation and recognition of gesture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19, No. 12:1325–1337, 1997.
- [23] P. Bofill and M. Zibulevsky. Blind separation of more sources than mixtures using sparsity of their short-time Fourier transform. In *Proc. International Workshop on Independent Component Analysis and Blind Signal Separation (ICA)*, Helsinki, Finland, 2000.
- [24] P. Bofill and M. Zibulevsky. Underdetermined blind source separation using sparse representations. *Signal Processing*, 81, 2001.
- [25] A. Bregman. *Auditory Scene Analysis*. MIT Press, 1990.
- [26] G.J. Brown. *Computational Auditory Scene Analysis: A Representational Approach*. PhD thesis, Univeristy of Sheffield, UK, 1992.

- [27] J.C. Brown. Calculation of a constant Q spectral transform. *Journal of the Acoustical Society of America*, 89 (1), January 1991.
- [28] J. J. Burred, M. Haller, S. Jin, A. Samour, and T. Sikora. Audio Content Analysis. In Y. Kompatsiaris and P. Hobson, editors, *Semantic Multimedia and Ontologies: Theory and Applications*. Springer, January 2008.
- [29] J. J. Burred, A. Röbel, and X. Rodet. An accurate timbre model for musical instruments and its application to classification. In *Proc. Workshop on Learning the Semantics of Audio Signals (LSAS)*, Athens, Greece, December 2006.
- [30] J. J. Burred and T. Sikora. On the use of auditory representations for sparsity-based sound source separation. In *Proc. International Conference on Information, Communications and Signal Processing (ICICS)*, Bangkok, Thailand, December 2005.
- [31] J. J. Burred and T. Sikora. Comparison of frequency-warped representations for source separation of stereo mixtures. In *Proc. 121st Convention of the Audio Engineering Society*, San Francisco, USA, October 2006.
- [32] J. J. Burred and T. Sikora. Monaural source separation from musical mixtures based on time-frequency timbre models. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.
- [33] J.J. Burred, A. Röbel, and T. Sikora. Polyphonic musical instrument recognition based on a dynamic model of the spectral envelope. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Taipei, Taiwan, April 2009.
- [34] O. Cappé and E. Moulines. Regularization techniques for discrete cepstrum estimation. *IEEE Signal Processing Letters*, 3 (4):100–102, 1996.
- [35] J.-F. Cardoso. Source separation using higher order moments. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Glasgow, UK, 1989.
- [36] J.-F. Cardoso. High-order contrast for Independent Component Analysis. *Neural Computation*, 11 (1):157–192, 1999.
- [37] J.-F. Cardoso, J. Delabrouille, and G. Patanchon. Independent Component Analysis of the cosmic microwave background. In *Proc. International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, 2003.
- [38] M. Casey. Sound classification and similarity tools. In P. Salembier B.S. Manjunath and T. Sikora, editors, *Introduction to MPEG-7*. John Wiley, 2002.

- [39] M. Casey and A. Westner. Separation of mixed audio sources by Independent Subspace Analysis. In *Proc. International Computer Music Conference (ICMC)*, Berlin, Germany, 2000.
- [40] M. A. Casey. *Auditory Group Theory with Applications to Statistical Basis Methods for Structured Audio*. PhD thesis, Massachusetts Institute of Technology, 1998.
- [41] Celemony. Direct Note Access. <http://www.celemony.com/cms/>, Retrieved on Monday, 1st September 2008.
- [42] S. Chen and D. Donoho. Basis Pursuit. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, USA, 1994.
- [43] E. C. Cherry. Some experiments on the recognition of speech, with one and two ears. *Journal of the Acoustical Society of America*, 25:975–979, 1953.
- [44] A. de Cheveigné and H. Kawahara. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111 (4):1917–1930, April 2002.
- [45] A. Cichocki, S. L. Shishkin, T. Musha, Z. Leonowicz, T. Asada, and T. Kurachi. EEG filtering based on blind source separation (BSS) for early detection of Alzheimer’s disease. *Clinical Neurophysiology*, 116, No. 3:729–737, 2005.
- [46] P. Comon. Independent Component Analysis - a new concept? *Signal Processing*, 36:287–314, 1994.
- [47] P. Comon. Blind identification and source separation in 2 x 3 under-determined mixtures. *IEEE Transactions on Signal Processing*, 52, No. 1:11–22, January 2004.
- [48] R. Cooney, N. Cahill, and R. Lawlor. An enhanced implementation of the ADReSS music source separation algorithm. In *Proc. 121st Convention of the Audio Engineering Society*, San Francisco, USA, 2006.
- [49] I. Daubechies. Time-frequency localization operators: A geometric phase space approach. *IEEE Transactions on Information Theory*, 34:605–612, 1988.
- [50] S. B. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28 (4):357–366, 1980.
- [51] J. R. Deller, J. H. L. Hansen, and J. G. Proakis. *Discrete-Time Processing of Speech Signals*. Wiley-IEEE Press, 1999.
- [52] M. Dickreiter. *Handbuch der Tonstudioteknik*. K.G. Saur, 6th edition, 1997.
- [53] R. O. Duda, R. F. Lyon, and M. Slaney. Correlograms and the separation of sounds. In *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, USA, 1990.

- [54] J.-L. Durrieu, G. Richard, and B. David. Singer melody extraction in polyphonic signals using source separation methods. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, USA, 2008.
- [55] C. von Ehrenfels. Über Gestaltqualitäten. *Vierteljahrsschrift für wissenschaftliche Philosophie*, 14:249–292, 1890.
- [56] A. El-Jaroudi and J. Makhoul. Discrete all-pole modeling. *IEEE Transactions on Signal Processing*, 39 (2):411–423, 1991.
- [57] D. P. W. Ellis. *Prediction-driven Computational Auditory Scene Analysis*. PhD thesis, Massachusetts Institute of Technology, 1996.
- [58] L. Ø. Endelt and A. La Cour-Harbo. Comparison of methods for sparse representation of music signals. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Philadelphia, USA, 2005.
- [59] D. Erdoğmuş, L. Vielva, and J. C. Príncipe. Nonparametric estimation and tracking of the mixing matrix for underdetermined blind source separation. In *Proc. International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, San Diego, USA, 2001.
- [60] S. Essid, G. Richard, and B. David. Instrument recognition in polyphonic music. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Philadelphia, USA, 2005.
- [61] M. R. Every. Separating harmonic and inharmonic note content from real mono recordings. In *Proc. Digital Music Research Network Doctoral Research Conference*, Glasgow, UK, 2005.
- [62] M. R. Every and J. E. Szymanski. Separation of synchronous pitched notes by spectral filtering of harmonics. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5), 2006.
- [63] N. H. Fletcher and T. D. Rossing. *The Physics of Musical Instruments*. Springer, 1998.
- [64] T. Galas and X. Rodet. Generalized discrete cepstral analysis for deconvolution of source-filter systems with discrete spectra. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, 1991.
- [65] M. M. Goodwin. *Adaptive Signal Models: Theory, Algorithms and Audio Applications*. PhD thesis, University of California, Berkeley, 1997.
- [66] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Music genre database and musical instrument sound database. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Baltimore, USA, 2003.

- [67] J. M. Grey. Multidimensional perceptual scaling of musical timbre. *Journal of the Acoustical Society of America*, 61:1270–1277, 1977.
- [68] R. Gribonval. Piecewise linear source separation. In *Wavelets: Applications in Signal and Image Processing, Proc. SPIE*, San Diego, USA, 2003.
- [69] R. Gribonval, L. Benaroya, E. Vincent, and C. Févotte. Proposals for performance measurement in source separation. In *Proc. International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, April 2003.
- [70] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U.K. Laine, and J. Huopaniemi. Frequency-warped signal processing for audio applications. *Journal of the Audio Engineering Society*, 48, No. 11, November 2000.
- [71] M. Helén and T. Virtanen. Perceptually motivated parametric representation for harmonic sounds for data compression purposes. In *Proc. International Conference on Digital Audio Effects (DAFX)*, London, UK, 2003.
- [72] J. Héroult, C. Jutten, and B. Ans. Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé. In *Proc. Colloque GRETSI*, pages 1017–1022, Nice, France, 1985.
- [73] P. Herrera, G. Peeters, and S. Dubnov. Automatic classification of musical instrument sounds. *Journal of New Music Research*, 32(1), 2003.
- [74] W. Hess. *Pitch Determination of Speech Signals*. Springer, 1983.
- [75] A. Horner. A simplified wavetable matching method using combinatorial basis spectra selection. *Journal of the Audio Engineering Society*, 49, No. 11, 2001.
- [76] C. Hourdin, G. Charbonneau, and T. Moussa. A multidimensional scaling analysis of musical instruments’ time-varying spectra. *Computer Music Journal*, 21, No. 2, 1997.
- [77] P. O. Hoyer. Non-negative Matrix Factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.
- [78] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for Independent Component Analysis. *Neural Computation*, 9 (7):1483–1492, 1997.
- [79] J. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.
- [80] ISO/IEC. 15938-4:2002 – Information technology – Multimedia content description interface – Part 4: Audio, July 2002.
- [81] K. Jensen. The timbre model. In *Proc. 144th Meeting of the Acoustical Society of America*, Cancún, Mexico, 2002.

- [82] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2002.
- [83] A. Jourjine, S. Rickard, and Ö. Yilmaz. Blind separation of disjoint orthogonal signals: Demixing N sources from 2 mixtures. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, Turkey, 2000.
- [84] J. Karvanen and A. Cichocki. Measuring sparseness of noisy signals. In *Proc. International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, April 2003.
- [85] K. Kashino and H. Murase. A sound source identification system for ensemble music based on template adaptation and music stream segregation. *Speech Communication*, 27, 1999.
- [86] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka. Application of the bayesian probability network to music scene analysis. In D. F. Rosenthal and H. G. Okuno, editors, *Computational Auditory Scene Analysis*, pages 115–137. Lawrence Erlbaum Associates, 1998.
- [87] T. Kinoshita, S. Sakai, and H. Tanaka. Musical sound source identification based on frequency component adaptation. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI) CASA Workshop*, 1999.
- [88] P. Kisilev, M. Zibulevsky, and Y.Y. Zeevi. A multiscale framework for blind separation of linearly mixed signals. *Journal of Machine Learning Research*, 4/7-8, 2003.
- [89] T. Kitahara, M. Goto, and H. G. Okuno. Musical instrument identification based on f0-dependent multivariate normal distribution. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, China, 2003.
- [90] A. Klapuri. *Signal Processing Methods for the Transcription of Music*. PhD thesis, Tampere University of Technology, Tampere, Finland, 2004.
- [91] A. Klapuri and M. Davy, editors. *Signal Processing Methods for Music Transcription*. Springer, 2006.
- [92] A. Klapuri, T. Virtanen, and M. Helén. Modeling musical sounds with an interpolating state model. In *Proc. European Signal Processing Conference (EUSIPCO)*, Antalya, Turkey, 2005.
- [93] M. Kuraya, A. Uchida, S. Yoshimori, and K. Umeno. Blind source separation of chaotic laser signals by Independent Component Analysis. *Optics Express*, 16 (2):725–730, 2008.
- [94] M. Lagrange, L. G. Martins, and G. Tzanetakis. Semi-automatic mono to stereo up-mixing using sound source formation. In *Proc. 122nd Convention of the Audio Engineering Society*, Vienna, Austria, May 2007.

- [95] M. Lagrange and G. Tzanetakis. Sound source tracking and formation using normalized cuts. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, USA, 2007.
- [96] S. Lakatos. A common perceptual space for harmonic and percussive timbres. *Perception & Psychophysics*, 62:1426–1439, 2000.
- [97] D. D. Lee and H. S. Seung. Learning the parts of objects by Non-negative Matrix Factorization. *Nature*, 401:799–791, 1999.
- [98] M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2):337–365, 2000.
- [99] J. K. Lin, D. G. Grier, and J. D. Cowan. Feature extraction approach to blind source separation. In *IEEE Workshop on Neural Networks for Signal Processing (NNSP)*, 1997.
- [100] A. Livshin and X. Rodet. Musical instrument identification in continuous recordings. In *Proc. International Conference on Digital Audio Effects (DAFX)*, Naples, Italy, 2004.
- [101] A. Livshin and X. Rodet. The significance of the non-harmonic “noise” versus the harmonic series for musical instrument recognition. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Victoria, Canada, 2006.
- [102] H.B. de Paula M.A. Loureiro and H.C. Yehia. Timbre classification of a single musical instrument. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, 2004.
- [103] R. C. Maher. Evaluation of a method for separating digitized duet signals. *Journal of the Acoustical Society of America*, 38 (12), 1990.
- [104] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41:3397–3415, December 1993.
- [105] L. G. Martins, J. J. Burred, G. Tzanetakis, and M. Lagrange. Polyphonic instrument recognition using spectral clustering. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.
- [106] S. McAdams, S. Winsberg, G. de Soete, and J. Krimphoff. Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes. *Psychological Research*, 58:177–192, 1995.
- [107] R. J. McAulay and T. F. Quatieri. Magnitude-only reconstruction using a sinusoidal speech model. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, New York, USA, 1984.
- [108] D. K. Mellinger. *Event Formation and Separation in Musical Sound*. PhD thesis, CCRMA, Stanford University, USA, 1991.

- [109] Y. Meron and K. Hirose. Separation of singing and piano sounds. In *Proc. International Conference on Spoken Language Processing (ICSLP)*, Sydney, Australia, 1998.
- [110] J. Meyer. *Akustik und musikalische Aufführungspraxis*. Verlag das Musikinstrument, 1972.
- [111] L. Miao and H. Qi. A blind source separation perspective on image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Minneapolis, USA, June 2007.
- [112] B. C. J. Moore and B. R. Glasberg. A revision of Zwicker's loudness model. *Acta Acustica*, 82, 1996.
- [113] K. Nakadai, H. G. Okuno, and H. Kitano. Real-time sound source localization and separation for robot audition. In *Proc. International Conference on Spoken Language Processing (ICSLP)*, Denver, USA, 2002.
- [114] T. Nakatani. *Computational Auditory Scene Analysis based on Residue-Driven Architecture and its Application to Mixed Speech Recognition*. PhD thesis, Kyoto University, Japan, 2002.
- [115] P. D. O'Grady and B. A. Pearlmutter. Hard-LOST: Modified k-means for oriented lines. In *Proc. Irish Signals and Systems Conference*, Belfast, UK, 2004.
- [116] P. D. O'Grady and B. A. Pearlmutter. Soft-LOST: EM on a mixture of oriented lines. In *Proc. International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, Granada, Spain, 2004.
- [117] P. D. O'Grady, B. A. Pearlmutter, and S. T. Rickard. Survey of sparse and non-sparse methods in source separation. *International Journal of Imaging Systems and Technology*, 15(1), 2005.
- [118] H. G. Okuno, T. Nakatani, and T. Kawabata. Understanding three simultaneous speeches. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, volume 1, pages 30–35, Nagoya, Japan, August 1997.
- [119] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy used by V1? *Vision Research*, 37:3311–3325, 1998.
- [120] A. V. Oppenheim, D. H. Johnson, and K. Steiglitz. Computation of spectra with unequal resolution using the Fast Fourier Transform. *Proceedings of the IEEE*, 59:299–301, February 1971.
- [121] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck. *Discrete-Time Signal Processing*. Prentice-Hall, 2nd Edition, 1999.

- [122] A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. One microphone singing voice separation using source-adapted models. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, 2005.
- [123] R. D. Patterson, M. Allerhand, and C. Giguere. Time-domain modelling of peripheral auditory processing: A modular architecture and software platform. *Journal of the Acoustical Society of America*, 98:1890–1894, 1995.
- [124] R. D. Patterson, I. Nimmo-Smith, D. L. Weber, and R. Milroy. The deterioration of hearing with age: Frequency selectivity, the critical ratio, the audiogram and speech threshold. *Journal of the Acoustical Society of America*, 72:1788–1803, December 1982.
- [125] M. S. Pedersen. *Source Separation for Hearing Aid Applications*. PhD thesis, Technical University of Denmark, 2006.
- [126] G. De Poli and P. Prandoni. Sonological models for timbre characterization. *Journal of New Music Research*, 26, 1997.
- [127] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*. Prentice Hall, 1978.
- [128] S. Rickard. Sparse sources are separated sources. In *Proc. European Signal Processing Conference (EUSIPCO)*, Florence, Italy, September 2006.
- [129] T. Ristaniemi and J. Joutsensalo. On the performance of blind source separation in CDMA downlink. In *Proc. International Workshop on Independent Component Analysis and Signal Separation (ICA)*, Aussois, France, 1999.
- [130] A. Röbel. *Neuronale Modelle Nichtlinearer Dynamischer Systeme mit Anwendung auf Musiksignale*. PhD thesis, Technical University of Berlin, 1993.
- [131] A. Röbel and X. Rodet. Efficient spectral envelope estimation and its application to pitch shifting and envelope preservation. In *Proc. International Conference on Digital Audio Effects (DAFX)*, Madrid, Spain, 2005.
- [132] D. F. Rosenthal and H.G. Okuno, editors. *Computational Auditory Scene Analysis*. Lawrence Erlbaum Associates, 1998.
- [133] Y. Sakuraba and H. G. Okuno. Note recognition of polyphonic music by using timbre similarity and direction proximity. In *Proc. International Computer Music Conference (ICMC)*, Singapore, 2003.
- [134] G.J. Sandell and W.L. Martens. Perceptual evaluation of principal-component-based synthesis of musical timbres. *Journal of the Audio Engineering Society*, 43, No. 12, December 1995.
- [135] E. D. Scheirer. *Music-Listening Systems*. PhD thesis, Massachusetts Institute of Technology, 2000.

- [136] M.R. Schroeder, B.S. Atal, and J.L. Hall. Optimizing digital speech coders by exploiting masking properties of the human ear. *Journal of the Acoustical Society of America*, 66, 1979.
- [137] D. Schwarz. Spectral envelopes in sound analysis and synthesis. Master's thesis, Universität Stuttgart / IRCAM, Paris, 1998.
- [138] X. Serra. Musical sound modeling with sinusoids plus noise. In C. Roads, S. Pope, A. Picialli, and G. De Poli, editors, *Musical Signal Processing*. Swets & Zeitlinger, 1997.
- [139] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (8):888–905, 2000.
- [140] G. Siamantas, M. R. Every, and J. E. Szymanski. Separating sources from single-channel musical material: A review and future directions. In *Proc. Digital Music Research Network Doctoral Research Conference*, London, UK, 2006.
- [141] M. Slaney. Auditory toolbox, version 2. Technical report, Interval Research Corporation, 1998.
- [142] M. Slaney. A critique of pure audition. In D. F. Rosenthal and H. G. Okuno, editors, *Computational Auditory Scene Analysis*. Lawrence Erlbaum Associates, 1998.
- [143] M. Slaney, D. Naar, and R. F. Lyon. Auditory model inversion for sound separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Adelaide, Australia, April 1994.
- [144] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, 2003.
- [145] J. O. Smith. *Physical Audio Signal Processing: for Virtual Musical Instruments and Digital Audio Effects, December 2005 Edition*. Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, 2005.
- [146] J. O. Smith. *Spectral Audio Signal Processing, March 2007 Version*. Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, March 2007.
- [147] J. O. Smith and J. S. Abel. Bark and ERB bilinear transforms. *IEEE Transactions on Speech and Audio Processing*, pages 697–708, November 1999.
- [148] M. Sterling, X. Dong, and M. Bocko. Representation of solo clarinet music by physical modeling synthesis. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, USA, March 2008.

- [149] S. S. Stevens, J. Volkman, and E. B. Newman. A scale for the measurement of the psychological magnitude of pitch. *Journal of the Acoustical Society of America*, 8 (3):185–190, 1937.
- [150] I. Takigawa, M. Kudo, and J. Toyama. Performance analysis of minimum ℓ_1 -norm solutions for underdetermined source separation. *IEEE Transactions on Signal Processing*, 52 (3):582–591, 2004.
- [151] A. C. Tang, B. A. Pearlmutter, N. A. Malaszenko, D. B. Phung, and B. C. Reeb. Independent components of magnetoencephalography: localization. *Neural Computation*, 14, No. 8:1827–1858, 2002.
- [152] D. Tardieu and X. Rodet. An instrument timbre model for computer aided orchestration. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, October 2007.
- [153] F. Theis and E. Lang. Formalization of the two-step approach to overcomplete BSS. In *Proc. Signal and Image Processing (SIP)*, Kauai, USA, 2002.
- [154] A. J. W. van der Kouwe, D. Wang, and G. J. Brown. A comparison of auditory and blind separation techniques for speech segregation. *IEEE Transactions on Speech and Audio Processing*, 9, No. 3, March 2001.
- [155] M. van Hulle. Clustering approach to square and non-square blind source separation. In *Proc. IEEE Workshop on Neural Networks for Signal Processing (NNSP)*, 1999.
- [156] B. L. Vercoe, W. G. Gardner, and E. D. Scheirer. Structured audio: Creation, transmission and rendering of parametric sound representations. *Proceedings of the IEEE*, 86, no. 5, 1998.
- [157] T. Verma, S. Levine, and T. Meng. Transient modeling synthesis: a flexible analysis/synthesis tool for transient signals. In *Proc. International Computer Music Conference (ICMC)*, Thessaloniki, Greece, 1997.
- [158] L. Vielva, D. Erdoğmuş, and J. C. Príncipe. Underdetermined blind source separation using a probabilistic source sparsity model. In *Proc. International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, San Diego, USA, 2001.
- [159] F. Villavicencio, A. Röbel, and X. Rodet. Improving LPC spectral envelope extraction of voiced speech by True-Envelope estimation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, 2006.
- [160] E. Vincent. *Modèles d’Instruments pour la Séparation de Sources et la Transcription d’Enregistrements Musicaux*. PhD Thesis, Université Paris VI, 2004.

- [161] E. Vincent. Musical source separation using time-frequency source priors. *IEEE Transactions on Audio, Speech and Language Processing*, 14 (1):91–98, 2006.
- [162] E. Vincent, C. Févotte, R. Gribonval, X. Rodet, É. Le Carpentier, L. Benaroya, A. Röbel, and F. Bimbot. A tentative typology of audio source separation tasks. In *Proc. International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, 2003.
- [163] E. Vincent, R. Gribonval, and C. Févotte. Performance measurement in blind audio source separation. *IEEE Transactions on Speech and Audio Processing*, 14 (4):1462–1469, 2006.
- [164] E. Vincent, M. G. Jafari, S. A. Abdallah, M. D. Plumbley, and M.E. Davies. Model-based audio source separation. Technical Report C4DM-TR-05-01, Queen Mary University, London, UK, 2006.
- [165] E. Vincent and M. D. Plumbley. A prototype system for object coding of musical audio. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, 2005.
- [166] E. Vincent and M. D. Plumbley. Single-channel mixture decomposition using bayesian harmonic models. In *Proc. International Conference on Independent Component Analysis and Blind Source Separation (ICA)*, Charleston, USA, 2006.
- [167] E. Vincent and X. Rodet. Underdetermined source separation with structured source priors. In *Proc. International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, Granada, Spain, September 2004.
- [168] T. Virtanen. Algorithm for the separation of harmonic sounds with time-frequency smoothness constraint. In *Proc. International Conference on Digital Audio Effects (DAFX)*, London, UK, 2003.
- [169] T. Virtanen. Sound source separation using sparse coding with temporal continuity objective. In *Proc. International Computer Music Conference (ICMC)*, Singapore, 2003.
- [170] T. Virtanen. Separation of sound sources by convolutive sparse coding. In *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing (SAPA)*, Jeju, Korea, 2004.
- [171] T. Virtanen. Unsupervised learning methods for source separation. In A. Klapuri and M. Davy, editors, *Signal Processing Methods for Music Transcription*. Springer, 2006.
- [172] T. Virtanen and Klapuri A. Separation of harmonic sounds using linear models for the overtone series. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Orlando, USA, 2002.

- [173] T. Virtanen and A. Klapuri. Separation of harmonic sound sources using sinusoidal modeling. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, Turkey, 2000.
- [174] T. Virtanen and A. Klapuri. Separation of harmonic sounds using multipitch analysis and iterative parameter estimation. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, USA, 2001.
- [175] H. Viste and G. Evangelista. A method for separation of overlapping partials based on similarity of temporal envelopes in multichannel mixtures. *IEEE Transactions on Audio, Speech and Language Processing*, 14 (3):1051–1061, 2006.
- [176] B. Wang and M.D. Plumbley. Musical audio stream separation by Non-negative Matrix Factorization. In *Proc. UK Digital Music Reserarch Network (DMRN) Summer Conf.*, 2005.
- [177] D. Wang and G.J. Brown. *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*. Wiley-IEEE Press, 2006.
- [178] Z. Wang, J. Cheung, Y. Xia, and J. Chen. Minimum fuel neural network and their applications to overcomplete signal representation. *IEEE Transactions on Circuits and Systems*, 47 (8):1146–1159, August 2000.
- [179] D. Wessel. Timbre space as musical control structure. *Computer Music Journal*, 3 (2):45–52, 1979.
- [180] J. Woodruff, B. Pardo, and R. Dannenberg. Remixing stereo music with score-informed source separation. In *Proc. International Conference on Music Information Retrieval (ISMIR)*, Victoria, Canada, 2006.
- [181] Ö. Yilmaz and S. Rickard. Blind separation of speech mixtures via time-frequency masking. *IEEE Transactions on Signal Processing*, 52, No. 7, July 2004.
- [182] B. A. Zibulevsky, P. Pearlmutter, P. Bofill, and P. Kisilev. Blind source separation by sparse decomposition. In S. J. Roberts and R. M. Everson, editors, *Independent Component Analysis: Principles and Practice*. Cambridge, 2001.
- [183] E. Zwicker and H. Fastl. *Psychoacoustics. Facts and Models*. Springer, 1990.

Index

- ℓ_0 norm, 28
- ℓ_1 norm, 29, 47, 75, 157
- ℓ_2 norm (Euclidean), 29, 47
- ℓ_p norm, 28
- ℓ_ϵ norm, 29

- AB stereophony, 21
- Adaptive transforms, 22
- ADSR envelope, 85, 94, 98, 131, 173
- Atonality, 11, 68
- Auditory scales, 57
 - Bark scale, 57
 - ERB scale, 58
 - Mel scale, 59, 120
- Auditory Scene Analysis (ASA), 48, 92
- Autoregressive model, 23, 86
- Azimuth, 17

- Bark scale, 57
- Basis decomposition, 23
- Binaural recording, 21
- Blind Source Separation (BSS), 2, 11
- Bridge hill, 116

- Cent, 137
- Central limit theorem, 43
- Cepstrum, 87, 91, 120
- Chromatic scale, 55
- Close miking, 20
- Clustering, 44, 94
- Cochleagram, 49
- Cocktail party effect, 1, 68
- Computational Auditory Scene Analysis (CASA), 48, 51, 100
 - Data-driven, 49
 - Prediction-driven, 49
- Computer-Aided Orchestration (CAO), 95
- Constant Overlap-Add (COLA), 61, 90
- Constant-Q, 51, 55
- Constant-Q Transform (CQT), 56
- Correlogram, 49
- Critical bands, 57, 130
- Cross-validation, 103, 134

- Direct Injection, 20
- Directionality pattern, 19
- Discrete All-Pole, 87
- Discrete Fourier Transform (DFT), 24
- Disjointness, 66
- Dynamic Time Warping (DTW), 174

- Eigenvalue decomposition (EVD), 35
- Envelope Interpolation (EI), 102
- Equal Rectangular Bandwidth (ERB) scale, 58
- Even-determined separation, 9, 15, 45
- Exponential distribution, 27

- F-Measure, 124
- FastICA, 43
- Filter bank, 53
 - Critically downsampled, 53
 - Modulated, 53
- Filter bank summation (FBS), 62
- Formant, 85, 116
- Fourier Series, 89
- Frequency support, 25
- Frequency warping, 53
- Frobenius norm, 38, 67

- Gabor expansion, 26
 - Generalized, 54
- Gaussian distribution, 27, 42, 47, 108
- Gaussian Mixture Models (GMM), 6, 94, 108, 110, 130
- Gaussian Process (GP), 111, 114
- Gestalt psychology, 48
- Gradient descent, 43
- Gram matrix, 77

- Head Related Transfer Function, 22
- Heterophony, 11
- Hidden Markov Models (HMM), 95, 110, 130, 173
- Homophony, 11, 68
- Homorhythmic, 11, 68
- Hough transform, 45

- Impulse-type distribution, 28, 41
- Independent Component Analysis (ICA), 1, 23, 41, 98, 128
- Independent Subspace Analysis (ISA), 98, 128, 156
- Infomax principle, 44
- Intensity stereophony, 19
- Inter-channel Intensity Difference, 19, 45, 156
- Inter-channel Phase Difference, 21, 156
- Interpolation
 - Cubic, 89, 111
 - Linear, 88

- Kernel density estimation, 45, 73

- Kullback-Leibler divergence, 109, 128
 Kurtosis, 29, 43
- Laplacian distribution, 28, 47, 76, 82
 Lexicographic ordering, 26, 27, 54
 Liftering, 88, 91
 Linear Discriminant Analysis (LDA), 112, 173
 Linear Predictive Coding (LPC), 86
- Mahalanobis distance, 109
 Main-lobe width, 53
 Matching Pursuit, 32
 Maximum A Posteriori (MAP), 38, 46
 Maximum Likelihood, 44, 138, 162, 173
 Mel Frequency Cepstral Coefficients (MFCC), 6, 30, 59, 93, 94, 119
 Mel scale, 59, 120
 Monody, 9
 Monophony, 9
 Moore-Penrose pseudoinverse, 47
 MPEG-7, 94, 97
 MS stereophony, 19
 Multidimensional Scaling (MDS), 93
 Music Information Retrieval (MIR), 4, 94, 110, 119
 Musical noise, 48, 76
 Mutual information, 43
- Negentropy, 30, 43
 Nonnegative Matrix Factorization (NMF), 129
 Nonnegative Sparse Coding (NSC), 129
 Normalized Cut, 122
- Onset detection, 134
 Overcomplete decomposition, 31
 Overdetermined separation, 9, 15, 46
 Overdubbing, 19
 Overlap-Add (OLA), 62
- Panning, 20
 Parseval's theorem, 67, 71
 Partial Indexing (PI), 101
 Partial tracking, 92
 Parzen windows, 45, 73
 Peak picking, 90
 Piecewise linear source separation, 55
 Polyphony (musical texture), 11
 Potential function, 45, 73
 Precision, 124
 Preserved Signal Ratio (PSR), 67
- Prince Shōtoku Computers, 2
 Principal Component Analysis (PCA), 23, 24, 32, 42, 46, 93, 98, 131
 Principal Curves, 173
 Probability distribution, 27
 - Gaussian, 27, 42, 47, 108
 - General exponential, 27
 - Impulse-type, 28, 41
 - Laplacian, 28, 47, 76, 82
 - Subgaussian, 28, 30
 - Supergaussian, 28, 30
 - Uniform, 28
- Prototype curve, 111
 Prototype envelope, 114
 Pseudoinverse, 47
- QR algorithm, 35
 Quality factor Q , 55
- Recall, 123
 Relative Spectral Error (RSE), 106
- Scalogram, 51
 Self Organizing Maps (SOM), 93
 Semi-blind Source Separation (SBSS), 2, 39, 83
 Semitone, 55
 Short-Time Fourier Transform, 26, 51
 - Filter bank interpretation, 55
 - Generalized, 54
- Shortest path synthesis, 47, 75
 Signal to Error Ratio (SER), 76, 146
 Singular Value Decomposition (SVD), 35
 Sinusoidal modeling, 89, 129
 Source to Artifacts Ratio (SAR), 78, 160, 166, 170
 Source to Distortion Ratio (SDR), 78, 166, 170
 Source to Interference Ratio (SIR), 67, 78, 170
 Source-filter model, 86
 Sparsity, 22, 27
 Spectral envelope, 85
 Spectral Modeling Synthesis (SMS), 93
 Spectral Signal to Error Ratio (SSER), 146, 166
- Stereophony, 17
 - AB stereophony, 21
 - Intensity stereophony, 19
 - MS stereophony, 19

-
- Time-of-arrival stereophony, 21
 - XY stereophony, 19
 - Structured Audio Coding (SAC), 5, 94, 95
 - Subgaussian, 28, 30
 - Supergaussian, 28, 30
 - Sweet spot, 17

 - Temporal envelope, 85
 - Texture (musical), 9
 - Timbre, 83
 - Timbre space, 93
 - Time–frequency decompositions, 25
 - Time–frequency masking, 47
 - Time-of-arrival stereophony, 21
 - Tonal fusion, 4
 - Tonality, 11, 68
 - Transient, 83
 - Transient Modeling Synthesis (TMS), 93
 - True Envelope, 88

 - Uncertainty principle, 26, 53
 - Underdetermined separation, 9, 15
 - Uniform distribution, 28

 - Vibrato, 94, 99

 - W-Disjoint Orthogonality (WDO), 66
 - Wavelet, 51, 57
 - Weft, 49
 - Whitening, 29, 35, 98, 109, 114
 - Wiener filtering, 48, 130

 - XY stereophony, 19