
Iterative Separation of Note Events from Single-Channel Polyphonic Recordings

Alejandro Delgado Castro

B.Sc. Electronic Engineering (2005)

M.Sc. Electrical Engineering (2010)

Doctor of Philosophy

University of York

Electronic Engineering

July 2019

Department of Electronic Engineering
UNIVERSITY *of* York

**Iterative Separation of Note Events from
Single-Channel Polyphonic Recordings**

by Alejandro Delgado Castro

This is to certify that the thesis presented here meets the accepted standards with respect to scope, quality and originality as a dissertation for the degree of
Doctor of Philosophy in Electronic Engineering.



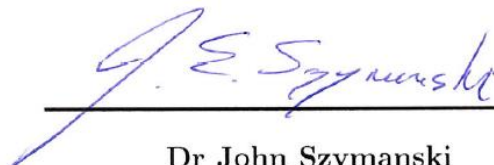
Professor Simon Dixon

Queen Mary College, University of London
External Examiner



Mr Anthony Tew

Department of Electronic Engineering, University of York
Internal Examiner



Dr John Szymanski

Department of Electronic Engineering, University of York
Supervisor

Date of Examination (VIVA): September 16th, 2019.

*A mi familia,
a la sangre que corre
por mis venas.*

To my family.

Abstract

This thesis is concerned with the separation of audio sources from single-channel polyphonic musical recordings using the iterative estimation and separation of note events. Each event is defined as a section of audio containing largely harmonic energy identified as coming from a single sound source. Multiple events can be clustered to form separated sources. This solution is a model-based algorithm that can be applied to a large variety of audio recordings without requiring previous training stages.

The proposed system embraces two principal stages. The first one considers the iterative detection and separation of note events from within the input mixture. In every iteration, the pitch trajectory of the predominant note event is automatically selected from an array of fundamental frequency estimates and used to guide the separation of the event's spectral content using two different methods: time-frequency masking and time-domain subtraction. A residual signal is then generated and used as the input mixture for the next iteration. After convergence, the second stage considers the clustering of all detected note events into individual audio sources.

Performance evaluation is carried out at three different levels. Firstly, the accuracy of the note-event-based multipitch estimator is compared with that of the baseline algorithm used in every iteration to generate the initial set of pitch estimates. Secondly, the performance of the semi-supervised source separation process is compared with that of another semi-automatic algorithm. Finally, a listening test is conducted to assess the audio quality and naturalness of the separated sources when they are used to create stereo mixes from monaural recordings.

Future directions for this research focus on the application of the proposed system to other music-related tasks. Also, a preliminary optimisation-based approach is presented as an alternative method for the separation of overlapping partials, and as a high resolution time-frequency representation for digital signals.

Table of Contents

| | |
|--|-----------|
| Table of Contents | 9 |
| List of Figures | 15 |
| List of Tables | 25 |
| List of Acronyms | 27 |
| Acknowledgement | 31 |
| Author's Declaration | 33 |
| 1 Introduction | 35 |
| 1.1 Motivation | 35 |
| 1.2 The Ability of Hearing | 36 |
| 1.3 An Overview of the Proposed Solution | 37 |
| 1.4 Potential Applications | 39 |
| 1.4.1 Wireless Communications | 39 |
| 1.4.2 Geophysical Exploration | 39 |
| 1.4.3 Medical Signal Processing | 40 |
| 1.4.4 Image Enhancement and Recognition | 40 |
| 1.4.5 Audio and Music Technology | 40 |
| 1.5 Contributions | 42 |
| 1.6 Thesis Outline | 43 |
| 2 Audio Signals: Representation and Modelling | 45 |
| 2.1 Preamble | 45 |
| 2.2 Musical Theory | 46 |

| | | |
|----------|--|-----------|
| 2.2.1 | Terms and Definitions | 46 |
| 2.2.2 | Tones and Notes | 47 |
| 2.2.3 | Cross-section of a Note | 47 |
| 2.2.4 | Partials, Harmonics and Overtones | 48 |
| 2.3 | Classification of Sounds | 49 |
| 2.3.1 | Harmonic Sounds | 49 |
| 2.3.2 | Inharmonic Sounds | 50 |
| 2.3.3 | Monophonic and Polyphonic Sounds | 50 |
| 2.4 | Models for Mixing Processes | 51 |
| 2.4.1 | Multi-Channel Models | 51 |
| 2.4.2 | Single-Channel Instantaneous Model | 52 |
| 2.4.3 | Possible Scenarios | 52 |
| 2.5 | Representation of Audio Signals | 53 |
| 2.5.1 | Short-Time Fourier Transform | 53 |
| 2.5.2 | Wigner-Ville Distribution | 55 |
| 2.5.3 | Wavelet Transform | 55 |
| 2.5.4 | Continuous Complex Wavelet Transform | 57 |
| 2.5.5 | Cochleagram | 58 |
| 2.5.6 | Time-Frequency Reassignment | 59 |
| 2.6 | Signal Decomposition Models | 60 |
| 2.6.1 | Sinusoidal Modelling | 60 |
| 2.6.2 | Independent Component Analysis | 61 |
| 2.6.3 | Independent Subspace Analysis | 63 |
| 2.6.4 | Nonnegative Matrix Factorisation | 63 |
| 2.6.5 | Sparse Coding | 64 |
| 2.6.6 | Wavelet Analysis | 65 |
| 2.6.7 | Matching Pursuit | 65 |
| 2.7 | Summary | 67 |
| 3 | Audio Source Separation Techniques | 69 |
| 3.1 | Preamble | 69 |
| 3.2 | Basic Problem | 69 |
| 3.3 | Separation of Estimated Sources | 70 |

| | | |
|----------|--|-----------|
| 3.3.1 | Sinusoidal Synthesis | 71 |
| 3.3.2 | Time-Frequency Masking | 72 |
| 3.4 | Model-driven Separation Approaches | 73 |
| 3.4.1 | Computational Auditory Scene Analysis | 73 |
| 3.4.2 | Statistical Approaches | 74 |
| 3.4.3 | Harmonicity | 76 |
| 3.4.4 | Separation of Overlapping Harmonics | 77 |
| 3.5 | Data-driven Separation Approaches | 77 |
| 3.5.1 | Deep Neural Networks | 78 |
| 3.5.2 | Convolutional Neural Networks | 79 |
| 3.6 | Prior Information in Source Separation | 79 |
| 3.6.1 | Non-Informed Methods | 79 |
| 3.6.2 | Informed Methods | 80 |
| 3.7 | Performance Measurement and Evaluation | 83 |
| 3.7.1 | Blind Source Separation Evaluation | 83 |
| 3.7.2 | Perceptual Evaluation of Audio Source Separation | 84 |
| 3.8 | Summary | 84 |
| 4 | An Iterative Note Event-based Multipitch Estimator | 87 |
| 4.1 | Preamble | 87 |
| 4.2 | Related Work in Multipitch Estimation | 88 |
| 4.2.1 | Multipitch Detectors | 89 |
| 4.2.2 | Note Trackers | 90 |
| 4.2.3 | Multipitch Streamers | 90 |
| 4.3 | Proposed System | 91 |
| 4.3.1 | Note Events | 91 |
| 4.3.2 | Overview of the System | 92 |
| 4.3.3 | Input Signal | 94 |
| 4.3.4 | Saliency Measurement | 94 |
| 4.3.5 | Predominant Note Event | 95 |
| 4.3.6 | Interfering Events and Preservation Rates | 98 |
| 4.3.7 | Interpolation and Boundary Correction | 103 |
| 4.3.8 | Separation of Spectral Content | 104 |

| | | |
|----------|---|------------|
| 4.3.9 | Convergence of the Iterative Stage | 105 |
| 4.3.10 | Power-based Revision | 106 |
| 4.4 | Evaluation of Performance | 107 |
| 4.4.1 | Methodology | 107 |
| 4.4.2 | Seven Simultaneous Violins | 108 |
| 4.4.3 | Notes in Harmonic Relation | 109 |
| 4.4.4 | Real Music | 111 |
| 4.5 | Summary | 115 |
| 5 | Semi-supervised Source Separation based on Clustering of Note Events | 117 |
| 5.1 | Preamble | 117 |
| 5.2 | Spectral Peak Picking | 118 |
| 5.3 | Spectral Peak Parameters | 121 |
| 5.4 | Tracking Harmonics | 124 |
| 5.4.1 | Effects of Inharmonicity | 124 |
| 5.4.2 | Effects of Vibrato | 126 |
| 5.5 | Separation of Spectral Content | 127 |
| 5.5.1 | Detection and Classification of Overlapping Partial | 128 |
| 5.5.2 | Semi-Overlapping Partial | 132 |
| 5.5.3 | Fully-Overlapping Partial | 134 |
| 5.6 | Note Event Extraction | 137 |
| 5.6.1 | Time-Frequency Masking | 137 |
| 5.6.2 | Time-Domain Subtraction | 143 |
| 5.6.3 | Time-Frequency Masking vs. Time-Domain Subtraction | 146 |
| 5.6.4 | Number of Harmonics Extracted per Frame | 146 |
| 5.7 | Clustering of Note Events | 148 |
| 5.8 | Evaluation of Performance | 149 |
| 5.8.1 | Seven Simultaneous Violin Notes | 150 |
| 5.8.2 | Influence of the Extraction Order | 152 |
| 5.8.3 | Influence of the Automatic Pitch Tracking Stage | 154 |
| 5.8.4 | Influence of the Number of Harmonics Extracted per Frame | 155 |
| 5.8.5 | Source Separation in Real Music | 156 |
| 5.8.6 | Potential Sources of Error | 161 |

| | | |
|----------|--|------------|
| 5.9 | Summary | 164 |
| 6 | Mono-to-Stereo Upmixing | 169 |
| 6.1 | Preamble | 169 |
| 6.2 | Previous Approaches | 170 |
| 6.3 | Proposed Method | 171 |
| 6.3.1 | Source Panning | 172 |
| 6.4 | Evaluation of Performance | 173 |
| 6.4.1 | Database | 173 |
| 6.4.2 | Listening Test | 173 |
| 6.4.3 | Results and Discussion | 174 |
| 6.5 | Summary | 177 |
| 7 | Conclusions and Further Work | 179 |
| 7.1 | Final Remarks | 179 |
| 7.1.1 | Note Event-based Multipitch Estimation | 180 |
| 7.1.2 | Semi-Supervised Audio Source Separation | 181 |
| 7.2 | Further Work | 182 |
| 7.2.1 | Extraction of Multiple Note Events per Iteration | 182 |
| 7.2.2 | Onset Detection | 184 |
| 7.2.3 | WAV-to-MIDI Conversion | 185 |
| 7.2.4 | Automatic Clustering of Note Events | 185 |
| 7.2.5 | Optimisation and Beyond | 186 |
| 7.3 | Plan for the Future | 196 |
| | References | 197 |

List of Figures

| | | |
|------|--|----|
| 1.1 | Human auditory system | 36 |
| 1.2 | Block diagram of the proposed estimation/separation process showing its main components and the way they relate to each other. | 38 |
| 2.1 | Sections of a musical note | 48 |
| 2.2 | A horn playing the note E3. (a) Waveform. (b) Magnitude spectrum. | 49 |
| 2.3 | A sound produced by a tambourine. (a) Waveform. (b) Magnitude spectrum. | 50 |
| 2.4 | Different sounds in the frequency domain. (a) Monophonic signal in which a violin is playing the note F4, (b) Polyphonic signal comprising a violin and a horn. Harmonic peaks of the violin are marked with asterisks. | 51 |
| 2.5 | Hanning-windowed spectrogram of a speech signal consisting of a male voice pronouncing the words <i>brilliant</i> , <i>arresting</i> , <i>extravagant</i> . The frame size is 2048 samples, with 50% overlap, while the sampling frequency is 44.1 kHz. | 54 |
| 2.6 | Wigner-Ville Distribution (WVD) of a pure tone with frequency of 1 kHz. | 55 |
| 2.7 | Four different wavelet functions. | 56 |
| 2.8 | Scalogram of the sum of two pure tones with frequencies 200 Hz and 500 Hz. The sampling frequency was set to 4 kHz. | 57 |
| 2.9 | Continuous Complex Wavelet Transform (CCWT) of the speech utterance <i>brilliant</i> , <i>arresting</i> , <i>extravagant</i> . Note the logarithmic y-axis. Image generated and provided on request by Jesús Ponce de León Vázquez (jponce@unizar.es) in September 2016. | 58 |
| 2.10 | Cochleagram of the same speech excerpt presented in Figures 2.5 and 2.9. This graph was generated using the code by Ning Ma [30]. | 59 |
| 2.11 | Time-Frequency reassignment using a frequency-modulated sinusoid as an input signal. | 60 |

| | | |
|------|--|-----|
| 2.12 | Decomposition process using ICA and two synthetic input signals. | 62 |
| 2.13 | Nonnegative Matrix Factorisation (NMF) decomposition of an input signal made up from three pure tones at frequencies 1 kHz, 3 kHz and 2 kHz. (a) Spectrogram, (b) Basis vectors \mathbf{W}_v , and (c) Activations \mathbf{H}_a | 64 |
| 2.14 | Decomposition of an input signal using matching pursuit. | 66 |
| 3.1 | Block diagram of a general separation process. | 70 |
| 3.2 | Sinusoidal synthesis of two different notes. (Top) Original signal. (Bottom) Sinusoidal synthesis. | 71 |
| 3.3 | Blind harmonic adaptive decomposition user interface [79]. | 82 |
| 3.4 | Interactive Sound Source Separation Editor (ISSE) user interface [29]. | 82 |
| 4.1 | Simplified block diagram of the proposed multipitch detection system. The iterative estimation/separation stage is delimited by the dashed line border. | 93 |
| 4.2 | Selection of the first predominant note event in a mixture of violin and flute. The violin plays the notes D \sharp 4 and B4 while the flute plays a G4. (a) Original unmixed sources. (b) Ground-truth pitch trajectories. (c) Raw pitch estimates during the first iteration. Markers are used to identify estimates in the same level. (d) Predominant note event and other detected events during the first iteration. | 97 |
| 4.3 | Predominance of note events during the first iteration of the system on a mixture of violin and flute. The contribution of the total salience $\hat{\mathcal{S}}(\tau)$ and duration $\eta\hat{\mathcal{D}}(\tau)$ in the predominance of each note event, is shown using different shades in each of the vertical bars. | 98 |
| 4.4 | Absolute magnitude spectra of two different audio signals. (a) One violin playing the note A3. (b) Two violins playing notes A3 and A4. The spectra shown were computed using a frame size of 2048, 87.5% overlap, $f_s = 44.1$ kHz, no zero-padding, and a Hanning window. | 99 |
| 4.5 | Pitch contours generated with Duan's algorithm. (a) One violin playing the note A3. (b) Two violins simultaneously playing the notes A3 and A4. | 99 |
| 4.6 | Analysis of a mixture consisting of two violin notes: A3 and A4. (a) Predominant note event selection from a set of eight candidates. (b) Predominant note event at 220 Hz and four interfering events: one real at 440 Hz and three spurious at 660 Hz, 880 Hz and 1320 Hz. | 100 |

| | | |
|------|--|-----|
| 4.7 | Comparison between the pitch-scaled contours of four potentially interfering events detected during the first iteration of the system on a mixture of two octave-related violin notes. | 103 |
| 4.8 | Boundary correction. (a) Original and adjusted pitch trajectory of the predominant note event in a mixture of two violin notes. (b) Saliency contour and significance threshold at 5% of the maximum for the same mixture of two violin notes. | 104 |
| 4.9 | Estimated pitch trajectories for a test mixture consisting of seven simultaneous violin notes. Markers identify pitch estimates of a particular note event, while numbers (IES-TFM and IES-TDS) indicate the order of extraction. Solid lines represent the ground-truth pitch contours. | 109 |
| 4.10 | Estimated pitch trajectories for a test mixture from set 7, consisting of four simultaneous flute notes (C4, C5, G5, and C6). Markers identify pitch estimates of a particular note event, while numbers (IES-TFM and IES-TDS) indicate the order of extraction. | 111 |
| 4.11 | Pitch trajectories of the note events detected with the IES-TDS algorithm on four excerpts with polyphony 4 of the Bach10 database. Numbers indicate the extraction order while the ground-truth pitch trajectories are marked with solid lines. | 114 |
| 5.1 | Peak-picking method applied to three different notes in isolation. (a-b) Violin B4 ($f_0 = 490$ Hz), (c-d) Piano A3 ($f_0 = 210$ Hz), and (e-f) Cello D2 ($f_0 = 73$ Hz). . . | 121 |
| 5.2 | Parameter estimation for a harmonic partial associated with a real violin note. Reference points are marked with asterisks while estimated parameters are marked with diamonds. | 123 |
| 5.3 | Estimation of the integrated partial amplitude for two different notes, considering five frequency bins to define the significance area of each partial. | 123 |
| 5.4 | Normalised mean-squared error between the real and estimated harmonic frequencies for a set of synthetic notes with different pitches and levels of inharmonicity. (Top) Using a ground-truth pitch trajectory. (Bottom) Using an automatically estimated pitch trajectory. | 125 |

5.5 Spectrograms and harmonic magnitude contours of two real piano notes in isolation. Estimated harmonic trajectories in each frame are shown with circles in (a) and (b). These results were obtained using automatically estimated pitch trajectories. 126

5.6 Spectrograms and harmonic magnitude contours of two real violin notes in isolation. Estimated harmonic trajectories in each frame are shown with circles in (a) and (b). These results were obtained using automatically estimated pitch trajectories. 128

5.7 Average frequency deviation between estimated components during the estimation of their centre frequencies in a number of synthetic mixtures consisting of two sinusoidal components with different phase differences. 130

5.8 Decomposition of two different spectral peaks using Parsons' method. (a) Spectral peak associated with a single harmonic of a real viola note. (b) Shared peak associated with two semi-overlapping harmonics in a mixture of viola and clarinet. In both cases, an asterisk (*) and a circle (o) are used to mark the peak positions of the dominant and the secondary components, respectively. Notice the magnitude of the secondary component in (a) is very low. 133

5.9 Original and estimated components associated with the semi-overlapping peak shown in Figure 5.8(b), taken from a mixture of viola and clarinet. (a) Fundamental partial of the clarinet note A \sharp 5. (b) Second harmonic of the viola note A4. 134

5.10 Incomplete separation of two fully-overlapping partials in a mixture involving two violins playing the notes A3 ($f_0 = 220$ Hz) and A4, where note A3 is the predominant one and two interferers are centred at 440 Hz and 880 Hz. (Top) Separation of the second harmonic of note A3, centred at 440 Hz, which collides with the first harmonic of the interferer at 440 Hz, thus the gain $G_2 = 0.75$ is applied. (Bottom) Separation of the fourth harmonic of note A3, centred at 880 Hz, which simultaneously collides with the second harmonic of the interferer at 440 Hz and with the first harmonic of the interferer at 880 Hz, hence the gain $G_4 = 0.5$ is applied. (a,d) Observed overlapping partial, (b,e) Attenuated dominant component, (c,f) Comparison between the original harmonic and the attenuated dominant component. 135

| | | |
|------|--|-----|
| 5.11 | Estimated spectral content of the violin note A3 in a mixture containing the violin notes A3 and A4. (a) Comparison between the estimated spectral content and the original mixture. (b) Comparison between the estimated spectral content and the original note A3. (c) Residual. | 136 |
| 5.12 | Extraction masks for two different frequency components from a mixture of viola (A4) and clarinet (A \sharp 5), where the viola has been selected as the predominant note event. | 139 |
| 5.13 | A single frame of the time-frequency mask used to extract a viola note A4 from a mixture in which the clarinet note A \sharp 5 is also present. (a) Spectrum of the input mixture and extraction mask. (b) Original and estimated spectra of the viola note A4. | 140 |
| 5.14 | First note event extraction from a mixture of saxophone and bassoon. (a) Spectrum of the mixture and extraction mask for the saxophone note E3. (b) Original and estimated spectra of the saxophone note. Spurious interferers are present at 330 Hz and 430 Hz (with preservation rates $\alpha_1 = \alpha_2 = 0.25$). | 141 |
| 5.15 | Second note event extraction from a mixture of saxophone and bassoon, after removing the saxophone note. (a) Input spectrum and extraction mask for the bassoon note A2. (b) Original and estimated spectra of the bassoon note. A spurious interferer is present at 330 Hz (with preservation rate $\alpha = 0.25$). | 141 |
| 5.16 | Time domain subtraction applied to a set of audio mixtures with polyphony two. (Top) Input mixture. (Middle) Predominant note event. (Bottom) Residual of the first iteration. | 145 |
| 5.17 | Comparison between the extracted note events in Figure 5.16 and the original ones. (Top) Original note. (Bottom) Extracted note event. | 145 |
| 5.18 | Extraction of the spectral content associated with a cello note (D2) in isolation using time-domain subtraction and time-frequency masking. (a) Spectra of the original and estimated cello notes (using both extraction methods). (b) Spectra of the corresponding residual signals. | 147 |
| 5.19 | Clustering of estimated note events from a mixture of viola and clarinet. (a) Estimated pitch trajectories of note events. (b) Extracted note events: (1) Viola A4, (2) Clarinet D \sharp 6, (3) Clarinet A \sharp 5, (4) Clarinet D \sharp 5, and (5) Clarinet G5. The extraction order is shown with numbers. | 148 |

| | | |
|------|---|-----|
| 5.20 | Residual signal obtained after extracting all five note events presented in Figure 5.19(b). | 149 |
| 5.21 | Comparison between the original and estimated sources from a mixture of viola and clarinet. (Top) Original source, (Bottom) Estimated source. | 150 |
| 5.22 | Separation performance for a mixture of seven synchronous violin notes of the same duration using four different algorithms: the proposed system with time-frequency masking (IES-TFM), the proposed system with time-domain subtraction (IES-TDS), the MIDI-informed system in [28] (MIDI), and the iterative residual-based system in [23] (IDG). | 152 |
| 5.23 | Separation performance for 12 mixtures of single musical notes using the proposed time-frequency masking approach and three note-event extraction orders. | 153 |
| 5.24 | Separation performance on the same 12 mixtures of single musical notes using the proposed time-domain subtraction approach and three note-event extraction orders. | 154 |
| 5.25 | Separation performance on 12 mixtures of single notes using a combination of the proposed iterative pitch detection stage and each of the extraction approaches: time-frequency masking (IES-TFM) and time-domain subtraction (IES-TDS). | 155 |
| 5.26 | Separation performance exhibited by the proposed system on 12 mixtures of single notes, using time-frequency masking for extraction and two different ways to define the maximum number of harmonics extracted in every frame: Fixed and Pitch Dependent. | 156 |
| 5.27 | Separation performance exhibited by the proposed system on 12 mixtures of single notes, using time-domain subtraction for extraction and two different ways to define the maximum number of harmonics extracted in every frame: Fixed and Pitch Dependent. | 156 |
| 5.28 | Separation performance on 12 test mixtures with polyphony 2 in Group 1 (two harmonic sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle). | 158 |
| 5.29 | Separation performance on 12 test mixtures with polyphony 3 in Group 2 (three harmonic sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle). | 159 |

| | | |
|------|--|-----|
| 5.30 | Separation performance on 12 test mixtures with polyphony 3 in Group 3 (two harmonic and one percussive sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle). | 159 |
| 5.31 | Separation performance on 12 test mixtures with polyphony 4 in Group 4 (four harmonic sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle). | 160 |
| 5.32 | Original and separated sources for a representative test mixture from Group 1. (Top) Violin. (Bottom) Clarinet. | 162 |
| 5.33 | Original and separated sources for a representative test mixture from Group 2. (Top) Violin. (Middle) Clarinet. (Bottom) Tenor saxophone. | 163 |
| 5.34 | Original and separated sources for a representative test mixture from Group 3. (Top) Violin. (Middle) Clarinet. (Bottom) Percussion. | 164 |
| 5.35 | Original and separated sources for a representative test mixture from Group 4. (From Top to Bottom) Violin, Clarinet, Tenor Saxophone and Bassoon. | 165 |
| 6.1 | Simplified block diagram of the proposed mono-to-stereo conversion system. . . . | 171 |
| 6.2 | Panning of the sources. (a) Diagram of the stereo setup showing the positioning of two sources at different angles. (b) Left and right gains following the constant power law. | 172 |
| 6.3 | Sample trial of the listening test on Qualtrics. | 175 |
| 6.4 | Results of the listening test. (a) Original data from all participants. (b) Mean values per item and 95% confidence intervals. (Mono) Original monaural mixture, (ISSE) Interactive Sound Source Separation Editor, (IES) Proposed system, and (IES+R) Proposed system with final residual panned in the middle. | 176 |
| 7.1 | Audio mixture consisting of violin and clarinet. (a) Note events detected during the first iteration. (b) Predominance of each detected note event. | 183 |
| 7.2 | F-Measures obtained using a preliminary onset detector based on the proposed separation framework and three other onset detectors [141]. | 184 |
| 7.3 | An example of WAV-to-MIDI conversion using an audio excerpt from the Bach10 database as an input recording. (a) Estimated pitch contours. (b) Equivalent MIDI piano roll. | 185 |

| | | |
|------|---|-----|
| 7.4 | Magnitude, real and imaginary spectra for two pure sinusoids having the same amplitudes and frequencies, but different phase angles. | 188 |
| 7.5 | Three different sinusoidal components with centre frequencies separated 30 Hz from each other. (Top) $s_1(t) \rightarrow S_1(f)$, (Middle) $s_2(t) \rightarrow S_2(f)$, and (Bottom) $s_3(t) \rightarrow S_3(f)$ | 190 |
| 7.6 | Input mixture generated by mixing the sinusoidal components in Figure 7.5. . . . | 190 |
| 7.7 | Results of the optimisation-based estimation process. (a) Identified components in the time domain. (b) Comparison between the observed and approximated overlapping partials. | 190 |
| 7.8 | Two sinusoidal components with centre frequencies separated only 5 Hz from each other. (Top) $s_1(t) \rightarrow S_1(f)$, and (Bottom) $s_2(t) \rightarrow S_2(f)$ | 191 |
| 7.9 | Input mixture generated by mixing the sinusoidal components in Figure 7.8. . . . | 191 |
| 7.10 | Results of the optimisation-based estimation process. (a) Identified components in the time domain. (b) Comparison between the observed and approximated overlapping partials. | 191 |
| 7.11 | Ideal time-frequency representation of a synthetic input signal consisting of several sinusoidal components with different centre frequencies and occurring at different times. | 192 |
| 7.12 | Spectrogram of the input signal characterised in Figure 7.11 using a frame size of 1024 samples ($f_s = 44.1$ kHz). | 192 |
| 7.13 | Spectrogram of the input signal characterised in Figure 7.11 using a frame size of 8192 samples ($f_s = 44.1$ kHz). | 192 |
| 7.14 | Optimisation-based process used to improve the frequency resolution of the spectrogram in Figure 7.12. Dots are used to mark the centre frequencies of the estimated components found in every frame. | 193 |
| 7.15 | Magnitude spectra of the original mixture and the estimated components for a single frame of the spectrogram in Figure 7.12 at $t = 0.1$ s. The solid, dashed and dotted lines in (b) show that the original shared peak in (a), centred at 400 Hz, has been resolved as a combination of 3 overlapping components using the optimisation-based approach. | 194 |

-
- 7.16 Optimisation-based process used to improve the frequency resolution of a short-frame spectrogram containing several real musical notes being played by a viola and a clarinet. (a) Spectrogram of the original mixture using a frame size of 1024. (b) Spectrogram of the original mixture showing the centre frequencies of the estimated components in every frame. 195
- 7.17 Magnitude spectra of the original mixture and the estimated components for a single frame of the spectrogram in Figure 7.16 at $t = 1.5$ s ($f_s = 44.1$ kHz). . . . 195

List of Tables

| | | |
|-----|--|-----|
| 4.1 | Simultaneous violin notes contained in each test mixture. | 108 |
| 4.2 | F-Scores for multipitch estimation on seven mixtures of simultaneous sustained single-note violin sources. (A) Accuracy, (P) Precision and (R) Recall. | 108 |
| 4.3 | Details of selected notes in harmonic relation. | 110 |
| 4.4 | F-Scores for multipitch estimation on 15 mixtures of simultaneous harmonically-related notes. (A) Accuracy, (P) Precision and (R) Recall. | 110 |
| 4.5 | F-Scores for multipitch estimation on several audio mixtures consisting of different numbers of harmonically-related notes played by four types of instruments. . . . | 112 |
| 4.6 | F-Scores for multipitch estimation on the Bach10 dataset. Instruments: (V) violin, (C) clarinet, (S) saxophone, and (B) bassoon. | 113 |
| 5.1 | Original and estimated components from a set of four overlapping peaks consisting of two sinusoidal tones with frequencies $f_1 = 180$ Hz, $f_2 = 210$ Hz, and different phase angles. | 131 |
| 5.2 | Performance metrics for the separation of 2-7 synchronous violin notes. | 151 |
| 5.3 | Characteristics of the selected test recordings used in Section 5.8.5. | 157 |
| 6.1 | Overview of the listening test items. | 173 |
| 6.2 | Pairwise comparison. Methods: (1) Mono, (2) ISSE, (3) IES, and (4) IES+R. . . | 176 |

List of Acronyms

- AMT** Automatic Music Transcription. 41
- ANSI** American National Standard Institute. 46
- ASA** Auditory Scene Analysis. 37
- BSS Eval** Blind Source Separation Evaluation. 83, 84, 149
- CASA** Computational Auditory Scene Analysis. 73, 74, 77
- CCWT** Continuous Complex Wavelet Transform. 15, 57–59, 89
- CNN** Convolutional Neural Networks. 79, 91, 184
- CWT** Continuous Wavelet Transform. 55, 56
- DAW** Digital Audio Workstation. 172, 185
- DNN** Deep Neural Network. 78, 91
- DWT** Discrete Wavelet Transform. 65
- EMD** Empirical Mode Decomposition. 75
- EMG** Electromyography. 40
- GUI** Graphical User Interface. 81
- HMM** Hidden Markov Models. 78, 90
- IBM** Ideal Binary Mask. 72
- ICA** Independent Component Analysis. 16, 42, 61–63, 74, 75

- IFT** Inverse Fourier Transform. 57
- ILD** Interaural Level Difference. 169
- ISA** Independent Subspace Analysis. 63, 74, 75, 80
- ISD** Itakura-Saito Divergence. 64
- ISSE** Interactive Sound Source Separation Editor. 16, 82, 157, 173, 175
- ISTFT** Inverse Short-Time Fourier Transform. 54, 140
- ITD** Interaural Time Difference. 169
- KLD** Kullback-Leibler Divergence. 64, 75
- LSD** Least Square Distance. 64
- MIDI** Musical Instrument Digital Interface. 81, 185
- MIR** Music Information Retrieval. 42
- MIREX** Music Information Retrieval Evaluation Exchange. 88
- MOS** Mean Opinion Score. 84
- MSE** Mean-Squared Error. 124, 187
- NMF** Nonnegative Matrix Factorisation. 16, 41, 63, 64, 76, 77, 80, 82, 167, 173, 177
- NSCT** Non-subsampled Contourlet Transform. 40
- ODF** Onset Detection Function. 184
- PCA** Principal Component Analysis. 75
- PEASS** Perceptual Evaluation of Audio Source Separation. 84
- PLCA** Probabilistic Latent Component Analysis. 76, 89
- PLVM** Probabilistic Latent Variable Models. 82
- SAR** Source to Artifact Ratio. 83, 149

SDR Source to Distortion Ratio. 83, 149

SIR Source to Interference Ratio. 83, 149

SiSEC Signal Separation Evaluation Campaign. 84

SNR Signal-to-Noise Ratio. 59, 72, 74

STFT Short-Time Fourier Transform. 53, 54, 59, 63, 67, 75, 76, 92, 119, 150, 186

WT Wavelet Transform. 55

WVD Wigner-Ville Distribution. 15, 55

Acknowledgement

Firstly, I wish to express my gratitude to my supervisor, Dr John E. Szymanski, who made this great adventure possible and who has constantly provided me with dedicated support, motivation and guidance during all these years at York. Moreover, his enthusiasm and positivism have proven to be indispensable to succeed in this process. I would also like to thank my thesis advisor, Mr Tony Tew, for his valuable guidance along the way.

Thank you to the Department of Electronic Engineering, University of York, especially the staff and fellow students who have also provided me with support and advice during this journey. This appreciation also extends to all friends I made during these years in Britain, in particular, to the members of the Leisure and Progression Group (L&P) at the York City Rowing Club, who certainly made my experience even more enjoyable.

I am very grateful for the support I have received from the University of Costa Rica and the Costa Rican Ministry of Science, Technology and Telecommunications, without which I would certainly not be writing this now. My gratefulness also extends to my friends and colleagues at the Campus Guanacaste and the Department of Electrical Engineering, University of Costa Rica, who have always supported me in pursuing this endeavour.

Finally, most of all, my warmest gratitude to my parents and family who have always been loving, supportive and understanding throughout the years.

Author's Declaration

I now declare that the contents of this thesis are the product of my own work, and they have not previously been presented for an award at this, or any other, University. All contributions from external sources, such as publications and websites, have been explicitly stated and referenced appropriately. Additionally, I declare that some portions of research have been previously presented at national and international conferences. These publications are listed as follows.

- A. Delgado Castro and J. E. Szymanski. “Multipitch Estimation Applied to Single-Channel Audio Source Separation: Relevant Techniques and Challenges”. *Proceedings of the 9th York Doctoral Symposium on Computer Science and Electronic Engineering*. York, 2016.
- A. Delgado Castro and J. E. Szymanski. “Improved Pitch Trajectory Estimation for Polyphonic Single-Channel Audio Mixtures”. *Proceedings of the 11th Digital Music Research Network Workshop*. London, 2016.
- A. Delgado Castro, G. Siamantas and J. E. Szymanski. “Onset Detection via Separation of Harmonic Content from Musical Notes”. *Proceedings of the 10th York Doctoral Symposium on Computer Science and Electronic Engineering*. York, 2017.
- A. Delgado Castro and J. E. Szymanski. “Semi-Automatic Mono-to-Stereo Upmixing via Separation of Note Events”. *Proceedings of the AES International Conference on Immersive and Interactive Audio*. York, 2019.
- A. Delgado Castro and J. E. Szymanski. “Semi-Supervised Audio Source Separation based on the Iterative Estimation and Extraction of Note Events”. *Proceedings of the 16th International Conference on Signal Processing and Multimedia Applications*. Prague, 2019.

Chapter 1

Introduction

1.1 Motivation

Sounds are produced when objects start vibrating. These objects can be strings, membranes, air columns enclosed by metal or wooden pipes, vocal folds, etc. For every object that is able to produce sound, the term audio source is given to it. A string quartet playing a piece of music is considered to be a group of four audio sources.

We know about sounds because we can hear them; hearing is a wonderful ability also present in many other forms of life, but it plays a crucial role in human beings. The universe is full of sounds and listening is one of the ways in which we can discover the world around us. It took eons, but evolution finally achieved a fascinating auditory system that has several marvellous features, and the capacity of isolating sounds coming from different sources is absolutely one of them.

People normally pay little attention to how wonderful this ability is, but it is critical for our survival and communication. Just imagine a couple having dinner in a fancy restaurant located in a crowded city. They are chatting whilst the waiter serves two cups of wine, they are celebrating another anniversary. In that particular venue, the cars pass close to the front door, some noise escapes from the kitchen and relaxing music can be heard across the saloon. The interesting thing is that, even with the noise of cars passing by and the interference produced by the waiter, the background music and the noise from the kitchen, the couple is still talking

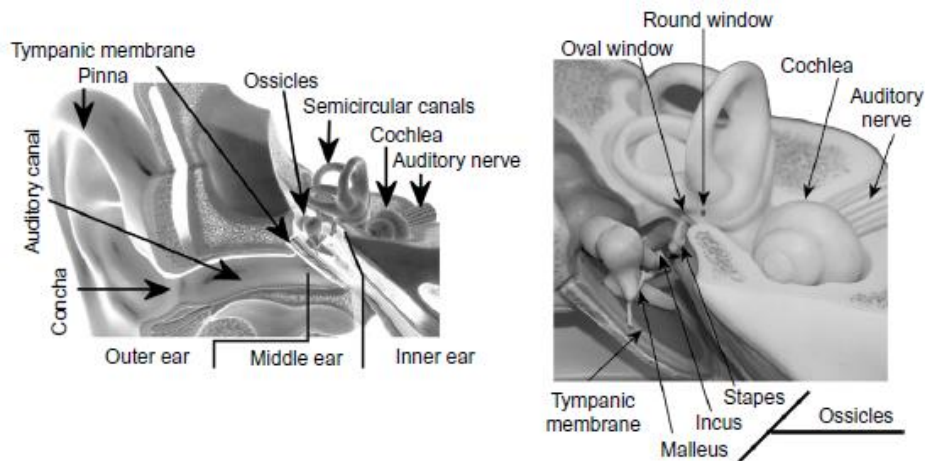


Figure 1.1: The main structures of the human auditory system showing the outer, middle and inner ears [2].

and understanding what they are saying to each other. That is because they are able to select and pay attention to just one source of sound, in this case their voices, and ignore all the rest.

Scientists have always been fascinated with many features of the human auditory system, but creating artificial systems capable of performing in the same way has proven to be rather difficult. In particular, the feature of isolating audio sources from different types of mixtures has received significant attention in recent decades [1]. The term audio source separation has been created to group all these processing techniques that share the same principal objective, which is basically, to separate the individual components of a mixture of sounds.

In audio source separation the input mixture usually comes as a digital recording and its characteristics will depend on the way it was created. When the original sounds are captured using just one sensor, i.e. one microphone, the result is a single-channel recording in which all the sounds are added together to produce the audio mixture. Separating sources in this type of input signal is the main topic of this work.

1.2 The Ability of Hearing

Hearing is the ability to perceive sounds from the environment. In human beings, it is achieved by the human auditory system, the anatomy of which is presented in Figure 1.1. It can be separated into three parts: the outer ear, the middle ear, and the inner ear [2].

The outer ear consists of the pinna (with its many valleys, ridges and depressions), the auditory canal and the eardrum. It has an acoustic effect on sounds, helping us with the localisation of sound sources and enhancing some frequencies with respect to others. Within the

eardrum, the tympanic membrane converts acoustic pressure vibrations from the outside into mechanical vibration which is transferred to the middle ear.

Three small bones (ossicles, incus and stapes) transmit the movements of the tympanic membrane to the oval window of the cochlea. The oval window forms the boundary between the middle and the inner ears. The middle ear also protects the hearing system from the effects of loud sounds.

Inside the inner ear, there is a snail-like structure known as the cochlea. Its principal function is to convert mechanical vibrations, which reach the cochlea at the oval window, into nerve firings to be processed eventually by the brain. Essentially, it is a tube coiled approximately into a spiral with about 2.75 turns. The tube is divided into three sections by Reissner's membrane and the basilar membrane, the latter being responsible for carrying out a frequency analysis of input sounds. The hair cells attached to the basilar membrane bend when it is displaced by input sounds and trigger nerve firings which are then processed by the brain.

The ability of human listeners to perceptually segregate concurrent sounds has been widely studied, and this research has inspired many computational system for sound source separation. According to some of these studies, the mixture of sounds reaching the ears is subject to an Auditory Scene Analysis (ASA), which occurs in two stages. First, the acoustic signal is decomposed into a number of sensory components. Then, in the second stage, the extracted components that are likely to have arisen from the same source are recombined into a perceptual stream, either automatically or by exploiting user interaction [3]. Although the separation approach presented in this thesis is based on this principle, the decomposition of the input mixture has been designed in such a way that the extracted components can be easily recognised as musical notes coming from a single harmonic source, which has the advantage of reducing the overall complexity of the subsequent grouping stage.

1.3 An Overview of the Proposed Solution

The main objective of the research presented here is to create a novel separation method for single-channel audio recordings, in which most of the underlying harmonic sources are estimated as the concatenation of some smaller sections called note events. The diagram in Figure 1.2 presents the main components of the proposed system, as well as its principal outcomes.

The initial single-channel recording is analysed so that important parameters can be estimated, taking into account some initial assumptions. The estimated parameters are then used

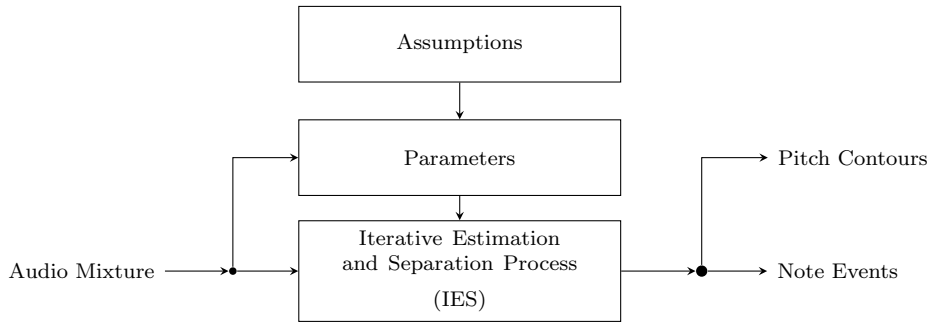


Figure 1.2: Block diagram of the proposed estimation/separation process showing its main components and the way they relate to each other.

to guide the iterative separation stage in which a number of note events are extracted from the original mixture. Each separated note event is presented as an individual track, along with its corresponding pitch contour, at the output of the system. A general description of the inner blocks is presented below.

- **Single-Channel Audio Mixture.** This is the input signal under analysis, normally, an uncompressed digital audio file with just one channel of information representing the original mixture of sound sources.
- **Assumptions.** Considering that the system does not have any previous knowledge about the input signal, several conjectures have to be established in order to continue with the separation process. Among the assumptions considered here are harmonicity of the sound sources, frame-level stationarity of the input signal, and statistical independence of the sources.
- **Parameters.** Given the aforementioned assumptions, a number of parameters are estimated from the input mixture in order to guide the separation stage. For example, the maximum number of note events to be extracted, the number of harmonic partials being separated in every frame, and the thresholds to delimit every note event.
- **Iterative Estimation and Separation.** Note events are detected and extracted from within the original mixture using an iterative method. Spectral filters and time-domain subtraction are the techniques used here to disentangle the selected sounds.
- **Note Events.** This is one of the system's outputs. It consists of the separated note events in the time domain, which can be recombined in order to reconstruct separated sources.

- **Pitch Contours.** These are the estimated variations in pitch with time of the separated note events, which provide an important insight into the musical notes being played in the original audio mixture.

1.4 Potential Applications

Decomposing an audio signal into its constituent components can be highly useful in many sectors of science and engineering. In fact, the range of possible applications goes beyond the musical and audio technology field. Important usages of source separation have been reported in wireless communications, geophysical exploration, medical signal processing, and image enhancement or recognition [4]. Some examples are presented in the upcoming paragraphs to illustrate how source separation algorithms have been applied to these fields.

1.4.1 Wireless Communications

Finding the direction of arrival in communication signals has received significant research efforts in recent years. One example study can be found in [5], where a smart antenna system is proposed to generate a strong lobe in a desired direction and nodes in undesired directions. The system allows an enhancement in security and capability in fourth generation mobile networks. A combined method using several blind source separation techniques was used for the accurate determination of weak signals. An enhancement in incoming signal angle detection was reported by using the proposed algorithm.

1.4.2 Geophysical Exploration

Seismic simultaneous source separation has recently become of interest in geophysical exploration, mainly because of its efficiency. In this field, seismic sources (explosive charges, vibrators or airguns) are used to provide acoustic energy for acquisition of seismic data. The detonation of an explosive is referred to as the seismic ‘shot’, which are normally placed below the weathered layer of the earth, improving the coupling of the seismic source to the sub-surface and avoiding problems with the very variable acoustic velocities in the weathering layer.

Simultaneous source acquisition is a common practice where blended data are usually overlapped between shot records. Thus, being able to separate the blended data and recover the single-shot seismic signals is of great importance. The study conducted by Zhou et al. [6] presented an approach based on sparse coding, which is a well-known source separation technique.

By combining patchwise dictionary learning with sparse inversion, the observed accuracy and robustness of the separated signals were enhanced, after conducting several tests using real and synthetic signals.

1.4.3 Medical Signal Processing

Electromyography (EMG) represents a way to evaluate and record muscle electrical activity. It has been used to study motor unit behaviour, while decomposition algorithms have been used to discriminate between individual motor unit action potentials from multi-unit signals. An iterative extraction schema for the sources is proposed in [7], in which the assumptions of the convolutive blind separation model are satisfied. The study also presented an approach, based on convolutive sphering of the observations, as a way to extract the sources. The obtained results showed that the proposed system provided an efficient framework for the decomposition of multi-channel invasive and non-invasive EMG signals.

1.4.4 Image Enhancement and Recognition

Blind separation of motion-blurred alias images has been studied in recent years. For example, [8] proposed a method in which the Non-subsampled Contourlet Transform (NSCT) is used as an image enhancement algorithm. The permuted alias image is firstly decomposed into low and high frequency subbands using sparse decomposition based on the proposed NSCT algorithm. A Bayesian shrinkage threshold is then used with a nonlinear gain function to enhance the resulting coefficients. Finally, the permuting image is separated by estimating correlation coefficients between the permuted alias image and its enhanced version. Positive results were reported when the method was used in several cases.

1.4.5 Audio and Music Technology

A wide variety of source separation algorithms have been developed for audio and musical signals. In 2003, Vincent et al. [9] proposed a way to classify source separation algorithms according to whether the separated signals are supposed to be listened to or not. Two basic categories were formulated and a brief description for both is presented below.

- **Audio Quality Oriented.** Applications in this category can be subdivided into two families. The first one groups applications where the main objective is to extract each individual source, while the second family refers to those applications in which the goal is

to listen to a new mixture of the sources. All these applications are expected to require extracted signals of a reasonably high quality.

- **Significance Oriented.** Applications within this category aim at retrieving features or mixing parameters to describe complex audio signals at various cognitive levels, considering different aspects of sounds.

Considering audio quality oriented applications, several examples can be mentioned. A short list of possible applications is presented below.

- **Upmixing and Remixing.** Converting single-channel recordings into multi-channel ones is the main goal of methods in this group. A single-channel to stereo conversion system was proposed by FitzGerald in 2011 [10]. By using sound source separation beforehand, the method is able to place the sources at distinct points in the stereo field, which results in more natural sounding upmixes. A stereo to stereo conversion was presented by Woodruff et al. in [11] where an informed source separation schema was designed, based on written score and spatial information, to isolate individual sound sources. The proposed system allows remixing stereo mixtures without access to the original source tracks. Finally, a stereo to multichannel conversion was introduced in [12] as an alternative to render an acoustic scene given a stereo input.
- **Restoration and Denoising.** Audio source separation has proven to be useful also in denoising and recovering historical or nostalgic material. In 2009, Fevotte et al. employed a Nonnegative Matrix Factorisation (NMF) algorithm to denoise and upmix an original piece of early jazz music. A different class of blind source separation was used in [13] to enhance speech signals in binaural hearing aids. The proposed algorithm proved to be an attractive alternative to beamforming as non *a priori* knowledge on the sensor positions was required.

Significance oriented applications, on the other hand, have focused on two major areas where active research is still being conducted. These two areas are presented below.

- **Automatic Music Transcription (AMT).** Applications in this area concentrate on automatically extracting the score of some particular music recording. Benetos et al. presented a detailed review of automatic music transcription algorithms in [14]. They actually suggested that, because of the different ways in which sources blend with each

other, applications in this area are closely related to sound source separation and, as a result, many systems operate by first isolating the signals of different instruments from the mixture and then analysing them separately. The benefit that this preprocessing stage has in the overall performance of the system was also pointed out.

- **Music Information Retrieval (MIR).** According to [15] audio signals usually contain music, speech, voices, and even background noise, so they have to be classified separately. The same study suggests that the separation of mixed signals is helpful for music retrieval, classification and segmentation. A review of current music retrieval algorithms by Casey et al. [16] indicates that source separation has made great strides in extracting information about individual musical parts from polyphonic mixtures, especially those methods based on Independent Component Analysis (ICA) and Bayesian approaches.

1.5 Contributions

The overall contribution of this thesis is a high-quality semi-supervised audio source separation system for single-channel recordings, based on the iterative estimation and extraction of note events, which are then clustered into individual sources by exploiting end-user interaction. The following is a list of specific contributions.

- An unsupervised multipitch estimation algorithm in which note events are detected and extracted from within the input mixture in an iterative fashion, which provides additional advantages in terms of note tracking, and does not require any previous training.
- A novel strategy to detect the pitch trajectory of the predominant note event in a single-channel audio mixture, based on salience measurements and time-continuity of pitch estimates.
- A strategy to detect harmonically-related note events based on the degree of correlation between the predominant pitch contour and the pitch contours of other potentially interfering events.
- An improved separation method for overlapping partials, in which two different strategies are used depending on the estimated proximity of the underlying frequency components.
- Two algorithms to extract the spectral content associated with the predominant note event in every iteration, based on time-frequency masking and time-domain subtraction.

- An interactive framework in which the end-user can listen to the extracted note events in order to decide the best way to cluster them into separated audio sources.
- A novel mono-to-stereo upmixing process based on the proposed semi-supervised audio source separation system.
- A preliminary optimisation-based approach for the separation of complex overlapping partials and the generation of time-frequency representations with improved resolution in time and frequency.

1.6 Thesis Outline

The remainder of this thesis is organised as follows:

Chapter 2 defines several important concepts in audio signal analysis and presents a review of different methods to represent audio signals in the time-frequency plane, illustrating their advantages and disadvantages. It also discusses several models that have been used successfully to decompose audio signals into a number of basis functions. Chapter 3 provides a brief review of previous audio source separation techniques and the most important strategies used in performance evaluation.

In Chapter 4, the proposed note event-based multipitch estimator is presented and evaluated. The discussion starts with a review of several previous methods, and then introduces the proposed iterative estimation-separation process, which includes detailed descriptions of its constituent parts, including the selection of the predominant note event in every iteration, and the final power-based revision of the estimated pitch trajectories.

In Chapter 5, a novel semi-supervised audio source separation system is presented and evaluated, based on the user-assisted clustering of note events. The separation of the spectral content associated with each note event is fully described, including the peak-picking strategy, the handling of overlapping harmonics, and the final extraction of the spectral energy by means of either time-frequency masking or time-domain subtraction.

Chapter 6 presents an application of the proposed audio source separation framework for the conversion of monaural recordings into stereo. The quality and naturalness of the stereo mixes are evaluated by means of a listening test, while the obtained results are compared against other similar solutions.

Finally, Chapter 7 summarises the conclusions of this work and presents potential future research directions on this matter, illustrating the use of optimisation as a powerful alternative

for the separation of complicated overlapping partials, and as a way to obtain time-frequency representations of digital signals that provide improved resolutions both in time and frequency.

Chapter 2

Audio Signals: Representation and Modelling

2.1 Preamble

Audio signals are the main subject of analysis in this research. Hence, the nature of sound and its principal characteristics have to be presented in advance, along with important concepts from music theory, before the proposed separation process can be introduced. Section 2.2 to 2.4 contain a comprehensive review of this material.

Since audio sources are highly dynamic structures, it might be useful to find an alternative representation in which the main components of these structures are more easily identified and separated. Therefore, Section 2.5 presents a selection of commonly used time-frequency representations for audio signals, while Section 2.6 deals with different methods to decompose them into a set of basic elements. These techniques have been successfully used in previous separation approaches, which makes it possible to include advantages and limitations in some of the cases.

2.2 Musical Theory

Music is made up sounds and those sounds are normally grouped to form elaborated structures such as melodies and accompaniment [17]. Throughout history, human beings have developed several ways to write down music on paper, and therefore, preserve and share musical compositions. Scores are the most commonly used method to codify music [18]; it is basically a symbolic notation that works in a similar way as written languages. Within this musical code, however, there are several aspects that have to be defined.

2.2.1 Terms and Definitions

The main elements of music notation are briefly described in this section, in order to establish a framework that will be used in the rest of this report.

- **Event.** This corresponds to the basic unit in music, and the most common event is probably the occurrence of a note.
- **Pitch.** Frequency is a physical measure of vibrations per second, whilst pitch is the corresponding perceptual experience of frequency [18]. The American National Standard Institute (ANSI) defined it as “that auditory attribute of sound according to which sounds can be ordered on a scale from low to high” [19]. It is important to establish that pitch is limited to sounds within the range of human hearing, while frequency is not. The pitch of any harmonic sound is closely related to its fundamental frequency (f_0), which is a concept that will be discussed further in upcoming sections of this report.
- **Timbre.** The concept of timbre is widely used and has a long tradition. However, its meaning is fuzzy and encompasses an enormous variety of phenomena [20]. In music scores, timbre usually means the type of instrument to be played, but it can also be used to describe an instrument’s sound quality as sharp, dull, shrill and so forth [18], or to describe any particular playing style.
- **Loudness.** This refers to the intensity of any particular sound. The range from the softest to loudest sound for an instrument is commonly known as dynamic range. Loudness depends upon a number of perceptual and acoustical factors and is not easy to characterise in general terms [18].

- **Duration.** The duration of any sound is basically the time it lasts. In music, the beat is used as the fundamental unit of time measurement and it corresponds to the pulse of the music, while tempo refers to the number of beats per minute. The onset is the stipulated moment for a sound to begin, counted in beats from the beginning of the score. The onset time is the same moment counted in seconds from the beginning of the recording [18].

2.2.2 Tones and Notes

Helmholtz gives a straightforward description of how musical tones are perceived by human beings. According to [21], “*a musical tone strikes the ear as a perfectly undisturbed, uniform sound which remains unaltered as long as it exists, and it presents no alternation of various kinds of constituents*”. The conclusion is, therefore, that musical tones are the simpler and more regular elements of the sensations of hearing.

The sensation of a musical tone in human beings is produced when the ear is excited by a regular motion of the air, which is generated by an equally regular motion of the sonorous body. Those regular motions can be oscillations, vibrations, or swings, and it is necessary for these motions to be regularly periodic [21].

In western common notation, notes are characterised by three different sonic quantities: pitch, loudness and timbre. If onset and duration are also assigned, the result is a note [18]. Notes combined in temporal order form a score, which provides all the necessary information to correctly interpret and perform those notes. Furthermore, when notes are performed in sequence, the result is a melody, whilst notes performed simultaneously are called harmony.

2.2.3 Cross-section of a Note

Musical notes can be divided into parts according to their temporal evolution. These four parts are: attack, decay, sustain, and release. Figure 2.1 shows the waveform of a violin playing the note A3, in which vertical lines were used to separate its parts. A detailed explanation of each part is presented below following [22].

- **Attack.** This is defined as the initial section of the note, from the start point to the instant where the note reaches its maximum amplitude.
- **Decay.** In this section, the amplitude of the note starts to decrease. The decay ends when the amplitude of the note reaches an almost steady state.

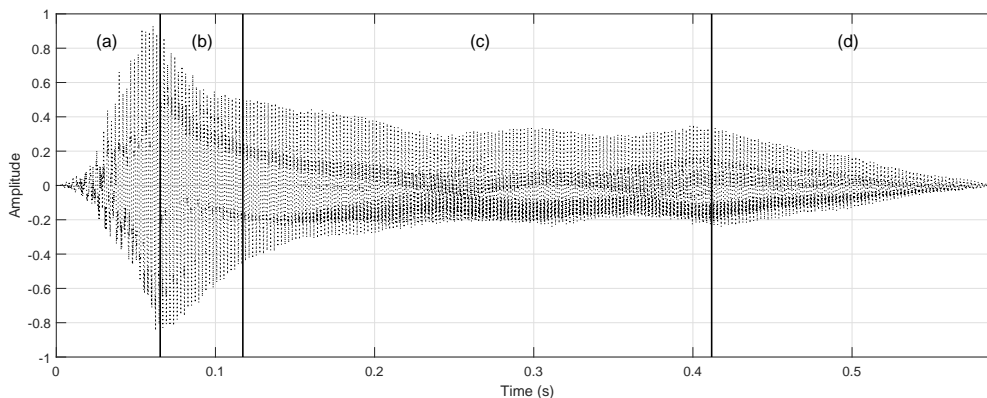


Figure 2.1: Waveform of a violin playing the note A3, and its corresponding sections. (a) Attack, (b) Decay, (c) Sustain, and (d) Release.

- **Sustain.** During this stage, the vibration of the instrument has settled to some stable level where the amplitude and frequency of the note do not change significantly.
- **Release.** At this point the supply of energy to the vibrating object stops and the amplitude of the vibration starts to decrease once again. The release ends when the oscillation disappears.

Attacks are also associated with transients, which have very rich spectral content, almost noise-like [23]. During the sustain, which is the steady-state portion of the note, a clear structure of partials dominates and more information can be estimated, especially the pitch of the note.

2.2.4 Partial, Harmonics and Overtones

Musical notes can be constructed from a collection of sinusoids. Each individual sinusoid that collectively forms the original note is called a partial, because each carries a partial characterisation of the whole sound. Within the literature, the term components is also used to refer to partials. Each partial is created by a specific part of the vibrating system of the instrument, and they can be observed in the magnitude spectrum of the signal. The principal properties of partials are their magnitude and the frequency at which they are located [18].

If the partials of some particular sound are located at frequencies that are positive integer multiples of a basic frequency, those partials are called harmonics. The greatest common divisor in a series of harmonics is called the fundamental frequency, denoted by f_0 , and corresponds to the lowest partial of the note. The remaining partials are higher in frequency and the term overtones is used to refer to them [18].

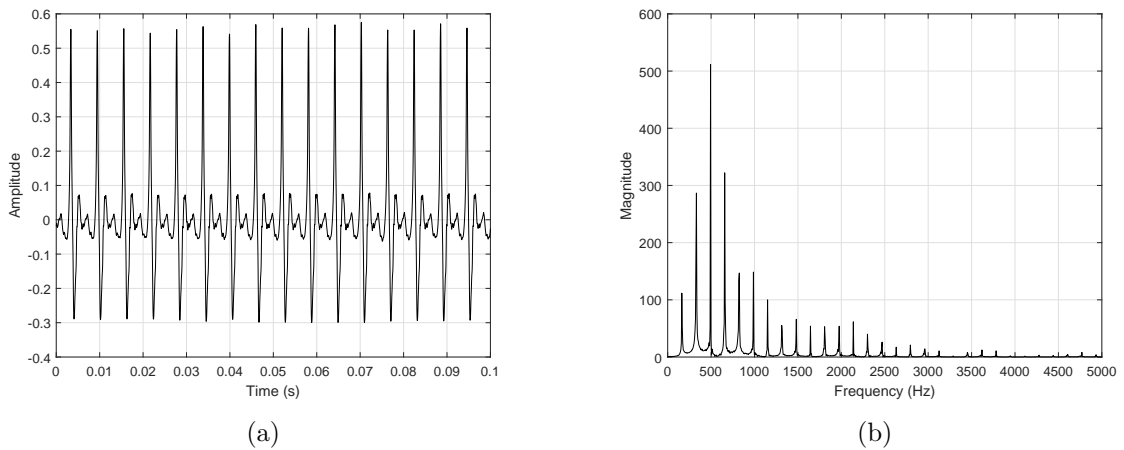


Figure 2.2: A horn playing the note E3. (a) Waveform. (b) Magnitude spectrum.

2.3 Classification of Sounds

Sounds can be classified according to several criteria. The level of periodicity and differences in timbre are characteristics that can be used to distinguish different sounds. Two types of sounds are particularly important in this research, and they are discussed in the following sections.

2.3.1 Harmonic Sounds

Harmonic sounds normally present a characteristic structure in frequency where all the components are approximately regularly spaced with respect to its fundamental frequency, which is closely related to the perceived pitch of the sound.

The representation of a harmonic sound can be seen in Figure 2.2, both in the time domain and the frequency domain. The waveform corresponds to a horn playing the note E3 and the magnitude spectrum on the right clearly depicts the set of harmonically-related partials that form the sound.

It has been demonstrated that every musical instruments exhibits varying degrees of non-linearity in their excitation and feedback system [24]. Therefore, a perfect harmonic series of partials cannot be observed in the frequency domain. In some cases, such as pianos, a deviation between partials and ideal harmonics is to be expected due to the stiffness and linear density of their strings. This deviation is frequency dependent and increases with frequency [25].

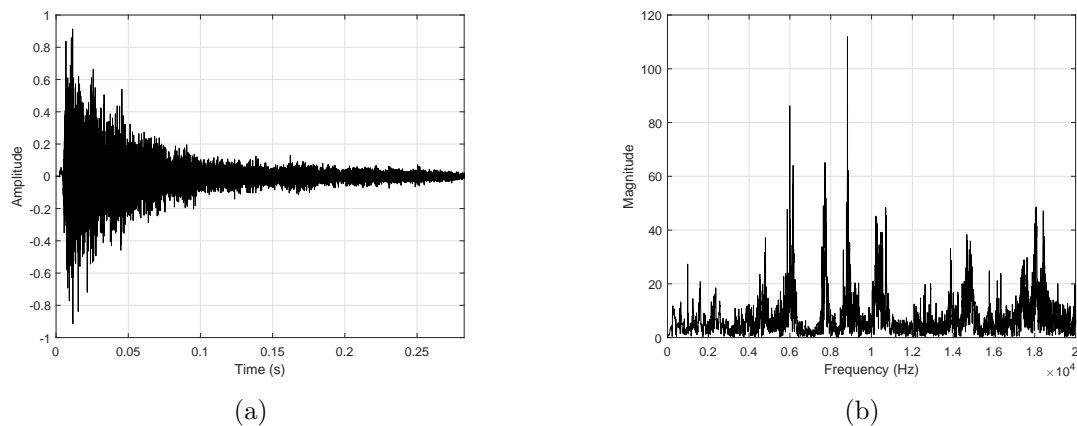


Figure 2.3: A sound produced by a tambourine. (a) Waveform. (b) Magnitude spectrum.

2.3.2 Inharmonic Sounds

For some other sounds, for example most drums and bells, the observed partials in the magnitude spectrum are not regularly spaced and therefore a fundamental cannot be identified. Inharmonicity comes from the multidimensional nature of this type of vibrating object. In the case of drums, the vibrating object is a membrane, which is two-dimensional, while bells are three-dimensional. Hence, the solution of the wave equation in these cases does not result in a harmonic model.

Figure 2.3 shows the time-domain waveform and the magnitude spectrum of a sound produced by a tambourine. The spectrum on the right presents a non-structured set of energy peaks surrounded by a significant level of noise, which is the normal case for many non-structured sounds.

2.3.3 Monophonic and Polyphonic Sounds

When just one source of sound is present in a piece of music, be this source either instrumental or vocal, the term monophonic is applied to it. On the other hand, polyphonic music refers to those pieces in which two or more sound sources are active simultaneously [17]. There are other instruments in which multiple notes can be played at the same time, such as piano and guitar, in which case a single source can also be a polyphonic one.

The complexity of the resulting sound increases as its polyphony grows. If harmonic partials associated with different sources have very similar centre frequencies, the term overlapping partials is used to refer to them. A highly polyphonic sound is more difficult to analyse, in part because the probability of observing a larger number of overlapping partials is higher. As an

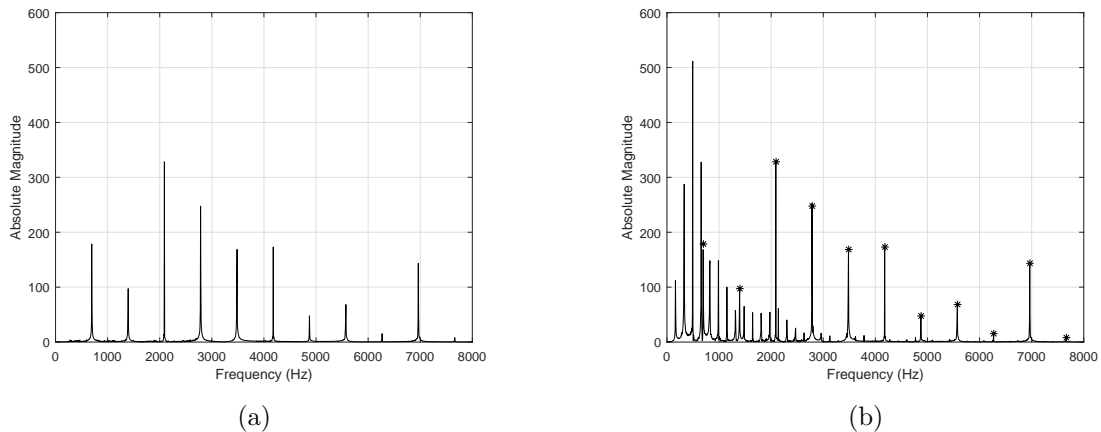


Figure 2.4: Different sounds in the frequency domain. (a) Monophonic signal in which a violin is playing the note F4, (b) Polyphonic signal comprising a violin and a horn. Harmonic peaks of the violin are marked with asterisks.

example, the magnitude spectrum of a monophonic sound is compared with that of a polyphonic signal in Figure 2.4. These examples were made from a violin playing the note F4 and a horn playing the note E2. The magnitude spectrum on the right has the harmonics of the violin marked with asterisks.

2.4 Models for Mixing Processes

It is usual for audio signals originating from different sources to coexist simultaneously in the form of a mixture. Two main entities are involved in any kind of signal mixing or separation process: the original source signals and the mixture channels [23].

Audio recordings are normally produced in recording studios, where the sounds coming from different sources are captured by one or more microphones, which in this case are also called sensors. The output of each sensor corresponds to a single mixture channel, which is basically a single observation of the sound mixture. In multi-channel recordings, multiple observations of the mixture are available, whilst in single-channel recordings only one observation can be analysed.

2.4.1 Multi-Channel Models

The most general way to express the mixing process between the source signals $s_r(n)$ and the mixture channels $x_c(n)$ can be formulated mathematically as the following equation.

$$x_c(n) = \sum_{r=1}^R \sum_{v=-\infty}^{+\infty} a_{r,c}(v) s_r(n-v) \quad \forall c = 1, 2, \dots, C \quad (2.1)$$

The coefficients $a_{r,c}(v)$ represent a time-varying filtering process between the r -th source and the c -th channel, v denotes delay in samples, R is the total number of sources, and C denotes the total number of channels. Mixtures modelled in this way are called convolutive, where sources and microphones are moving in a reverberant space [23].

Assuming an anechoically-recorded mixture and ignoring the delays, equation 2.1 can be further simplified.

$$x_c(n) = \sum_{r=1}^R a_{r,c} s_r(n) \quad \forall c = 1, 2, \dots, C \quad (2.2)$$

Mixtures characterised by this simplified model are called instantaneous, and they can be described as a system of linear equations. The assumption of linearity in the mixing model helps to simplify the problem and reduces its indeterminacy [23].

2.4.2 Single-Channel Instantaneous Model

The case of extreme indeterminacy occurs when just one mixture channel is available, i.e. $C = 1$. In this case, the mixing model can be expressed as follows.

$$x(n) = \sum_{r=1}^R a_r s_r(n) \quad (2.3)$$

The problem of single-channel audio source separation in instantaneous mixtures is, thus, the problem of estimating $s_r(n)$ and a_r , $\forall r = 1, 2, \dots, R$, when the only known quantity is the mixture $x(n)$.

2.4.3 Possible Scenarios

Sound source separation problems can be classified by the number of sources and sensors. Basic characteristics for each category are presented below.

- **Over-determined Case.** The number of sensors is greater than the number of sources, i.e. $R < C$.
- **Determined Case.** The number of sensors is equal to the number of sources, i.e. $R = C$.
- **Under-determined Case.** The number of sensors is less than the number of sources, i.e. $R > C$.

The single-channel source separation problem is the extreme case of under-determined source separation [26], and it constitutes the problem that will be addressed in the present research.

2.5 Representation of Audio Signals

Audio signals are normally non-stationary, meaning that their features change over time, especially their characteristics in frequency. To deal with this particular behaviour of audio signals, they are nearly always analysed in short segments, called frames, rather than using the whole signal. There are many reasons for this, but probably the most important is that the human auditory system analyses only a short segment of audio signals at a time [27]. Hence, using a time-frequency representation to perform spectral analysis might be closer to human hearing.

Another aspect that must be considered before choosing a framework to represent audio signals, is whether it is possible to resynthesise the original signal from the transformed representation, and the artifacts that might be introduced during the transformation process [28]. For many audio applications, perfect reconstruction is particularly important, so that the signals can be analysed and synthesised without inserting significant levels of distortion.

As the main objective of the present research is to separate musical structures arising from different sources in a single-channel recording, it might be important to use a representation in which those structures are evident and more easily separable. In the following subsections several time-frequency representations, commonly used in previous systems, are introduced.

2.5.1 Short-Time Fourier Transform

The Short-Time Fourier Transform (STFT) is a powerful general-purpose tool in audio processing, which assumes the signal is stationary over a sufficiently short period of time. With a suitably chosen frame length, and using an appropriate analysis window function, the STFT measures the local time and frequency evolution of the signal over a section where it is assumed to be quasi-stationary. Then perfect reconstruction can be obtained from this transformed domain by using an overlap-add technique [28].

When the STFT is applied to a finite discrete signal, it can be thought of as a process that transforms a time-domain vector into a complex time-frequency domain matrix. Once transformed, the time-frequency signal can be analysed, visualised, further processed, and transformed back into the time-domain [29]. An intensity plot of the STFT magnitude, usually on a logarithmic scale such as decibels (dB), is called a spectrogram. Figure 2.5 shows a classic spectrogram of a speech signal corresponding to a male voice saying the words *brilliant, arresting, extravagant*.

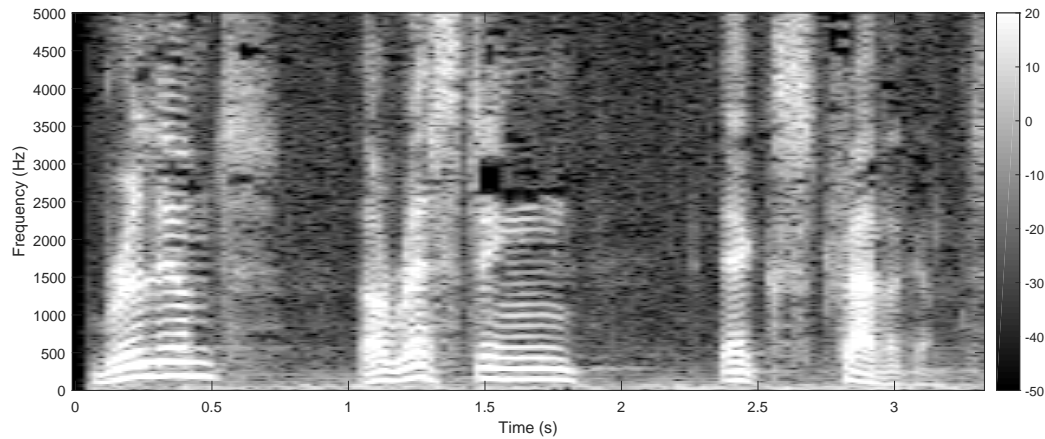


Figure 2.5: Hanning-windowed spectrogram of a speech signal consisting of a male voice pronouncing the words *brilliant*, *arresting*, *extravagant*. The frame size is 2048 samples, with 50% overlap, while the sampling frequency is 44.1 kHz.

The STFT can be mathematically defined by the following equation.

$$STFT_x^h(k, m) = X(k, m) = \sum_{n=0}^{N-1} x(n + mL)h(n)e^{-j2\pi k(n+mL)/N} \quad (2.4)$$

where $X(k, m)$ is the complex value of the k -th frequency coefficient at the m -th time frame, $x(n)$ is the input signal at time n , N is the frame size, L is the hop size, and $h(n)$ is the window function. The time-frequency representation is controlled by the variables k and m , where $k = 0, 1, 2, \dots, K - 1$ defines a particular frequency bin, from 0 Hz up to its maximum value (Nyquist Frequency), and $m = 1, 2, 3, \dots, M$ is the frame number covering the whole duration of the signal. The total number of frames is denoted by M .

To transform back into the time domain, the Inverse Short-Time Fourier Transform (ISTFT) is used. The ISTFT is defined by the following equation.

$$x(n) = \frac{1}{N} \sum_{m=1}^M \sum_{k=0}^{K-1} X(k, m)e^{j2\pi k(n+mL)/N} \quad (2.5)$$

The negative effect of using a window function is that the spectral peaks appear broadened. Instead of having the shape of a delta function placed at a single frequency, they end up having the shape of the Fourier transform of the window function [23].

It has to be mentioned that the frame length has an important effect on this representation. A long frame increases the resolution in frequency, but the localisation in time worsens, so any information about rapid temporal signal changes is averaged. A shorter frame, on the other hand, produces the opposite situation, good localisation in time and low frequency resolution.

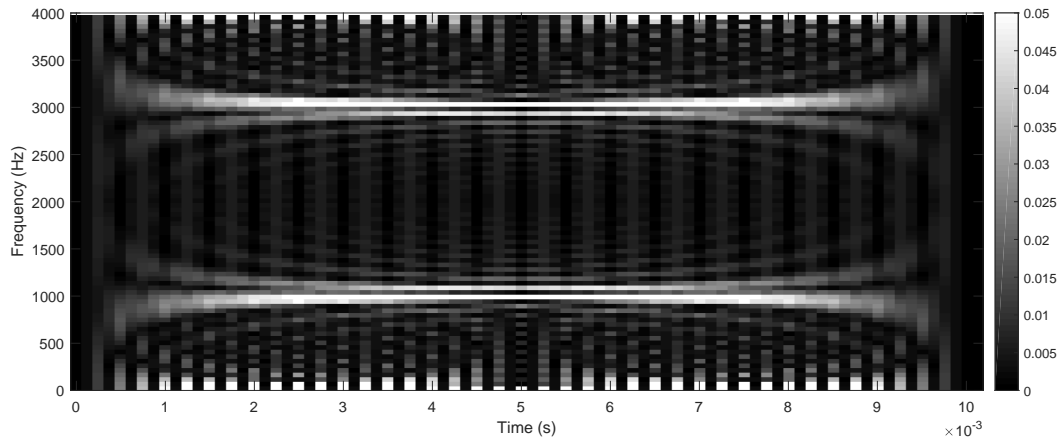


Figure 2.6: Wigner-Ville Distribution (WVD) of a pure tone with frequency of 1 kHz.

2.5.2 Wigner-Ville Distribution

The Wigner-Ville Distribution (WVD) appeared for the first time in 1932, when Ville presented an adaptation of the framework developed by Wigner in the field of quantum thermodynamics [22]. The WVD of a signal $x(t)$ can be expressed by the following equation.

$$WVD_x(k, \phi) = \int_{-\infty}^{+\infty} x\left(k + \frac{\tau}{2}\right) x^*\left(k - \frac{\tau}{2}\right) e^{-j\phi\tau} d\tau \quad (2.6)$$

By observing the definition in Equation 2.6, we can conclude that the WVD is essentially the autocorrelation of the signal under analysis. The operation indicated by (*) is the complex conjugate of the function. An example is presented in Figure 2.6, where the WVD is applied to a single tone with frequency of 1 KHz. Spurious terms can also be identified in this time-frequency representation, which is an issue that limits the range of applications that could really benefit from this type of framework.

2.5.3 Wavelet Transform

During the 1980's, Grossmann, Morlet and other researchers introduced the Wavelet Transform (WT), in which the transformation is achieved by taking the convolution between the input signal and several versions of the same basic function [22]. The basic function used is called a Mother Wavelet and there are several families of commonly used functions having different features and possible application areas. Figure 2.7 shows four different wavelet functions.

The Continuous Wavelet Transform (CWT) represents the input signal as the sum of time-translated dilations and contractions of the mother wavelet, represented by $\psi(t)$ and defined by the following relation [28].

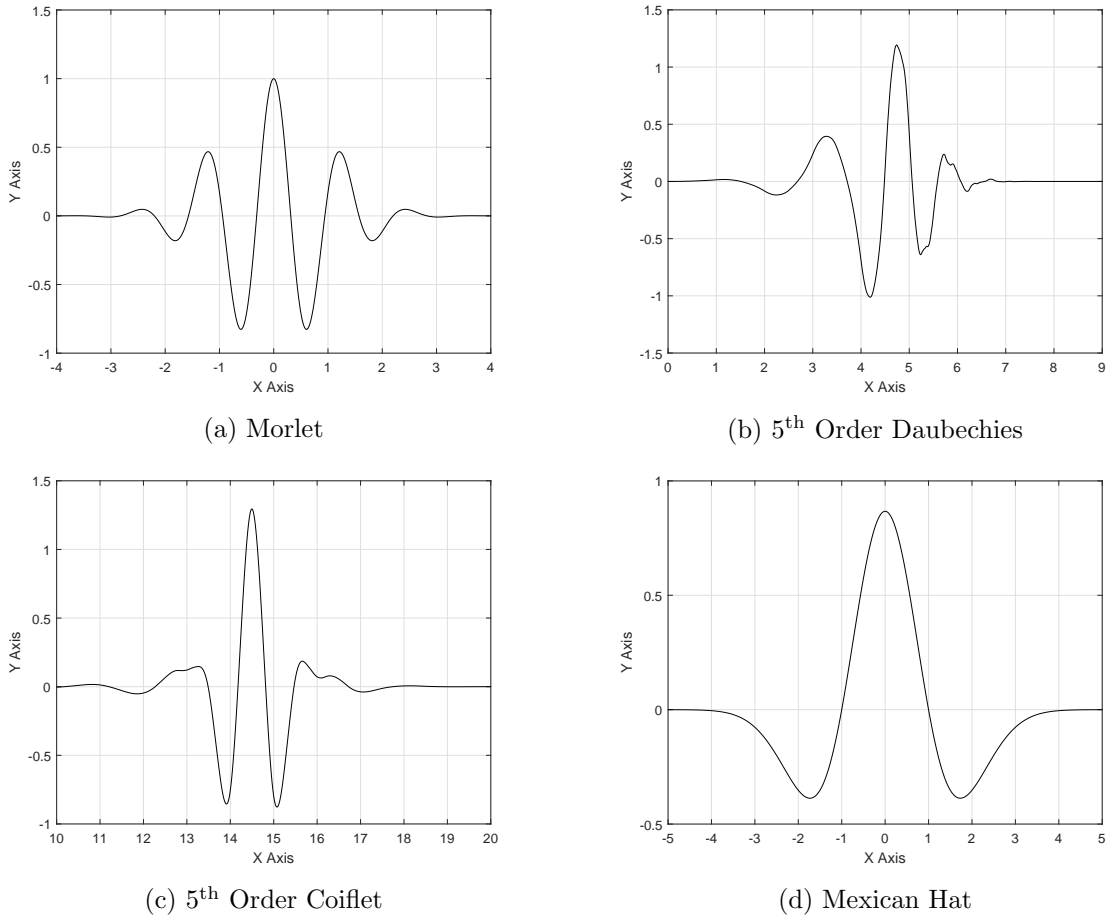


Figure 2.7: Four different wavelet functions.

$$\psi_{\tau,a}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-\tau}{a}\right) \quad (2.7)$$

In Equation 2.7, τ controls the translation in time and a is the scale factor, which is closely related to frequency. The CWT is defined by the following equation.

$$W(\tau,a) = \int_{-\infty}^{+\infty} x(t)\psi_{\tau,a}^*(t)dt \quad (2.8)$$

The wavelet coefficients $W(\tau,a)$ measure the similarity between the input signal $x(t)$ and the basis function $\psi_{\tau,a}(t)$. To reconstruct a time-domain function from the wavelet coefficients, an inverse transformation can be used. It is defined as follows.

$$x(t) = C_{\varphi}^{-1} \int_{-\infty}^{+\infty} \int_0^{+\infty} W(\tau,a)\psi_{\tau,a}\frac{da}{a^2}d\tau \quad (2.9)$$

The factor C_{φ}^{-1} represents the admissibility criterion. If the CWT is applied to the input signal in a frame-based schema, the result is a time-frequency representation called scalogram.

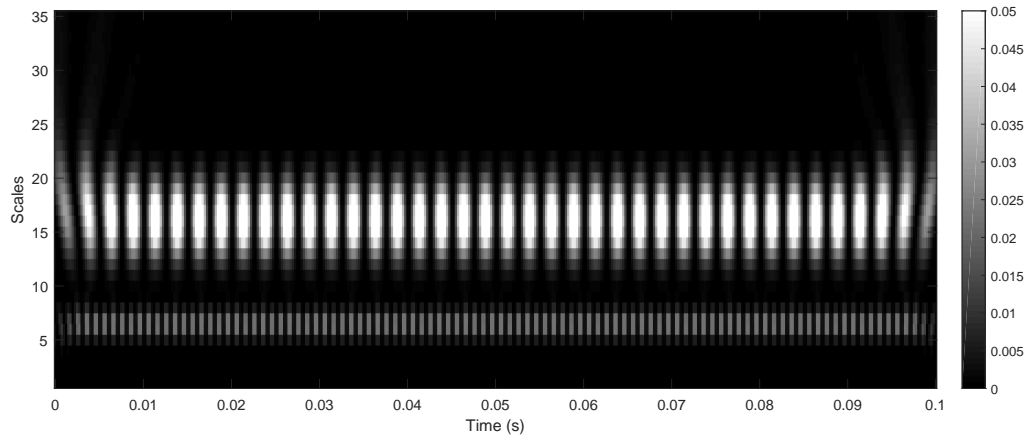


Figure 2.8: Scalogram of the sum of two pure tones with frequencies 200 Hz and 500 Hz. The sampling frequency was set to 4 kHz.

In Figure 2.8 the scalogram of the sum of two pure tones can be observed. The frequencies of the tones are 200 Hz and 500 Hz, while the sampling frequency used was 4 kHz.

2.5.4 Continuous Complex Wavelet Transform

Another approach to the scalogram was proposed by Ponce de León in [22], and the name Continuous Complex Wavelet Transform (CCWT) was given to it. This time-frequency representation is obtained by using a complex mother wavelet to separate the magnitude and phase information of the input signal.

Some interesting relations between the CCWT and the Fourier transform were observed and according to [22], it is possible to calculate the wavelet coefficients by taking the Inverse Fourier Transform (IFT) of the result obtained when the Fourier transform of the input signal is multiplied by the Fourier transform of every time-translated and scaled version of the mother wavelet.

In this case, the Fourier transform of every time-translated and scaled version of the original mother wavelet constitute a filter bank that separates different frequency content of the input signal. This type of analysis is normally referred as a multi-resolution analysis, because the signal is decomposed using a different resolution in every frequency band.

An example is presented in Figure 2.9 where a speech signal, consisting of a male voice pronouncing the words *brilliant*, *arresting*, *extravagant*, is represented using this technique.

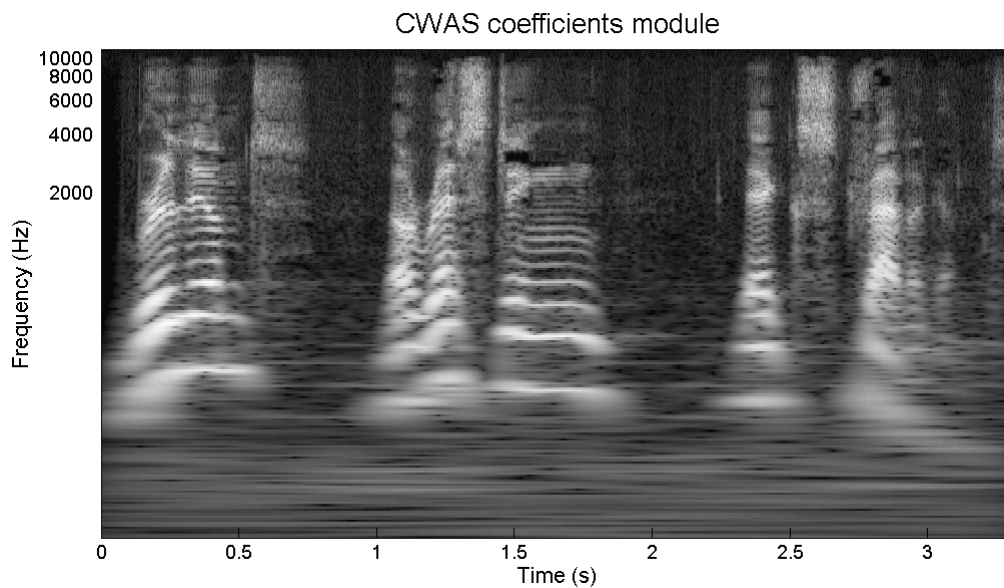


Figure 2.9: Continuous Complex Wavelet Transform (CCWT) of the speech utterance *brilliant, arresting, extravagant*. Note the logarithmic y-axis. Image generated and provided on request by Jesús Ponce de León Vázquez (jponce@unizar.es) in September 2016.

2.5.5 Cochleagram

The cochlea is an organ within the inner ear and its main objective is to convert sound into neural spikes. Sound produces mechanical impulses at the outer ear, which are then transmitted across the middle ear to produce vibrations on the oval window, which is a flexible membrane, and its motion sets the fluid within the cochlea in motion. Inside the cochlea, this motion is transmitted to the basilar membrane and the final transducing medium is the collection of hair cells sitting atop the basilar membrane that implement the transformation to the neural spikes within auditory nerve bundles [31].

Several models have been proposed to describe the way in which sound is represented by the human auditory system. Most of them give an interpretation of the auditory system as a filter bank, where Gammatone filters are normally used to model the cochlea. Gammatone filters are approximately logarithmically spaced, with constant quality factor (Q), for frequencies between $f_s/10$ and $f_s/2$, and approximately linearly spaced for frequencies below $f_s/10$, where f_s is the sampling frequency in Hz. Hence, this characteristic results in selective non-uniform resolution in the time-frequency representation of the analysed audio signal [4].

Like some of the previous methods, the cochleagram based on Gammatone filter banks also has a non-uniform time-frequency resolution, but it is more balanced between high and low frequency areas in comparison with a constant- Q representation [4]. Figure 2.10 shows the

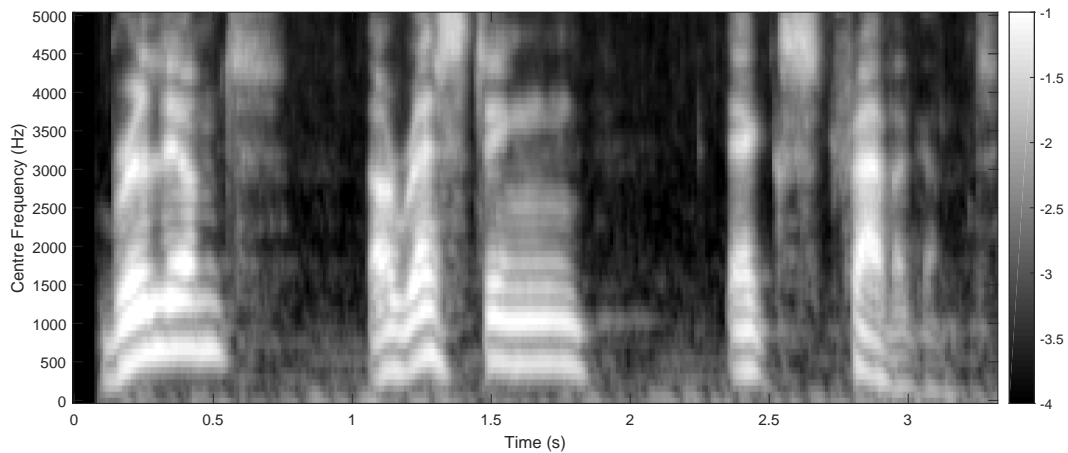


Figure 2.10: Cochleagram of the same speech excerpt presented in Figures 2.5 and 2.9. This graph was generated using the code by Ning Ma [30].

cochleagram of the same utterance excerpt presented previously as a spectrogram in Figure 2.5 and as a CCWT in Figure 2.9.

2.5.6 Time-Frequency Reassignment

In the analysis of audio signals, reassignment techniques have been used to sharpen other conventional time-frequency representations, in order to improve their readability by ensuring an optimal window alignment. Reassignment methods generate sharpened time-frequency estimates for each spectral component from partial derivatives of the short-time phase spectrum. Then, instead of locating these components at the geometrical centre of the analysis window, as in traditional spectral analysis, they are reassigned to the centre of gravity of their complex spectral energy distribution, also computed from the short-time phase spectrum following the principle of stationary phase. This method was initially applied to conventional spectrograms, but nowadays it is being applied to other types of time–frequency and time-scale transforms [32]. A comparison between the conventional STFT and its reassigned counterpart is presented in Figure 2.11 for a swept-frequency cosine function.

Although reassigning a spectrogram has been shown to produce sharply localised distributions, it should not be seen as a super-resolution process, given that a beating effect is likely to occur when more than one component is observed within the smoothing window, which results in interference fringes that prevent the correct identification of the underlying components [33].

Since reassignment methods localise both signal components of interest and noise at the same time, the discrimination between the two is very difficult when the Signal-to-Noise Ratio (SNR)

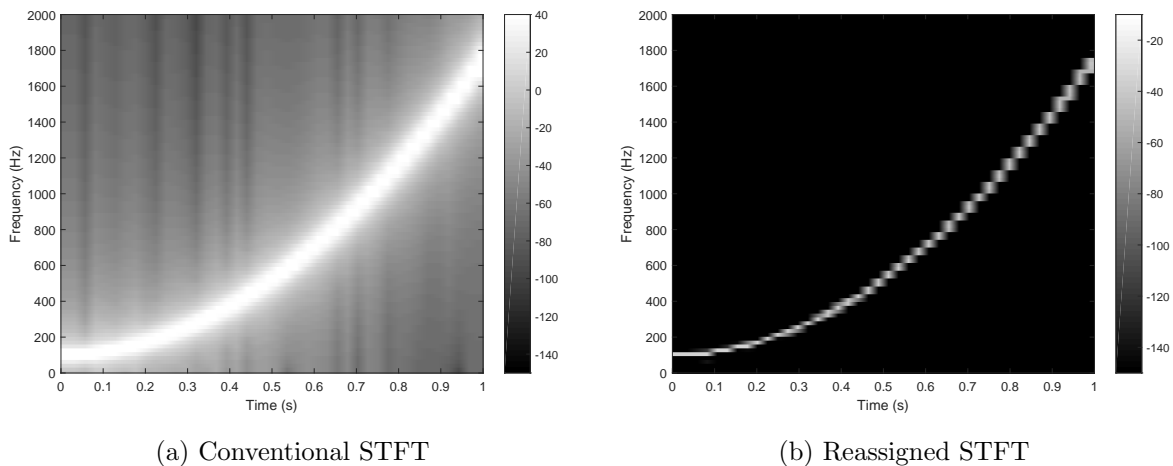


Figure 2.11: Time-Frequency reassignment using a frequency-modulated sinusoid as an input signal.

is very low. An alternative reassignment process was proposed by Ahrabian and Mandic in [34], in order to reassign oscillatory components of interest while suppressing noisy components, in which a variant of the retrieval of components was used.

2.6 Signal Decomposition Models

The time-frequency representation of an audio signal can be decomposed into a linear combination of some special functions that are known as spectral basis functions [17]. Several models have been presented in order to obtain those basis functions and therefore, decompose the original audio signal into its basic components. A selection of relevant models are explained in the following sections.

2.6.1 Sinusoidal Modelling

This is probably the most widely used signal model in speech and music processing. Any acoustic source that has resonance frequencies or vibrational modes, or any synthetic source containing a deterministic component is a good candidate to be decomposed using this framework. The schema assumes that any source can be modelled as an infinite sum of sinusoids with coefficients given by its Fourier transform, but for practical reasons, the number of sinusoids is limited to a maximum value. The decomposition of a time-domain signal $x(t)$ can be expressed by the following equation [28].

$$x(t) = \sum_{h=1}^H a_h(t) \cos(\phi_h(t)) = \sum_{h=1}^H \frac{a_h(t)}{2} \left[e^{j\phi_h(t)} + e^{-j\phi_h(t)} \right] \quad (2.10)$$

The deterministic components of a sound are then modelled as a sum of H sinusoids with time-varying amplitudes $a_h(t)$ and phases $\phi_h(t)$.

2.6.2 Independent Component Analysis

Independent Component Analysis (ICA) is a well-known transformation method in which the goal is to find a linear representation of non-Gaussian data so that the components are as statistically independent as possible. Such a representation might be able to capture the essential structure of the data in many applications, including feature extraction and signal separation [35].

The required non-Gaussianity of the input data does not limit the application of the algorithm. Moreover, there is a wide range of real signals, e.g. music, human speech, and electrical signals from different brain areas, that can be considered as not normally distributed data [36].

Assuming the existence of R independent sources $s_1(t), \dots, s_R(t)$ and the observation of as many mixture signals $x_1(t), \dots, x_R(t)$, these mixtures being linear and instantaneous, a representation of the mixing process is given by the following equation.

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \quad (2.11)$$

where $\mathbf{s}(t) = [s_1(t), \dots, s_R(t)]^T$ is an $R \times 1$ column vector collecting the source signals, vector $\mathbf{x}(t)$ collects the R observed signals, and the square $R \times R$ mixing matrix \mathbf{A} contains the mixture coefficients. The separation of the sources is achieved by computing an $R \times R$ separating matrix \mathbf{W} , also called de-mixing matrix, whose output can be expressed as follows [37].

$$\hat{\mathbf{s}}(t) = \mathbf{W}\mathbf{x}(t) \quad (2.12)$$

The vector $\hat{\mathbf{s}}(t)$ is an estimate of the original vector $\mathbf{s}(t)$ of source signals, and the goal of ICA is to find the de-mixing matrix \mathbf{W} (i.e. the inverse of \mathbf{A}) that will give $\hat{\mathbf{s}}(t)$ the best possible approximation of $\mathbf{s}(t)$ [36].

The basic extraction of independent components is illustrated in Figure 2.12. We start with two different input signals, presented in Figure 2.12(a), which are then linearly combined to create two different mixtures, shown in Figure 2.12(b). A scatter plot of the mixtures is presented in Figure 2.12(c). The data is whitened, by first subtracting the mean value and then decorrelating all samples, to produce the scatter plot in Figure 2.12(d). Then, ICA starts modifying the data in order to maximise the non-Gaussianity of their probability density functions. The

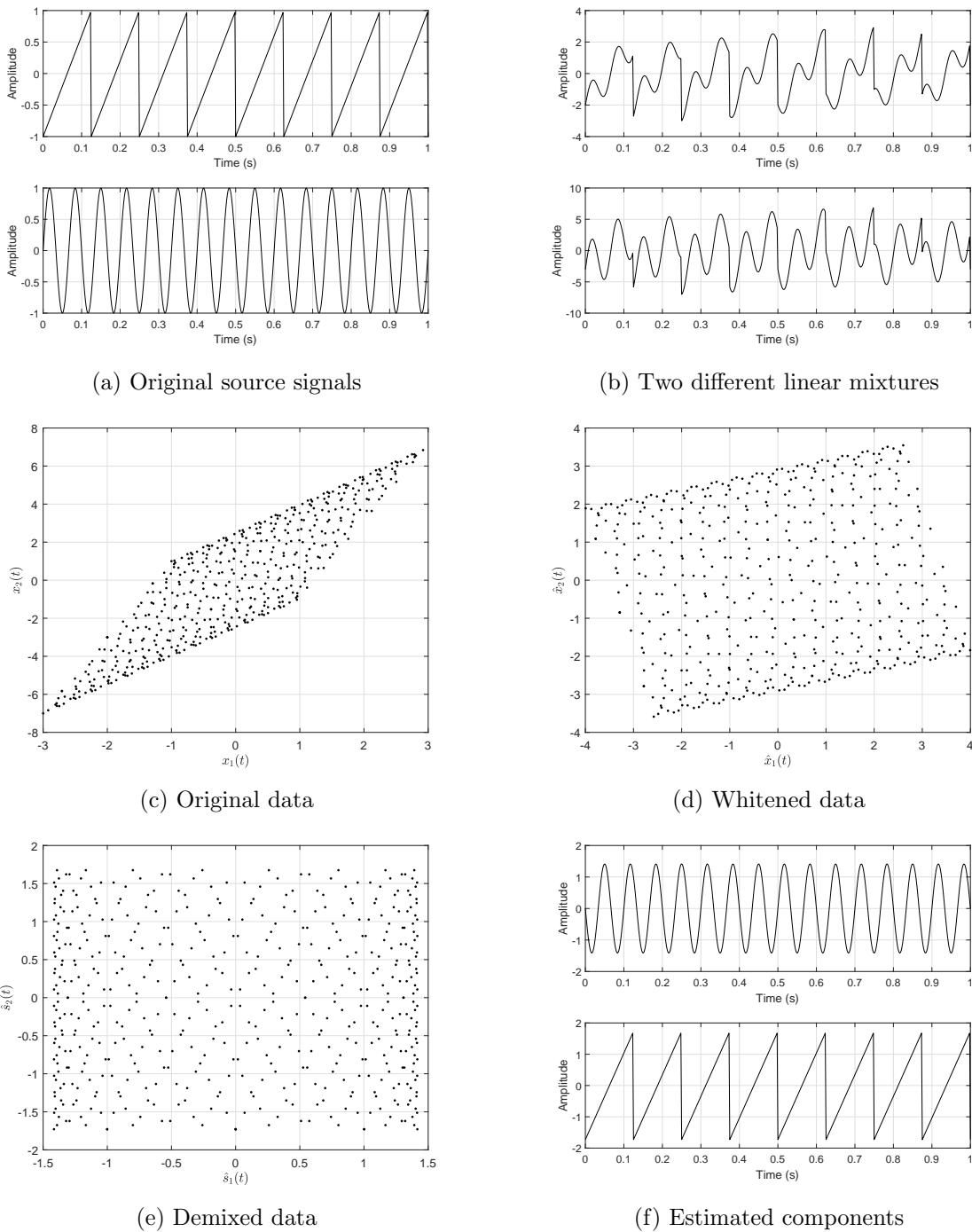


Figure 2.12: Decomposition process using ICA and two synthetic input signals.

scatter plot in Figure 2.12(e) shows the results. Finally, the estimated independent components are presented in Figure 2.12(f).

The example above reveals some important problems of ICA in which the estimated components can be arbitrarily scaled and randomly ordered. These are known as the scaling and

permutation ambiguities, respectively. The FastICA algorithm was presented by Hyvärinen et al. as a quick way to extract independent components from mixed data [35].

2.6.3 Independent Subspace Analysis

Although ICA has proven useful in the field of source separation, it suffers from one significant drawback since it works on the assumption that there are at least as many observations of the mixture as independent components, and in many practical situations this assumption is invalid [38].

Independent Subspace Analysis (ISA) was proposed by Casey and Westner in 2000 to separate individual audio sources from single-channel recordings [39]. It is based on ICA but the algorithm was extended in several ways. ISA identifies independent multi-component source subspaces of an input vector and then uses dynamic independent components to represent non-stationary signals.

The basic consideration of the ISA method is to decompose the time-frequency space, normally the STFT of a mixed signal, as a sum of independent source subspaces [40]. A significant limitation of ISA has been observed when the original source signals have very similar probability distributions, which may prove difficult or impossible to separate [38].

The ISA decomposition occurs when each frame of the input spectrogram, at time τ , is expressed as a weighted sum of ρ independent basis vectors denoted by \mathbf{z}_i , for $i = 1, 2, \dots, \rho$. These basis vectors are themselves spectral slices that show interesting features of the spectrum, and they are defined to be static, but each one is weighted by a time-varying scalar coefficient y_i^τ . The weighted sum of ρ basis vectors reconstructs a spectrogram from independent features. It can be expressed by the following equation.

$$\mathbf{x}^\tau = \sum_{i=1}^{\rho} y_i^\tau \mathbf{z}_i \quad (2.13)$$

The proposed subspace method is useful when the independent spectral features correspond to individual sources in a mixture [39].

2.6.4 Nonnegative Matrix Factorisation

Nonnegative Matrix Factorisation (NMF) is a process that approximates a single non-negative matrix as the product of two non-negative matrices, following the equation below.

$$\mathbf{V} \approx \mathbf{W}_v \mathbf{H}_a \quad (2.14)$$

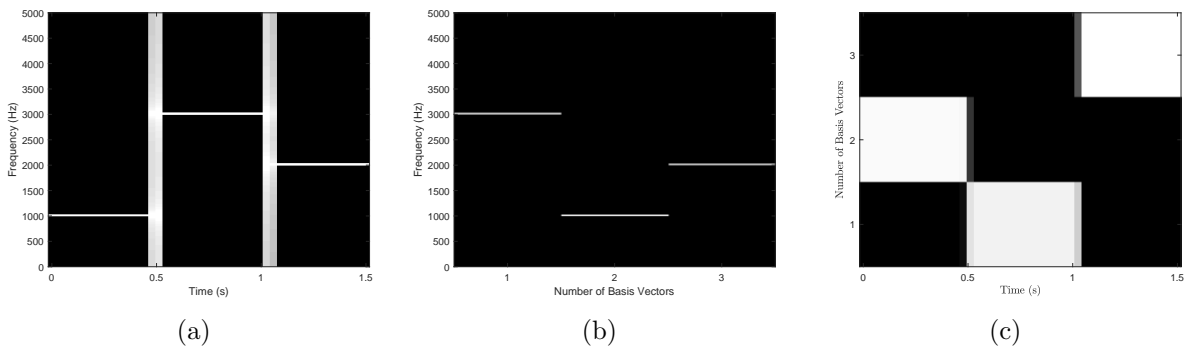


Figure 2.13: Nonnegative Matrix Factorisation (NMF) decomposition of an input signal made up from three pure tones at frequencies 1 kHz, 3 kHz and 2 kHz. (a) Spectrogram, (b) Basis vectors \mathbf{W}_v , and (c) Activations \mathbf{H}_a .

where $\mathbf{V} \in \mathfrak{R}_+^{N_f \times N_t}$ is a non-negative input matrix; $\mathbf{W}_v \in \mathfrak{R}_+^{N_f \times N_z}$ is a matrix of basis vectors, or dictionary elements; $\mathbf{H}_a \in \mathfrak{R}_+^{N_z \times N_t}$ is a matrix of corresponding activations, weights, or gains; N_f is the number of frequency components of the input matrix; N_t is the number of time frames of the input matrix; and N_z is the number of basis vectors. Typically $N_z < N_f < N_t$ resulting in a compressed, low-rank approximation of the data \mathbf{V} [29].

When NMF is used for audio applications, the single-channel recording is usually transformed into a time-frequency representation and the magnitude or power spectrogram is used as the matrix \mathbf{V} . The process approximates the spectrogram as a linear combination of prototypical spectra, or basis vectors, over time.

A simple NMF decomposition is shown in Figure 2.13 where an input signal consisting of three consecutive pure tones is decomposed using $N_z = 3$. The frequency of the pure tones are: 1 kHz, 3 kHz, and 2 kHz. The original spectrogram, the basis vectors \mathbf{W}_v and the activations \mathbf{H}_a are presented. The matrix \mathbf{W}_v contains the frequencies of the pure tones while matrix \mathbf{H}_a has the time instants when those frequencies are active.

The NMF decomposition usually minimises a cost function, e.g. the generalised Kullback-Leibler Divergence (KLD), the Least Square Distance (LSD) or the Itakura-Saito Divergence (ISD). Moreover, the variations of the algorithm have been successfully exploited in source separation, and some examples are mentioned below [4].

2.6.5 Sparse Coding

Sparse coding algorithms represent the original mixture by selecting a reduced number of basis elements taken from a larger set. The strategy of representing the mixture in this way relies on the assumption that not all the notes are played simultaneously in most musical signals. So,

the mixture can be explained using just a few components in the matrix decomposition, most of the rest being considered to be relatively insignificant [17].

In the field of sparse coding, the term sparsity implies many zeros in a vector or a matrix. The global idea is to characterise a signal using as few basis elements as possible, taken from a wide collection of basis elements. Let $x \in \mathfrak{R}^n$ be a signal and $D = [d_1, d_2, \dots, d_q] \in \mathfrak{R}^{n \times q}$ be a dictionary of normalised basis vectors, then sparse representation aims to find a sparse vector $\alpha \in \mathfrak{R}^q$ such that $x \approx D\alpha$, where the vector α is regarded as sparse code. In most cases the dictionary is learned previously from training data.

2.6.6 Wavelet Analysis

Wavelet analysis has also been used to decompose arbitrary signals into localised contributions labelled by a scale parameter. Many applications of the method have emerged to recognise and visualise characteristic features of speech and music sounds [41].

Tzanetakis et al. described some applications of the Discrete Wavelet Transform (DWT) for the problem of extracting information from non-speech audio; in particular, automatic classification and beat attribute extraction were explored in various types of music signals [42].

A continuous wavelet-like transform was investigated by Paradzinets, Harb and Chen and applied to automatic music transcription [43]. The audio signal was sequentially modelled by a number of harmonic tone structures. On each iteration a dominant harmonic structure was considered to be a pitch candidate. The overall results showed good precision rates for monophonic and polyphonic examples from classic pieces, but unsatisfactory results on modern popular music were observed.

A Bark-Scaled wavelet decomposition was used by Litvin and Cohen to generate a different time-frequency representation of the mixture, in which higher frequency resolution is achieved at lower frequencies. The posterior mean is used to estimate mixture components within the proposed representation [44].

Additional thoughts on the decomposition and reconstruction of audio signals using the wavelet transform are discussed in [45] and time-frequency representations providing equal resolution on a log-frequency scale are presented.

2.6.7 Matching Pursuit

Matching pursuit is a factorisation method in which the input signal is expanded into a finite sum of dictionary elements or atoms. A redundant or overcomplete dictionary of atoms

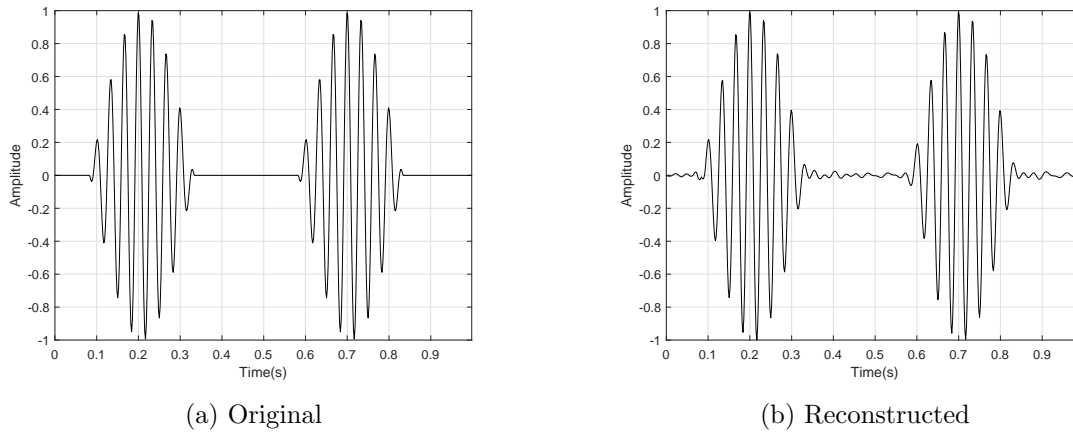


Figure 2.14: Decomposition of an input signal using matching pursuit.

allows the signal to be coded in terms of a minimal set of elements that provide an optimal fit by minimising the approximation error. Examples of atomic dictionaries often used are the ones formed by Gabor atoms, complex exponentials, wavelets, real sinusoids, and damped sinusoids [28].

The algorithm starts with an empty initial approximation and a residual that equals the original signal. Then, matching pursuit chooses the atom from the dictionary that best correlates with parts of the residual signal and generates a new one by subtracting the selected atom from the previous residual. The process repeats in every iteration and additional atoms are added to the approximation until a threshold criterion on the residual energy is reached, or after executing a maximum number of iterations. Figure 2.14 shows an example signal that is decomposed and reconstructed using Orthogonal Matching Pursuit with an overcomplete dictionary consisting of 1024 atoms based on wavelet packets and discrete cosine functions. In this case, the optimal fit has been achieved by selecting 25 atoms from the dictionary.

When polyphonic musical signals are decomposed using matching pursuit, the success of the factorisation depends upon the breadth of the dictionary. If a significant number of atoms from musical sources are added to the dictionary, its ability to decompose polyphonic music improves. However, large dictionaries introduce problems related to scalability and computational complexity.

Several attempts have been made to overcome this limitation; for instance, Tjoa and Liu proposed an approximate matching pursuit in which the signal is decomposed into a sparse combination of atoms with complexity that is sublinear in the size of the dictionary while the accuracy is preserved [46]. Another variant was presented in [47], in which an additional pre-processing step was added to the main sequence of matching pursuit, in order to perform an

analysis of the signal and extract important features. These features were then used to create dynamic mini-dictionaries comprising atoms that would correlate well with the underlying signal structures, thus leading to more efficient representations of particular supports of the signal.

Despite the benefits of atomic decompositions in terms of providing a compact representation of signals, they provide a less flexible framework for music processing than those delivered by other parametric alternatives, such as sinusoidal modelling.

2.7 Summary

In this chapter a brief introduction to the analysis of audio signals was presented, which included several important definitions from music theory that were considered useful for the understanding of the rest of this work.

A convenient classification of sounds was suggested, considering the scope of the research that will follow in further chapters. The differences between monophonic and polyphonic music were also addressed, emphasising the complexities associated with each case. Models for mixing processes were then discussed, while three different source separation problems were defined, based on the number of observed mixtures and underlying sources.

The representation of audio signals in the time-frequency plane was also addressed, by presenting several techniques which have been extensively explored in audio applications. While some of them impose an equally spaced grid on the time-frequency plane, other alternatives provide logarithmically distributed frequency axes, which are inspired in the way humans are believed to handle sound waves. Despite the efforts of creating new forms of time-frequency representations, the standard STFT still represents the basic representation in many separation approaches, mostly because of the efficiency of its computation.

Finally, an introduction to several decomposition models for audio signals was provided, in which advantages and limitations were discussed. While some of the models are currently considered as being too rigid, others are only suitable for multi-channel audio mixtures or present problems during the clustering of the basic functions. In the end, the final application is what really defines the type of decomposition model that is best to achieve the desired outcome.

Chapter 3

Audio Source Separation Techniques

3.1 Preamble

The main objective of this work is to propose an iterative approach to the separation of musical structures arising from different sources in single-channel recordings. The basic problem is introduced in Section 3.2, followed by a review of previous approaches in audio source separation. This discussion is divided in two parts; Section 3.4 deals with model-driven approaches, whilst Section 3.5 presents data-driven methods. Given that the system proposed here is a model-driven approach, the discussion of this type of algorithm has been emphasised and additional references have been provided. Data-driven approaches have been included as a way to present different trends that have received significant attention in recent years.

The way in which previous information and assumptions are handled is different in every algorithm, but categories can also be established depending on how general these assumptions are, or how much information of the underlying sources is required beforehand. These categories are presented and discussed briefly in Section 3.6, also providing examples of user interfaces that have been proposed in previous studies.

3.2 Basic Problem

The human auditory system is able to receive different sounds coming from different sources and then discriminate any particular source from the mixed signals [17]. When it comes to

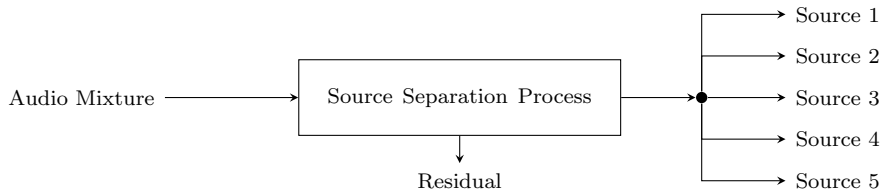


Figure 3.1: Block diagram of a general separation process.

artificial systems, the problem of separating multiple audio streams from polyphonic recordings is a very challenging one, since it is significantly ill-posed [48].

Audio source separation is the signal processing task which consists of recovering the original constitutive sounds, called sources, of an observed mixture, which can be either single-channel (monaural) or multi-channel (stereo, surround sound, etc.). Monaural and stereophonic recordings are still very common within the audio industry, where the sources are those individual sounds corresponding to musical instruments or voices [49]. Source separation techniques have a wide range of interesting applications, many of which were discussed in Section 1.4. A general structure of an audio source separation process is presented in Figure 3.1.

An ideal audio source separation system should be able to take a polyphonic input signal and perform an optimal characterisation of the underlying sources, in order to partition the energy of the mixture into a number of output channels, each one associated with an individual source. In this case, all the energy of the mixture should be allocated within the extracted sources so that the original mixture can be perfectly reconstructed by adding them together. Real systems, on the other hand, exhibit several limitations that do not allow a complete characterisation of the underlying sources, leading to incomplete separation results where the estimated sources are no longer disjoint. Also, a residual channel has to be generated to allocate unresolved energy of the mixture that could not be allocated to any of the estimated sources.

The way in which the output signals are estimated and separated will depend on the core structure of the algorithm, its assumptions, and the information or features that are used to characterise the sources. Some approaches estimate a set of output channels in just one pass, while other methods are designed to be iterative, i.e. one source is estimated at a time.

3.3 Separation of Estimated Sources

After a suitable characterisation of the underlying sources has been obtained, the next step is to isolate the identified structures from the rest of the original mixture in order to form a set of estimated sources. This stage is commonly referred as synthesis or extraction depending

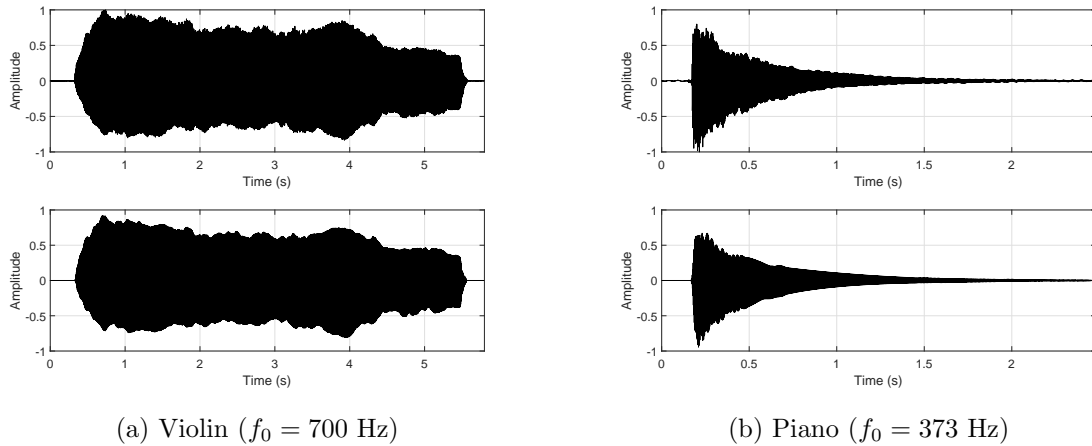


Figure 3.2: Sinusoidal synthesis of two different notes. (Top) Original signal. (Bottom) Sinusoidal synthesis.

on the strategy used to accomplish this task. Two of the most popular ones are described as follows.

3.3.1 Sinusoidal Synthesis

This strategy is based on sinusoidal modelling of sounds and characterises each source as a set of time-varying parameters which are then used to control a number of sinusoidal oscillators to reassemble each frequency component, and the estimated source is obtained by adding their output signals. The quality of the reconstruction depends on the accuracy of the estimated parameters, which tends to be relatively high for well-structured sounds and significantly lower for impulsive and noise-like sounds.

An estimated source can be cancelled out from the original mixture by means of subtraction in the time domain, but any deviation of the estimated parameters will lead to a non-perfect subtraction of the content and will lead to artefacts in the residual.

Figure 3.2 shows a comparison between two real notes and their corresponding sinusoidal synthesis obtained from estimated parameters of their first ten harmonic partials. Given that the model is only capturing the details of the stationary part of the signals, there are subtle differences between the originals and the reconstructions, especially for the piano note, which has a prominent attack that cannot be completely characterised by the model. This issue could lead to a degradation of the naturalness of the reconstructed sounds.

3.3.2 Time-Frequency Masking

An alternative to sinusoidal synthesis for retrieving an estimated source is to extract it from the original mixture. This extraction is achieved by multiplying the original time-frequency representation of the mixture by a suitable mask, which provides weightings on the time-frequency bins depending on their likelihood of belonging to one particular source. The extracted time-frequency representation is then converted back to the time domain by using a suitable inverse transformation. Each cell of the mask adopts a real value within the interval $[0, 1]$, and they can be classified as binary or non-binary [23].

Binary masks, also referred as hard masking, assume that the underlying sources are highly disjoint and their separation is feasible with little risk of creating interference that could be perceptually significant. If the target and interference energies are denoted by $s(t, f)$ and $n(t, f)$, respectively, then the Ideal Binary Mask (IBM) associated with the j -th source is defined as [50]:

$$IBM_j(t, f) = \begin{cases} 1 & \text{if } s(t, f) - n(t, f) > LC \\ 0 & \text{otherwise} \end{cases} \quad (3.1a)$$

$$(3.1b)$$

where the threshold LC is a local SNR criterion in dB. Advantages of binary masks over sinusoidal synthesis include higher robustness against background noise and room reverberation effects, while capturing additional details and features of the original sources. However, if there is significant overlap between different sources, the use of binary masks is expected to introduce significant amounts of interference, leading to lower separation performances.

To overcome the limitations of binary masks, the use of generalised Wiener filters as soft masks has been suggested. These are non-binary masks defined as [51]:

$$WSM_j(t, f) = \frac{X_j^r(t, f)}{\sum_{q=1}^Q X_q^r(t, f)} \quad (3.2)$$

where $X_j(t, f)$ is the estimated spectrogram of the j -th source, $X_q(t, f)$ is the estimated spectrogram of each q -th source, for $q = 1, 2, \dots, Q$, and Q is the total number of sources. The exponent $r = 1$ is used for magnitude spectrograms, while $r = 2$ is used for power spectrograms.

This approach has been designed to allocate the energy of a time-frequency tile across the sources according to a least-squares best fit, so that the extracted sources sum together to give the original mixture. A significant advantage of this strategy is that separation errors and artifacts

are often masked because of the presence of other sources [51], but if phase reconstruction is not considered, then the masking process will not be able to resolve overlapping content.

Other types of weighted non-binary masks can be constructed in those cases where the sources overlap, so the energy in a particular time-frequency tile can be shared between the interfering sources. However, this approach requires the identification of overlapping regions during the estimation of the spectrogram of each individual source. Sections 3.4 and 3.5 present an overview of the different ways in which separation methods handle source estimation.

3.4 Model-driven Separation Approaches

Approaches in this category attempt to capture knowledge and derive decisions by using explicit representations and rules. A model-driven system would take an audio signal and decompose it into some basic elements in order to perform measurements and compare against a set of rules that are previously defined to detect some specific patterns. They rely on a deep understanding of audio signals and mixing processes to establish the set of rules that controls the separation process. Their principal limitation is that models cannot accommodate an infinite amount of complexity and therefore they have to be simplified, which makes it difficult to work with noisy or exceptional cases. The following sections represent a review of model-driven approaches that have been applied with some success to audio source separation.

3.4.1 Computational Auditory Scene Analysis

The development of Computational Auditory Scene Analysis (CASA) has been inspired by psychological research that investigates how the auditory system could segregate acoustic signals into streams that correspond to different sources. It has been extensively used in speech separation and does not require any strong assumption on the acoustic properties of sources of interference. CASA systems usually consists of two parts: segmentation (decomposition into sensory segments) and grouping (segments of the same source are put together) [52].

When it comes to music analysis, CASA focuses on extracting higher-level musical information, such as pitch, rhythm, etc., from the input signal using computational algorithms and psycho-acoustical cues, which are then used during the grouping stage. Signal representations commonly used in CASA include magnitude spectrograms and correlograms, while the principal cues used for grouping are usually harmonicity, onset/offset times, and timbre.

A variation of CASA was presented in [3] where a segmentation system consisting of four stages was presented. A filter bank and a simulation of neuromechanical transduction by inner hair cells were used to model the auditory periphery. Then a symbolic description of the auditory scene is constructed with information from some particular representations called auditory maps. A search strategy was used to group elements according to their fundamental frequencies and onset/offset times. The system was used to separate speech from a variety of intrusive sounds, reporting an increment in SNR after separation for each noise condition.

A substantial improvement in performance was reported in [52] by using an objective quality assessment of speech to instruct CASA during the grouping process, yielding a better subjective perceptual quality of the separated speech sources. In [53] the idea of combining source localisation and source attributes is further explored by incorporating beamforming within a CASA-based system, which performs sound source separation by temporally and spatially filtering a multichannel input signal, and then grouping the resulting signal components into separated signals, based on source and location attributes.

Despite the benefits of many aspects included within the CASA framework, it is still inefficient when dealing with audio sources having similar pitches or a large number of overlapping partials, which tend to remain undetected.

3.4.2 Statistical Approaches

Most separation algorithms based on high-order statistics make the assumption of statistical independence of the underlying sources, and non-Gaussianity of their probability density functions. The observed input mixtures are then considered as linear combinations of the sources, so their separation is achieved by the identification of the mixing matrix, which can be obtained by detecting and analysing single-source sections, or by using other techniques such as Independent Component Analysis (ICA) or Independent Subspace Analysis (ISA).

Abdallah and Plumbley used ICA to decompose broadcasted audio signals into basis vectors and then explored their characteristics in terms of their position and spread in the time-frequency plane. It was observed that, under certain circumstances, these basis vectors corresponded closely to a wavelet basis and could be used to learn interesting representations of audio signals [54].

Although the use of ICA in source separation has concentrated on the determined and over-determined cases, several algorithms have been proposed for the single-channel case. In 2003, Jang et al. [55] presented an approach based on CASA and ICA, in which the main idea was

to exploit the inherent time structure of sound sources by learning *a priori* sets of time-domain basis functions that encode the sources in a statistically efficient manner.

The combination of ICA and binary masks was explored in [56], where an iterative algorithm to extract speech signals from audio mixtures was presented, in which the separation of six mixed speech signals from within two observed mixtures was reported as one of the achievements. A similar method was proposed by Barry et al. [57] where ICA was applied to contiguous frames of the STFT of the input mixture in order to generate the short-time spectra corresponding to each of the sources. Suggestions to cope with the scaling and permutation ambiguities were also presented.

Davies and James stated that the linear nature of the separation equations, considered within the ICA framework, limits the separation of sources when there is substantial overlap between them [58]. A mathematical framework was used to show that standard ICA can perform source separation in single-channel inputs only when the sources have disjoint spectral support.

A different decomposition of the time-frequency plane was investigated in [59], where ICA was used to construct a set of spectral and time bases, from which a number of time-frequency components were obtained. Then, a grouping algorithm was proposed for clustering these elements into subspaces in order to generate the constituent components of the mixture.

Another single-channel variation of ICA was proposed in [60], where the FastICA algorithm was evaluated in the task of separating real audio signals. This technique considered the application of ICA to a set of delayed versions of the original mixture in order to identify independent components, which were clustered by K-means and used as impulse responses to design separation filters. Positive results were reported for input mixtures of non-overlapping sources.

Casey and Westner proposed a separation method based on ISA in which the extracted features are grouped by partitioning a matrix of independent component cross-entropies that they called ixegram [39]. The proposed ixegram measures the mutual similarities of components in an audio segment with the aim of clustering them to yield the source subspaces and time trajectories.

A similar idea was exploited by a hybrid system of Empirical Mode Decomposition (EMD) and Principal Component Analysis (PCA), designed by Taghia and Doostari [40]. It was proposed as a way to construct artificial observations from the mixture. For separation purposes, the FastICA algorithm is used to find independent components, while a Kullback-Leibler Divergence (KLD) based K-means algorithm is used for clustering.

The estimation of the mixing matrix from single-source points has also been applied to under-determined source separation problems. Single-source points refer to time-frequency tiles where only one source exists, hence, they present a good directional clustering property. Two algorithms for detecting single-source points were presented in [61] and [62], where the complex-valued STFT of the input mixtures was further analysed, assuming that any two columns of the mixing matrix are uncorrelated. The method in [62] was then applied in [63] to the separation of speech signals.

3.4.3 Harmonicity

The assumption of harmonicity has been widely used for handling harmonic or nearly-harmonic instruments. Musical pitched sounds are modelled as the combination of two components, namely, deterministic and non-deterministic. The harmonic or nearly-harmonic model is used to characterise the deterministic section, where the sound is modelled as a series of slowly-decaying time-varying frequency components, comprising the fundamental partial and the overtones placed at integer multiples of the fundamental frequency or pitch. The amplitudes and phase angles of these components, together with the fundamental frequency, are time-varying parameters determined by the physical properties of the musical instrument and the person that is playing it. The non-deterministic section is assumed to contain any other impulsive structure, such as transients or attacks, and shaped noise, which are usually handled as stochastic processes [48, 64].

Several different approaches have been proposed for the separation of harmonic sounds from polyphonic mixtures. Many of these solutions include a previous multipitch estimation stage to generate the fundamental frequency trajectories for the underlying sounds, while different techniques are used for the separation, including additive synthesis [24, 65–67] and time-frequency masking followed by a suitable inverse transformation [26, 68–72]. Limitations of these approaches are tightly related to those of multipitch detection, in which the number and relative volumes of the underlying sources, the amount of overlap between them, and the presence of percussive or interfering sounds, are typical factors that degrade the quality of the separation.

Harmonicity has also been combined with spectral decomposition methods such as NMF [73–78], PLCA [79], and matching pursuit [46, 47]. However, several difficulties have been found, including the high number of parameters that have to be learned from solo excerpts and the complexities of clustering the resulting basis spectral vectors into separated sources.

3.4.4 Separation of Overlapping Harmonics

One of the key issues in the separation of pitched signals relates to the disentangling of overlapping harmonics, which are very common given the wide use of the twelve-tone equal temperament scale in tonal music [80]. Regardless of the separation approach, the goal is to estimate the parameters (amplitudes, centre frequencies and phase angles) of the underlying frequency components that coincide in any observed overlapping partial. Depending on the number of frequency components and their degree of overlap, achieving a proper separation represents an ill-posed problem.

Separation approaches can either treat this problem explicitly or implicitly. Separation techniques such as CASA aim to separate sound sources by means of acoustical cues without attempting the separation of overlapping partials, which limit their separation performance. Other methods, such as NMF or any similar subspace analysis, deal with this problem in an implicit way, where the spectral magnitudes in the overlapping regions are observed in order to recover the original components. However, given that the correct phase angles of these components cannot be estimated, an optimal separation is unlikely [70].

When overlapping harmonics are explicitly handled, methods usually rely on the availability of pitch trajectories for the underlying sources, which are normally detected first, in order to identify overlapping regions by considering a proximity criterion. Then, overlapping harmonics are resolved by means of sinusoidal modelling [81], spectral filtering [69], common amplitude similarity [70], amplitude and phase reconstruction [82], or harmonic bandwidth companding [80]. Positive results were reported in those cases where the pitch trajectories were accurately estimated and the underlying components were not fully overlapped.

3.5 Data-driven Separation Approaches

Data-driven solutions focus on identifying the best way to separate the underlying components of a mixture by previously observing and learning from a large number of examples within a training database. The reliability of these systems depends on the size of their training database and the relevance of the audio examples included, which have to be correctly labelled before starting with the training stage. The strength of this alternative is that it does not depend on a set of explicit rules, since the system is able to learn its own way to achieve the best separation of the underlying sources. Current limitations include long development stages

and lack of generality due to limited training data. The following sections present the main characteristics of a selection of previously-presented data-driven source separation approaches.

3.5.1 Deep Neural Networks

The interest in machine learning has grown significantly during the last few years, thanks to the availability of greater computational power and memory space. Within machine learning, the Deep Neural Network (DNN) has become one of the most popular architectures, inspired by information processing and communication patterns in the biological nervous system.

In general, a neural network is a collection of inter-connected units or nodes, usually referred to as artificial neurons. Each neuron can receive and process an input signal before sending an output to other artificial neurons connected to it. The output is usually computed by some non-linear function of the sum of the inputs, while connections between neurons typically have weights that can be adjusted during the training stage. Neurons are normally arranged in layers which perform different transformations on the inputs. An architecture consisting of an input layer, an output layer, and two or more hidden layers is considered a DNN. If enough data is available for training, a DNN can effectively learn representations and mappings. Different training strategies have been proposed, where gradient descent and back-propagation seem to be the most popular ones [83].

In single-channel audio source separation, DNNs have been introduced as part of two principal approaches. The first one uses DNNs to map features of the original mixture into features of the sources directly. The second approach maps the original mixture into some spectral masks that explain the contribution of each source in the mixed signal. While the first case is less sensitive to variations of the mixing ratio, the second one benefits from the bounded nature of spectral masks by imposing limits to the values a source can adopt. In both cases, training data is usually required and several alternatives are available to model these data, including Gaussian mixture models, Hidden Markov Models (HMM) or factorial HMM.

Applications of DNNs to the analysis and classification of audio signals include the separation of speech from music [84], singing voice separation and pitch extraction [85], and musical source separation and enhancement [86,87]. A comprehensive review of lead and accompaniment separation methods based on DNNs is presented in [83].

3.5.2 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a class of deep learning network which explicitly assumes that the input signals are images, reducing significantly the amount of parameters in the network. CNNs consider two-dimensional input vectors of pixel intensities and exploit the local spatial correlation among input neurons to learn localised features. Three types of layers are used to construct CNNs: convolutional layers, pooling layers, and fully connected layers.

In order to process an audio signal, a time-frequency representation is first generated to provide the network with a two-dimensional input array, from which a set of higher level features can be extracted. In source separation approaches, these features are used to design time-frequency masks for source extraction.

Chandna et al. [88] proposed a low-latency monaural source separation framework using CNNs, where models for the separated sources were learned by finding compressed representations for the training data. A similar approach was presented in [89], in which the aligned score of the music was used to learn source models. Then, a set of soft masks was designed to separate the underlying sources from real-life performances of the music. A sound event detection method was proposed in [90] where a variation of the CNN architecture was used to overcome limitations related to the lack of time-frequency invariance and temporal restrictions.

Despite the overall positive performance that data-driven approaches have already shown in terms of separation quality, they all seem to suffer from the same limitations. First, gathering a large amount of training data can be very difficult for some applications, which significantly restricts the learning capabilities of some methods that need large training sets. Second, model parameters in data-driven approaches are usually difficult to interpret, which makes it unclear how to provide user interaction within human-computer environments [83].

3.6 Prior Information in Source Separation

Audio source separation algorithms can also be classified in terms of the amount of prior information required about the sources and the mixing process.

3.6.1 Non-Informed Methods

This category groups separation methods in which the sources are completely unknown and training information is not used. These methods typically rely on a few very general assumptions,

e.g. harmonicity or statistical independence of the sources. Algorithms within this group are sometimes called blind or unsupervised, and some examples are presented below.

In 2001, Virtanen and Klapuri presented an audio source separation strategy based on multipitch estimation and sinusoidal modelling. The harmonic structure of sounds and spectral envelope continuity were used to estimate parameters of the source signals that allowed their separation via sinusoidal synthesis [65]. This approach was extended by Siamantas in 2009 by incorporating a cyclical, residual-based structure, to estimate and separate harmonic sounds from polyphonic mixtures using spectral filtering [23].

The algorithm presented by Herris in 2007, on the other hand, was based on the use of ISA to decompose and separate single-channel audio excerpts. Whilst good results were reported for stationary mixtures of synthetic sounds, the algorithm failed to separate real non-stationary audio recordings. For this case, the output signals were still mixtures of the sources while the audio quality was severely degraded [38].

In 2011, Gau proposed an unsupervised method in which several variations of the NMF algorithm were used to separate different audio components from music recordings. While separation results were reported as satisfactory for several real example mixtures, the estimation of the hidden parameters proved not to be straightforward [4].

Another algorithm, based upon the estimation of pitch trajectories, was presented by Duan et al. in 2014 [91]. The presented approach does not require previous training processes of the source models. Instead, the problem was cast as a constrained clustering process, where each cluster corresponded to one source.

3.6.2 Informed Methods

The algorithms presented in the previous section usually make some assumptions regarding the type of sources present in the mixture under analysis. These assumptions are general and they define the way in which the separation method is designed. Some other approaches, on the other hand, use additional information in order to guide the separation process and improve the quality of the results [17].

This additional information can be provided in several ways, e.g. the user can be asked to provide the number of sources in the mixture, a musical score, the time intervals of activity for each source, or a sung target sound. Expert users can also be asked to choose the desired source by selecting components from intermediate separation results, or by making corrections in

automatically estimated melody lines [75]. Some relevant informed methods will be reviewed in the upcoming paragraphs, focusing on those cases where the input is a single-channel recording.

Every and Szymanski presented a separation algorithm in which the transcribed score is required, *a priori*, using the Musical Instrument Digital Interface (MIDI) format [69, 92], as a starting point for the process. In each time frame, a spectral filter is constructed whose effect is to extract all harmonic partials associated with any particular instrument, from the original spectrum of the mixture. Three different filter shapes were also proposed as a way to handle overlapping harmonics.

The separation method by Every was extended in [93], when an additional stage was incorporated to separate overlapping impulsive content, by interpolating within individual frequency bands of the decaying envelope of each source across overlapping sections with other sources. Furthermore, three additional methods for separating overlapping inharmonic content were presented in [94], based on autoregressive models, bandwise noise power interpolation, and correlation of the harmonic amplitudes with the shape of the spectral noise envelope.

In 2009, Smaragdis and Mysore presented a user-based approach for isolating and removing sounds from single-channel mixtures, in which the user is required to present a guide sound that mimics the desired target the user wishes to separate [95]. Using that guide as a prior in a statistical sound mixture model, a methodology was proposed to efficiently extract complex structured sounds from dense mixtures.

Another user-guided audio source separation algorithm was presented by Durrieu and Thiran in 2012 [75]. The proposed user interface allows the user to select the desired audio source, by means of the assumed fundamental frequency track of that source. The system then automatically refines the selected pitch trajectory and separates the corresponding source from the mixture.

A supervised source separation system was introduced by Fuentes et al. in [79]. The time-frequency representation of the signal is first analysed through an algorithm that provides an estimation of the polyphonic pitch, from which the user can select the notes to be extracted. The separation is then achieved automatically by means of time-frequency masking. The proposed Graphical User Interface (GUI) is presented in Figure 3.3.

To decompose single-channel recordings into their respective sources an interaction paradigm and separation algorithm were presented by Bryan in 2014 [29]. The method works by allowing the user to roughly paint on time-frequency displays of the original mixture. The rough annotations are then used to constrain, regularise or otherwise inform an algorithm based on

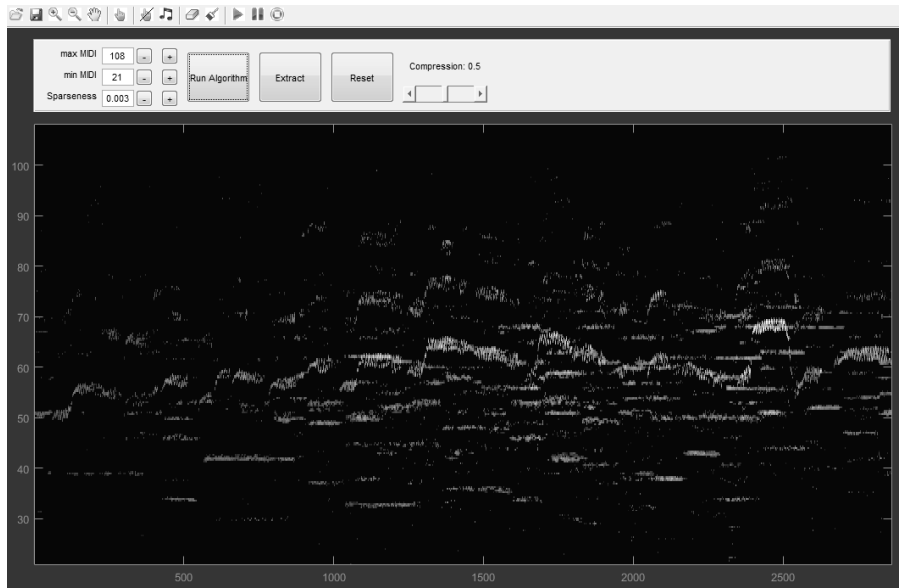


Figure 3.3: Blind harmonic adaptive decomposition user interface [79].



Figure 3.4: Interactive Sound Source Separation Editor (ISSE) user interface [29].

Probabilistic Latent Variable Models (PLVM). The output estimates are presented back to the user and the entire process is repeated again until the desired results are achieved. The system, also referred to as Interactive Sound Source Separation Editor (ISSE), can be seen in Figure 3.4.

The use of spectral models of instruments as a tool to discriminate the energy coming from different sources in a mixture was explored by Rodríguez-Serrano [17]. These models are trained beforehand and then used to guide the separation process, which is based on an informed variation of the NMF algorithm. The process can also be run without the training stage, hence becoming an unsupervised method. An additional strategy to solve the separation

of overlapping partials was also proposed, in which the instrument models are used to estimate the magnitude and phase of the separated overlapped harmonics.

3.7 Performance Measurement and Evaluation

Measuring performance in source separation is strongly related to the quality assessment of audio signals. An appropriate set of performance measures should allow the evaluation and comparison of different algorithms and models when they are applied to usual audio source separation problems. As may be foreseen, this topic is a research area on its own, where numerous alternatives have been proposed, ranging from human-based perceptual evaluation to objective quality metrics. A brief review of the most popular approaches is provided as follows.

3.7.1 Blind Source Separation Evaluation

Blind Source Separation Evaluation (BSS Eval) is an objective performance criterion initially proposed by Vincent et al. in [96], that helps evaluating and comparing source separation algorithms. Separate performance measures are computed by comparing each estimated source \hat{s}_j with a given true source s_j . Each estimated source is first decomposed as follows.

$$\hat{s}_j = s_{\text{target}} + e_{\text{interference}} + e_{\text{noise}} + e_{\text{artifacts}} \quad (3.3)$$

where $s_{\text{target}} = f(s_j)$ is a version of the true source s_j modified by some allowed distortion $f(\cdot)$, and where $e_{\text{interference}}$, e_{noise} , and $e_{\text{artifacts}}$ are the interference, noise and artifact error terms, respectively. These terms should represent the part of \hat{s}_j perceived as coming from s_j , from other unwanted sources, from sensor noise, and from other causes. Three different measures, namely, Source to Distortion Ratio (SDR), Source to Interference Ratio (SIR), and Source to Artifact Ratio (SAR), are defined and computed as energy ratios, expressed in decibels (dB), following the equations below.

$$SDR = 10 \log_{10} \frac{\|s_{\text{target}}\|^2}{\|e_{\text{interference}} + e_{\text{noise}} + e_{\text{artifacts}}\|^2} \quad (3.4)$$

$$SIR = 10 \log_{10} \frac{\|s_{\text{target}}\|^2}{\|e_{\text{interference}}\|^2} \quad (3.5)$$

$$SAR = 10 \log_{10} \frac{\|s_{\text{target}} + e_{\text{interference}} + e_{\text{noise}}\|^2}{\|e_{\text{artifacts}}\|^2} \quad (3.6)$$

These objective measures assign equal weights to all error terms, which means that all types of distortions contribute equally to the overall quality of the extracted source [97]. A set of MATLAB[®] functions, created by Févotte et al. and referred to as *BSS_Eval Toolbox*¹, is available online and can be used to calculate the aforementioned objective measures [98].

Given that BSS Eval has exhibited a correlation similar to human perception in certain cases, it constitutes an evaluation framework widely used up to this day [83].

3.7.2 Perceptual Evaluation of Audio Source Separation

An alternative set of measures called Perceptual Evaluation of Audio Source Separation (PEASS), was introduced in [99] as a way to predict the Mean Opinion Score (MOS) of human listeners. The authors proposed a similar decomposition scheme for the input signal considering different types of distortion, namely, interference, artifacts, and target distortions.

In a study conducted by Cano et al., evaluation results obtained via PEASS exhibited in general higher levels of correlation with the subjective listening scores than those obtained via BSS Eval, though none of the correlations observed was very strong [97]. According to Rafii et al., despite the relevance of the methodology considered in PEASS, it has not been widely accepted in practical scenarios, mostly because of its computational costs and its emphasis on speech [83].

An implementation of PEASS in MATLAB[®], which has been used in several editions of the Signal Separation Evaluation Campaign (SiSEC), is also available online².

3.8 Summary

The present chapter has introduced the problem of source separation in single-channel audio mixtures and explained the principal sections of the basic processing framework. The definition of estimated source was provided, while two frequently used methods of separation and reconstruction for these estimated sources were discussed. In both cases, it was emphasised that the separation quality depends on the ability of the system to capture the contributions of each underlying source within the mixed signal.

Whilst sinusoidal modelling has proven to be a simple and effective reconstruction process for a wide variety of sounds, it can fail to capture impulsive energy that typically exists in music.

¹http://bass-db.gforge.inria.fr/bss_eval/

²<http://bass-db.gforge.inria.fr/peass/PEASS-Software.html>

Time-frequency masking, on the other hand, is prone to produce distortion when significant overlap exists among the underlying sources.

A comprehensive review of previous source separation algorithms for audio signals followed, in which the principal advantages and limitations of model-driven and data-driven approaches were emphasised. The well-defined, but certainly rigid, structure of model-driven methods was contrasted with the flexibility of machine learning, which can suffer from highly demanding training stages where a large number of labelled examples are required.

Different alternatives to bring prior information into the algorithms provided an additional way to classify different separation approaches. The classic blind approach has been gradually replaced by interactive interfaces, where the end-user plays an important role in controlling the estimation process, simplifying the separation of complex audio mixtures and improving quality.

Finally, a review of two different strategies for the evaluation of source separation approaches was presented. This showed that objective measures are still widely accepted, while perceptual metrics are often considered as quite computationally demanding. Hence, in this work, performance evaluation will be carried out largely using the suggested objective measures.

Chapter 4

An Iterative Note Event-based Multipitch Estimator

4.1 Preamble

The task of estimating the fundamental frequencies of concurrent pitched sounds, hereafter referred to as multipitch estimation, represents a significant challenge in audio signal processing, arising from the wide range of harmonic or nearly-harmonic instruments and voices commonly used to make music. However, the many potential applications have generated a significant increase in interest in this field. Direct applications include automatic music transcription (AMT) [14, 100, 101], melody extraction [102–104], query by humming [105–107], lead and accompaniment separation [67, 72, 75, 83, 108], singer identification [109, 110], among many others.

The challenge increases rapidly as the polyphony of a piece of music goes up. A larger number of simultaneously played instruments or voices, in general referred to as sources, reduces the probability of estimating their pitches correctly [100]. The reason for this is quite simple. Time-frequency representations have limited resolution, both in time and in frequency, which makes it very difficult to disentangle harmonically related partials, particularly when the distance between their centre frequencies is reduced. When several partials are very close, it is said that they overlap, and the result is a distorted shape that gives little information about the parameters of the original components. The likelihood of overlapping partials increases with the number of concurrent sources.

For evaluation of multipitch detectors, the Music Information Retrieval Evaluation Exchange (MIREX)¹ considers two subtasks: Multiple-F0 Estimation (MFE) and Note Tracking (NT). The first requires the systems to return all active pitches at fixed time steps, whilst in the second one systems return the fundamental frequencies along with onset and offset times, for all note events present in the audio mixture [111]. In MIREX 2017, for example, twelve algorithms participated in the MFE subtask, while nine methods were presented in the NT subtask. Among all participants, only one used an iterative approach for melody estimation. The rest of the systems were designed following a rather conservative strategy, where pitch contours for all concurrent sounds in the mixture were estimated jointly.

An iterative approach should be more effective than the joint one, as the level of interactions between concurrent sources can be reduced in every iteration. Louder musical notes can be detected first. As they are removed from the mixture the residual should start to reveal weaker notes previously masked by the louder ones.

In this chapter, an iterative approach is proposed in which pitch contours are estimated in sections. Instead of simultaneously tracking all sources across the whole duration of the recording, the proposed system focuses on constructing pitch trajectories for individual note events which are identified, separated in the frequency domain, and extracted from within the original mixture in every iteration, starting with the louder and longer ones. The process continues until the energy left in the residual is below a significance threshold, or when a maximum number of iterations is reached. All detected contours are then reviewed and recombined to estimate the final set of trajectories in the original mixture.

4.2 Related Work in Multipitch Estimation

Generating a set of fundamental frequency estimates for a set of concurrent sources in an audio mixture is usually defined as a low-level task. It is the first step towards solving high-order problems such as principal melody extraction and automatic music transcription [112]. Existing algorithms are designed to deal with this problem at different levels; some methods only focus on generating pitch estimates in every time frame, while others also involve a higher interpretation of the estimates, such as note tracking and automatic pitch streaming.

¹https://www.music-ir.org/mirex/wiki/MIREX_HOME

4.2.1 Multipitch Detectors

Following Benetos et al. [14], multipitch estimators can be classified according to the models employed, as feature-based, statistical model-based, and spectrogram factorisation-based.

Feature-based algorithms concentrate on measuring the relative salience of pitch candidates using time-frequency representations. For example, Klapuri considered a computational model of the human auditory periphery system [113], from which fundamental frequencies were iteratively detected and cancelled out. Ponce de León et al. implemented a similar approach using the Continuous Complex Wavelet Transform (CCWT) [114]. Similarly, an equal loudness filter was used by Salamon and Gómez to enhance frequencies to which the human listeners are more sensitive [102], before evaluating a salience function based on the energy of prominent peaks in the frequency domain. In [115], a Modified Euclidean Algorithm (MEA) was applied to multipitch detection and melody tracking, emphasising special cases such as the missing fundamental problem.

Multiple or combined transformations have also been explored. For instance, Su and Yang [116] exploited information from spectral and temporal representations of the sound mixture, namely the magnitude spectrum and logarithmic cepstrum, to increase the salience or prominence of real pitches. Rychlicki-Kicior et al. went even further by proposing a generic approach [117], where different preprocessing methods, transformations, and further processing stages could be arranged into parallel processes, whose outputs were jointly analysed for pitch selection.

Other approaches have used a probabilistic framework to define multipitch estimation as a maximum likelihood or maximum a posteriori problem. Duan et al. [118] proposed a greedy search strategy to estimate fundamental frequencies by establishing a maximum-likelihood scheme to simultaneously model spectral peak and non-peak regions of the observed power spectrum. The benefits of informing a model in Probabilistic Latent Component Analysis (PLCA), by means of inserting user annotations, were explored by De Andrade et al. in [119], obtaining better parameter initialisation and performance than previous unsupervised versions. Karimian-Azari et al. [120] exploited block sparsity within a least-squares solution for simultaneous sparse source selection, incorporating a Bayesian prior and data-dependent regularisation coefficients.

Recently, sparse representations have been applied to feature extraction. In [121], the spectrogram of the signal was represented by a sparse linear combination of a number of spectral templates within a dictionary, where pitch candidates were found using a sparse weight vec-

tor. Similarly, a time-recursive estimator using sparse recursive least-squares and an adaptive penalty was proposed in [122], assuming only a small number of active sources at any time, and without any training.

4.2.2 Note Trackers

Given a set of fundamental frequency estimates, corresponding to a polyphonic input signal, the next step is to identify continuous segments that are likely to be individual musical notes [91]. It is common for note trackers to exploit temporal continuity or smoothness of pitch contours by grouping pitch estimates having similar times and frequencies [115].

Different techniques have been applied to note tracking systems. For instance, Bittner et al. used modified Harmonic Locked Loops (HLL) to create pitch contours, with the advantage of also providing the amplitude of each harmonic partial over time [123]. Gao et al. [101] suggested two non-negative matrix factorisation algorithms to generate a novel time-frequency representation, which improved onset detection in piano transcription. Cheng et al. proposed a note tracking system for automatic piano transcription [124], based on Hidden Markov Models (HMM), in which four templates were considered and trained to represent stages of piano sounds (silence, attack, decay and release). Machine learning was further explored in [125] as a way to break a piano music frame-level transcription into segments, and to classify them as active or non-active events.

4.2.3 Multipitch Streamers

A third level in multipitch analysis is to stream a set of pitch estimates into trajectories corresponding to each individual source in the mixture. Kirchhoff et al. presented a semi-automatic music transcription system in which the user is asked to label some notes for each instrument in the recording. Different methods were then proposed to estimate spectral templates at pitch positions that could not be annotated by the user, in order to derive a complete set of instrument templates. Experimental results showed that more refined instrument models were generated when the user annotated several notes for each instrument [126].

An unsupervised pitch streaming algorithm was proposed by Duan et al. in [91] as a way to exploit timbre features to cluster fundamental frequency estimates in music and speech signals. A semi-informed system, proposed by Arora and Behera [127], used Hidden Markov Random Fields (HMRF) to cluster pitch estimates into streams, with the total number of concurrent sources needed beforehand.

Recently, neural networks have also been used to extract information from music. A two-stage approach, based on a Deep Neural Network (DNN) [85], was applied to singing voice pitch estimation. Initially the DNN was used to separate the singing voice from the background instrumentation, while its pitch contour was estimated using a tracking method based on dynamic programming. Another approach [128], specific to piano music, considered an architecture comprising models of acoustic and music language. Neural networks were used to handle the probabilities of pitches in an audio frame and the correlations between combinations over time, while the use of Convolutional Neural Networks (CNN) as acoustic models was found to yield the best performance.

4.3 Proposed System

Music is very rich and non-stationary. Different playing styles transmit emotions and stimulate the audience in many different ways. However, this inherent complexity usually creates some difficulties for multipitch detectors. For example, the masking of weaker notes by louder ones, instruments playing in harmonic relation, or musical effects such as vibrato and tremolo. In order to cope with some of these complexities, an iterative unsupervised approach is proposed, in which the final pitch trajectories are constructed from a set of shorter pitch contours corresponding to smaller sections, hereafter referred to as note events.

4.3.1 Note Events

An audio signal can be expressed as the combination of harmonic and non-harmonic components, as expressed in (4.1).

$$x(t) = \tilde{x}(t) + z(t) \tag{4.1}$$

The term $\tilde{x}(t)$ is the harmonic or nearly-harmonic part of the mixture, comprising most of its energy, whilst $z(t)$ represents all non-harmonic elements, such as transients, note attacks or any other non-harmonic source present in the original mixture. The term $\tilde{x}(t)$ can also be seen as a combination of smaller note events, characterised by a short continuous pitch trajectory. A single note event has several possibilities:

- It could represent a single musical note.
- It could correspond to a section of a longer musical note.

- It could group several consecutive musical notes with close pitches, not necessarily related to the same source.

Note events have limited support, since the elements of their pitch contours are non-zero only for a finite number of frames in the time-frequency representation.

If $p_{(q,l)}$ is the fundamental frequency of the q -th note event at time frame l , then a frame-wise representation of the note event in the time domain can be obtained by using additive synthesis as expressed below.

$$\text{NE}_{(q,l)}(t) = \sum_{h=1}^{H_q} A_{(h,l)}^q \cos(2\pi h p_{(q,l)} t + \phi_{(h,l)}) \quad (4.2)$$

where H_q is the maximum number of harmonic partials used for reconstruction, $A_{(h,l)}^q$ are their corresponding amplitudes, and $\phi_{(h,l)}$ is the phase information of the original mixture. Pitch trajectories of note events are selected from an array of fundamental frequency estimates, generated in every iteration by running a multipitch estimator on the current input signal.

4.3.2 Overview of the System

A simplified block diagram of the system is presented in Figure 4.1. In every iteration, the Short-Time Fourier Transform (STFT) of the current input signal is taken, using a frame size of 2048 samples, sampling frequency of 44.1 kHz, a Hanning window function, and no zero-padding. Two different overlapping rates are considered between adjacent frames: $H_{S1} = 50\%$ is used during the estimation of the predominant note event, whilst $H_{S2} = 87.5\%$ is used during the separation of its spectral content from the current input mixture. The reasons for this are associated with computational costs and with the limitations of the baseline algorithm that generates the initial set of pitch estimates in every iteration. Further discussion on this topic is provided in Section 4.3.7.

The multipitch estimator by Duan et al. [118] is used as a baseline process to obtain the initial array of fundamental frequency estimates of the current input mixture in every iteration. The algorithm in [118] represents a flexible option that allowed its effective integration with the rest of the propose framework. Moreover, its accuracy has been found to be adequate, while a MATLAB[®] implementation of the algorithm is available online².

An initial set of fundamental frequency estimates is generated by applying the algorithm in [118] to the lower-rate note event estimation spectrogram, and used to select the pitch contour of

²<http://www2.ece.rochester.edu/projects/air/resource.html>

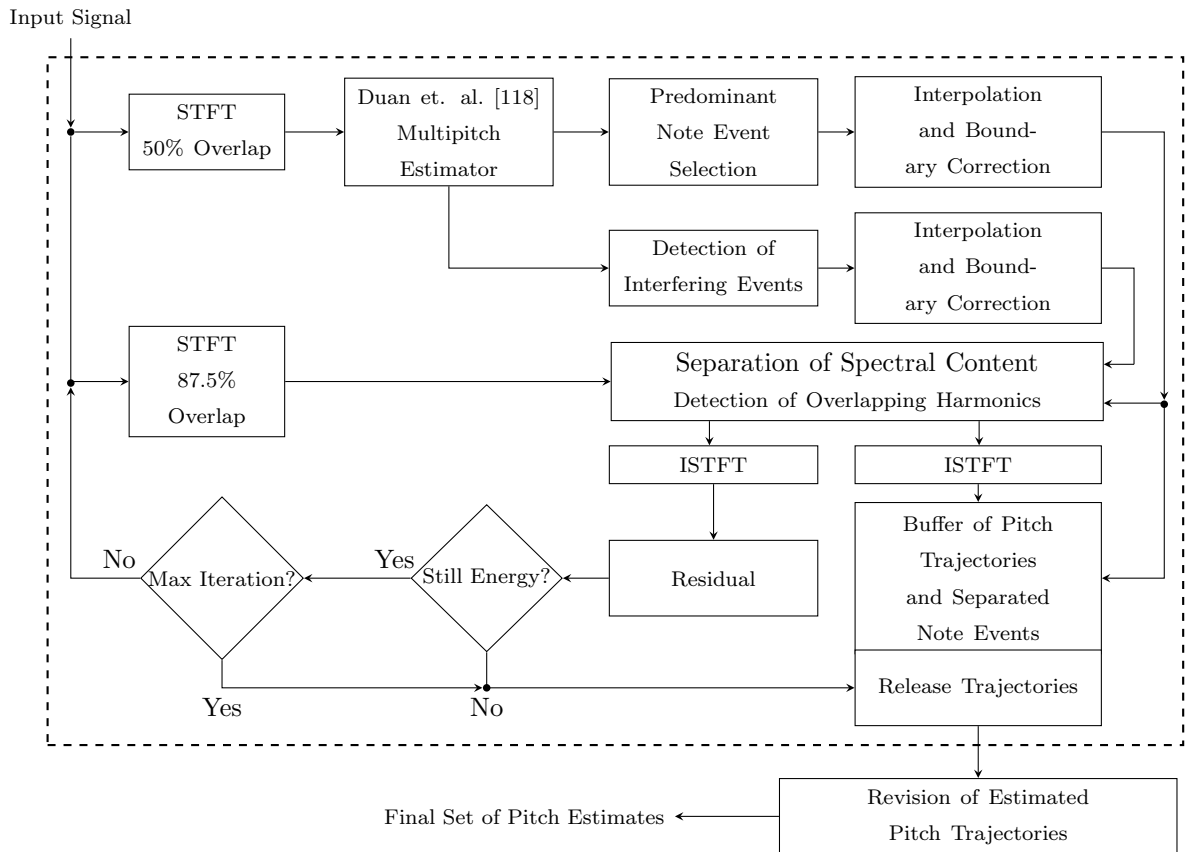


Figure 4.1: Simplified block diagram of the proposed multipitch detection system. The iterative estimation/separation stage is delimited by the dashed line border.

the predominant note event. Additional pitch estimates, not included as part of the predominant event, are analysed further in order to detect other potentially real notes in the mixture which might be in a harmonic relationship with the predominant note event, hereafter referred to as interfering events. The pitch trajectories of the predominant event and all detected interfering events are linearly interpolated so they can be applied to the higher-rate note event separation spectrogram.

Once the predominant note event has been selected, its pitch contour is used to guide the separation of its spectral content from the current input mixture. The predominant event is then reconstructed from its separated magnitude spectrogram, while its associated energy is extracted from within the current input mixture, leading to the generation of a residual signal. Both the separated note event and its pitch trajectory are temporarily stored in memory for the rest of the iterative stage. If there is significant energy in the residual, and the maximum number of iterations has not been reached, then the cycle repeats with the residual as the new input signal.

When the iterative stage is complete, the full set of estimated pitch trajectories is revised in order to detect and remove potential outliers. All pitch trajectories passing this checking stage constitute the final set of fundamental frequency estimates of the original input mixture.

Sections 4.3.4 to 4.3.8 explain the processing stages involved in a single iteration, while Section 4.3.10 addresses the post-processing stage that takes place when the iterative part is complete.

4.3.3 Input Signal

The proposed system receives an input signal $x(t)$ which is assumed to be a single-channel instantaneous mixture of R individual audio sources, with sample values within the range $[-1 \ +1]$. If the original audio sources $s_r(t)$ are available, then the original mixture $\mathbf{x}(t)$ is generated according to the following equation.

$$\mathbf{x}(t) = \sum_{r=1}^R s_r(t) \quad (4.3)$$

Then, the input signal $x(t)$ is obtained by normalising the original mixture according to the following relation.

$$x(t) = \frac{0.9}{\max\{|\mathbf{x}(t)|\}} \mathbf{x}(t) \quad (4.4)$$

4.3.4 Salience Measurement

An initial set of pitch estimates is generated by running Duan's algorithm [118] on the lower-rate note event estimation spectrogram, using the same Hanning window and $\xi = \sqrt{5} \times 10^{-2}$ as the silence threshold. As a result, an initial array of pitch estimates is obtained for every frame in the decomposition, following the structure in (4.5).

$$\mathbf{P} = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,L} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,L} \\ \vdots & \vdots & \ddots & \vdots \\ p_{J,1} & p_{J,2} & \cdots & p_{J,L} \end{pmatrix} \quad (4.5)$$

where J is the number of levels and L is the total number of frames in the decomposition. The number of frames L is controlled by the duration of the input signal, and by the frame size and hop size specified within the multipitch detector, which are the same ones used by the rest of

the system. The number of levels J , or the number of rows in array \mathbf{P} , is controlled by Duan's multipitch estimator itself and it varies depending on the complexity of the input signal.

For each fundamental frequency estimate in \mathbf{P} , a salience measure is computed based on the spectral magnitude summation of their first Γ partial amplitudes, as defined by (4.6).

$$s_{(j,m)} = \sum_{\gamma=1}^{\Gamma} |\mathbf{X}(\gamma p_{(j,m)}, m)| \quad (4.6)$$

where $s_{(j,m)}$ is the salience of the j -th pitch candidate in the m -th frame, with fundamental frequency $p_{(j,m)}$, and $\mathbf{X}(p, m)$ is the complex spectrogram of the current input signal. The result is a matrix \mathbf{S} , shown in (4.7), that contains the salience measures for all pitch candidates in \mathbf{P} .

$$\mathbf{S} = \begin{pmatrix} s_{1,1} & s_{1,2} & \dots & s_{1,L} \\ s_{2,1} & s_{2,2} & \dots & s_{2,L} \\ \vdots & \vdots & \ddots & \vdots \\ s_{J,1} & s_{J,2} & \dots & s_{J,L} \end{pmatrix} \quad (4.7)$$

4.3.5 Predominant Note Event

Note events are detected by finding continuous segments of estimates, across all levels of \mathbf{P} , for which the change in fundamental frequency between adjacent frames is not higher than one semitone. The τ -th detected note event, which exists in level $j = j_\tau$, between frames m_a and m_b , has a total salience $\mathcal{S}(\tau)$ and duration $\mathcal{D}(\tau)$ defined as:

$$\mathcal{S}(\tau) = \sum_{m=m_a}^{m_b} s_{(j_\tau, m)} \quad (4.8)$$

$$\mathcal{D}(\tau) = m_b - m_a \quad (4.9)$$

If the total number of note events detected in a single iteration is denoted as T , their total salience and duration are normalised according to the following relations.

$$\hat{\mathcal{S}}(\tau) = \frac{\mathcal{S}(\tau)}{\sum_{i=1}^T \mathcal{S}(i)} \quad (4.10)$$

$$\hat{\mathcal{D}}(\tau) = \frac{\mathcal{D}(\tau)}{\sum_{i=1}^T \mathcal{D}(i)} \quad (4.11)$$

which means that the total salience and duration of all detected note events in the current iteration add to 1. Then, the predominance of the τ -th note event is defined as follows.

$$\mathcal{PD}_\tau = \hat{\mathcal{S}}(\tau) + \eta \hat{\mathcal{D}}(\tau) \quad (4.12)$$

where η is a parameter of the system, usually in the range from 0 to 1, that controls the influence of the duration on the selection of the predominant event. If $\eta = 0$, the decision is entirely based on total salience, while any $0 < \eta \leq 1$ could be used to encourage the selection of long but relatively weak note events as the predominant one. After extensive experimentation, the value $\eta = 0.5$ has been found to be effective and will be used in the rest of this work.

Each detected note event is expanded to encompass potential missallocated estimates in adjacent frames. For instance, if the τ -th note event is considered (defined in between frames m_a and m_b at level j_τ), which starts with estimate $p_{(j_\tau, m_a)}$, the expansion will first try to find a similar estimate in frame $m_a - 1$. If the difference between estimates $p_{(j_\tau, m_a - 1)}$ and $p_{(j_\tau, m_a)}$ is less than a semitone and the change in salience does not indicate a transition to a different note event, then the trajectory of the note event is expanded using estimate $p_{(j_\tau, m_a - 1)}$ while the starting frame is updated by taking $m_a = m_a - 1$. The expansion continues at both ends of the pitch trajectory until a clear note transition or onset/termination is detected. The predominance of each note event is updated every time its pitch trajectory is expanded with a new pitch estimate.

Among the detected and expanded note events, the one with the greatest predominance is selected as the strongest and most significant note event, and consequently it will be chosen for extraction in the current iteration.

To illustrate the selection of the predominant note event, an example is presented in Figure 4.2, for an input mixture consisting of violin and flute. The notes D \sharp 4 and B4 are played by the violin, while the flute plays a G4, as shown in Figure 4.2(a-b). During the first iteration of the system, an array of raw pitch estimates (\mathbf{P}) is generated with Duan's algorithm [118]. This initial set of estimates is presented in Figure 4.2(c), from which the contours of the real notes can be distinguished. However, these real contours are formed with estimates from different levels of \mathbf{P} , whilst an important number of outliers are also present, making it very difficult to recognise the total number of notes present and their relative volumes. To overcome this problem, the proposed strategy constructs a set of 18 note events from the raw estimates in \mathbf{P} , which are shown in Figure 4.2(d). Each of these events is now associated with a single continuous pitch trajectory, allowing a finer analysis of the input mixture and its components.

Duplicated note events are likely to occur after expanding continuous segments in different levels of \mathbf{P} , as shown in Figure 4.2(d), where two numbers appear under the contours of each real note. This particular feature of the algorithm does not represent a problem, since duplicates

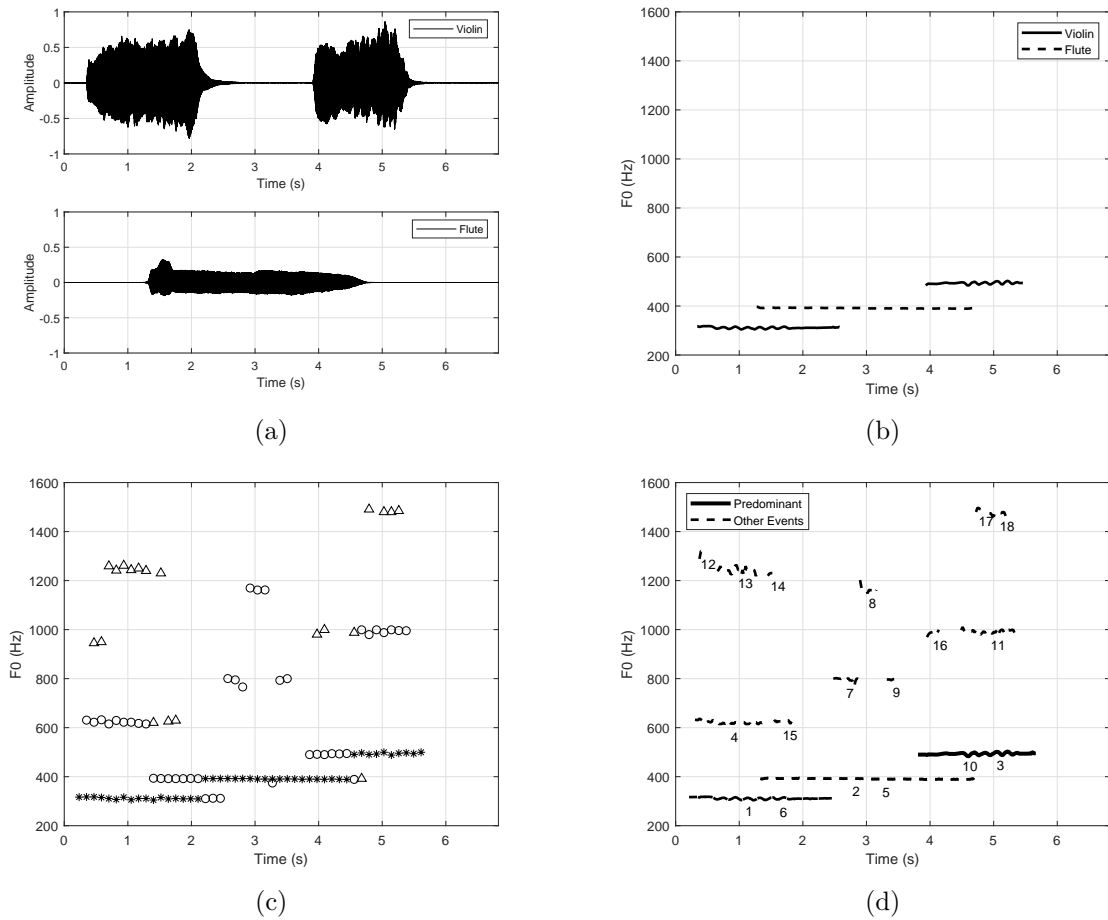


Figure 4.2: Selection of the first predominant note event in a mixture of violin and flute. The violin plays the notes D \sharp 4 and B4 while the flute plays a G4. (a) Original unmixed sources. (b) Ground-truth pitch trajectories. (c) Raw pitch estimates during the first iteration. Markers are used to identify estimates in the same level. (d) Predominant note event and other detected events during the first iteration.

are almost identical in length and they group the same set of pitch estimates. Hence, extracting one of the duplicates is usually equivalent to the extraction of both.

After computing the predominance of each note event according to Equation (4.12), the graph in Figure 4.3 is obtained. Here, six of the events have predominance above 0.15, which are the ones associated with the real notes in the mixture. From this group, event number 10 (violin B4) is the one with the highest predominance, hence it is selected as the predominant event in this iteration, and the energy within its harmonic structure will be the first to be extracted.

Since the predominance of note events is based on their salience and duration, weaker note events are allowed to compete against louder ones if they are relatively long. In Figure 4.3, events 2 and 5 are related with the flute note and they are also the ones showing the highest

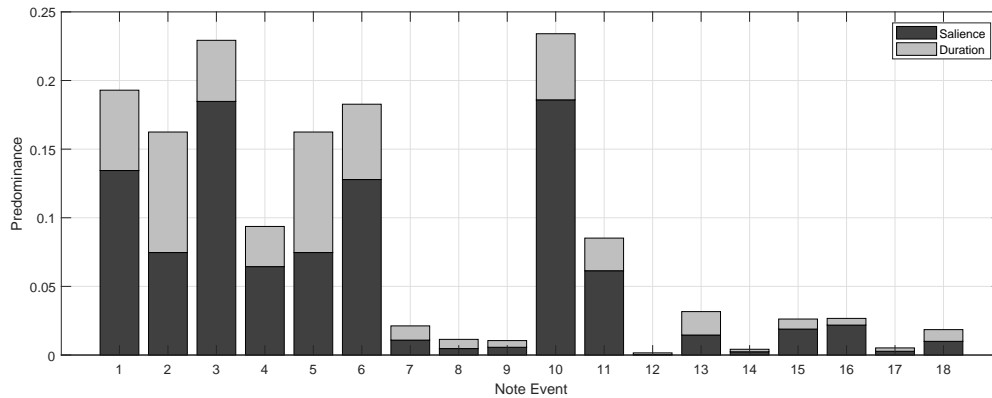


Figure 4.3: Predominance of note events during the first iteration of the system on a mixture of violin and flute. The contribution of the total saliency $\hat{S}(\tau)$ and duration $\eta\hat{D}(\tau)$ in the predominance of each note event, is shown using different shades in each of the vertical bars.

contribution from their duration. As these events are relatively weak, considering their duration in the predominance permits a better differentiation of these real events from other spurious ones having similar saliency levels, such as number 4 and 11.

Before presenting the proposed strategy for the extraction of the predominant note event, the concept of interfering events will be introduced in the following section, focussing on their importance in the detection of harmonically-related note events.

4.3.6 Interfering Events and Preservation Rates

In polyphonic music, different instruments may be playing notes in a harmonic relationship at the same time. This represents a challenge for any multipitch detector, due to the high level of overlap between harmonics of the underlying notes. To illustrate this problem, two audio excerpts are considered: a single violin playing the note A3 and a mixture of two violins playing the notes A3 and A4. From their magnitude spectra, presented in Figure 4.4, it is impossible to tell the number of instruments involved in each case. When the multipitch algorithm [118] is applied to these two excerpts, a set of pitch estimates is generated. The single violin, displayed in Figure 4.5(a), shows a continuous trajectory at the real pitch, while some outliers are included as errors at twice the fundamental frequency of the real note. Similarly, the mixture of two violins, presented in Figure 4.5(b), shows almost continuous trajectories at the underlying pitches and errors at integer multiples of the real pitches.

In both cases, the note A3 is the one with the highest predominance during the first iteration and hence it is selected as the predominant event. However, if the note A4 is actually present, the complete extraction of the predominant event also removes all of the A4 energy from the input

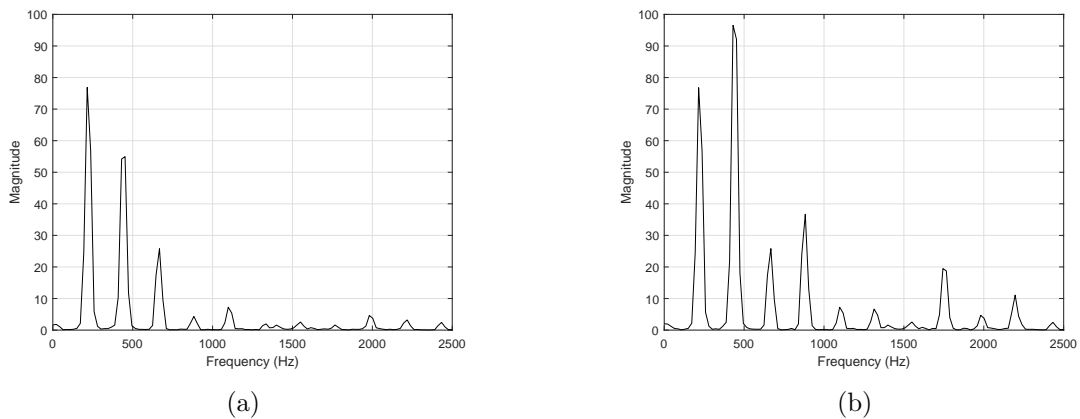


Figure 4.4: Absolute magnitude spectra of two different audio signals. (a) One violin playing the note A3. (b) Two violins playing notes A3 and A4. The spectra shown were computed using a frame size of 2048, 87.5% overlap, $f_s = 44.1$ kHz, no zero-padding, and a Hanning window.

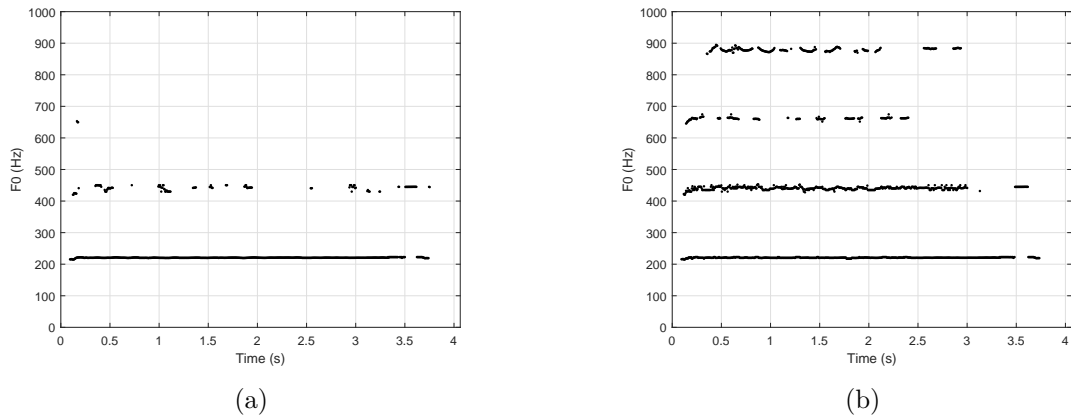


Figure 4.5: Pitch contours generated with Duan's algorithm. (a) One violin playing the note A3. (b) Two violins simultaneously playing the notes A3 and A4.

signal, making it impossible to detect the second violin during the next iteration. To avoid the extraction of multiple harmonically-related notes during a single iteration, the proposed algorithm uses all additional estimates provided by the original multipitch analysis to construct interfering events.

After selecting the predominant note event, all other note event candidates become potential interfering events. Their pitch contours are expanded by tracking their fundamental partials across the whole duration of the predominant note event, using the note event estimation magnitude spectrogram. Considering the same mixture of violin notes presented in Figure 4.5(b), the selected predominant note event (note A3) is presented in Figure 4.6(a), along with six other detected candidates, from which the interferers in Figure 4.6(b) are constructed. Notice that one

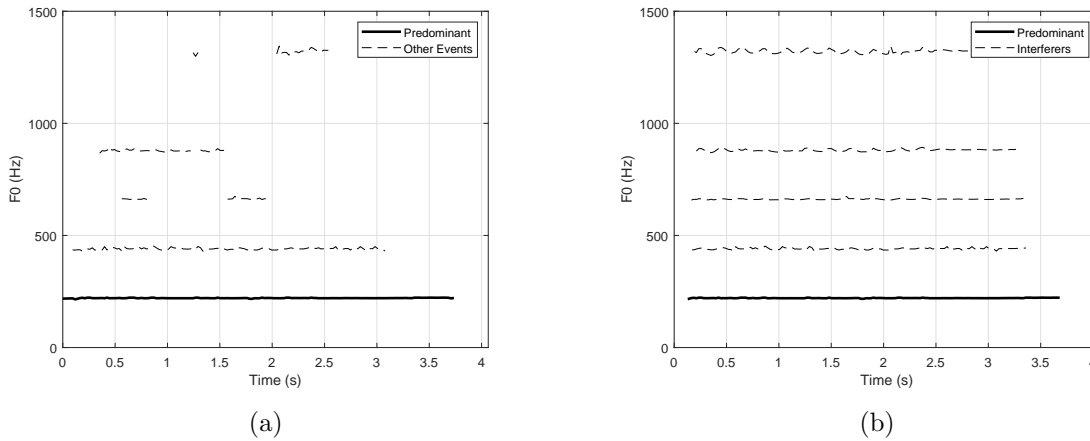


Figure 4.6: Analysis of a mixture consisting of two violin notes: A3 and A4. (a) Predominant note event selection from a set of eight candidates. (b) Predominant note event at 220 Hz and four interfering events: one real at 440 Hz and three spurious at 660 Hz, 880 Hz and 1320 Hz.

interferer points to the real note A4, while the rest are spurious events associated with higher harmonics.

If any harmonic partial of the predominant note event coincides with an interferer or with any of its overtones, some energy associated with these shared partials needs to be left in the residual after extraction, so that the underlying harmonically-related note can be detected in later iterations. However, in these particular cases, obtaining a suitable separation of the original note events is difficult, given that the original amplitudes and phase angles of the underlying components are not available. Moreover, real and spurious interfering events have to be correctly identified.

The proposed solution assigns a preservation rate (α) to each interfering event, ranging from zero up to an arbitrary maximum. If there is sufficient evidence that an interferer is an error (it does not associate with a real note present in the original mixture), zero preservation rate is assigned and all the energy of the selected partials is removed. If there is uncertainty, a non-zero preservation rate is assigned and some energy of the shared partials is left in the residual. The preservation rate depends on:

- The level of correlation between the pitch-scaled trajectories of the predominant note event and the interferer.
- The total salience associated with the interferer.
- The presence of multiple harmonically-related interferers.

If \mathcal{P}_{NE} and \mathcal{P}_{IE} denote the pitch trajectories of the predominant note event, with average fundamental frequency f_0 , and an interferer centred at a frequency nf_0 , respectively, their pitch-shifted trajectories are obtained as follows.

$$\mathcal{P}_{NE}^s = \frac{\mathcal{P}_{NE}}{f_0} \quad (4.13)$$

$$\mathcal{P}_{IE}^s = \frac{\mathcal{P}_{IE}}{nf_0} \quad (4.14)$$

Correlation between pitch-shifted trajectories is assessed by means of the p-value, while the total salience of an interferer is measured following the same procedure used with note events (described in Sections 4.3.4 and 4.3.5). The pitch-shifted contour of each detected interferer is compared with that of the predominant note event, and the maximum preservation rate is assigned if the resulting p-value is above 0.001, whilst zero preservation rate is used when the p-value is below 10^{-100} . Two additional bands are defined between these values to handle potential interferers that cannot be easily classified as either real notes or false positives. If the p-value of an interferer is within the range 10^{-100} to 10^{-40} , then the preservation rate is assigned using the following rule.

$$\alpha = \begin{cases} 0.05 & \text{if } \mathcal{S}_I < 0.5\mathcal{S}_P \\ 0.15 & \text{if } \mathcal{S}_I \geq 0.5\mathcal{S}_P \end{cases} \quad (4.15a)$$

$$(4.15b)$$

where \mathcal{S}_I and \mathcal{S}_P are the total salience of the interferer and the predominant note event, respectively. Also within this band, a preservation rate of 0.15 is used directly if the interferer itself has further potential interferers in harmonic relation. Interferers in this category are more likely to be outliers, so that low preservation rates are preferred. The second band is defined for interferers with a p-value in the range 10^{-40} to 0.001, and the preservation rate is assigned using two Sigmoid functions as follows.

$$\alpha = \begin{cases} \frac{A_{max} - 0.1}{1 + e^{0.8(7-\varepsilon)}} + 0.1 & \text{if } \mathcal{S}_I > 0.45\mathcal{S}_P \\ \frac{2(A_{max} - 0.1)}{1 + e^{0.8(7-\varepsilon)}} + 0.1 & \text{if } \mathcal{S}_I \leq 0.45\mathcal{S}_P \end{cases} \quad (4.16a)$$

$$(4.16b)$$

where A_{max} is the maximum preservation rate allowed and ε is a correlation measure based on the mean-squared error between the pitch-scaled contours of the predominant note event and

the interferer. Interferers in this category are more likely to be real notes. Finally, the maximum preservation rate is always assigned to the interferer if it is not in harmonic relation with the predominant note event.

The preservation rate basically defines the percentage of energy associated with each fully-overlapping partial that is not extracted during the current iteration. The percentage of energy that is not extracted as part of the predominant note event remains in the residual signal and, if it is still significant, then it can be detected as a different note event in later iterations. In this work, the maximum preservation rate has been set to 0.25 (or 25%), after experimenting with groups of integer-related notes played by several instruments, but a different value can be used depending on the likelihood of finding harmonically-related notes in the original mixture.

Considering the previous mixture of two octave-related violin notes, a total of four interfering events were detected during the first iteration as presented in Figure 4.6(b). One of these potential interferers associates with a real harmonically-related note at 440 Hz, while the rest are spurious events. The system has to analyse these interferers and set the preservation rates according to the likelihood of them being real harmonically-related notes. Figure 4.7 shows a comparison between the pitch-scaled contours of the interferers and that of the predominant note event, which are used to define the preservation rates by observing their correlation.

Figure 4.7(a) shows significant decorrelation between the contours, which are windowed to reduce boundary errors, meaning that the interferer has a high probability of being a real harmonically-related note. This hypothesis is reinforced by Figure 4.7(b,d), where the contours at integer multiples of 440 Hz also show little correlation. On the other hand, the contours in Figure 4.7(c) seem to be highly correlated, which means that the interferer might represent a pure overtone of the predominant note event. The algorithm, therefore, assigns preservation rates [0.25 0.25 0.15 0.25] to these interfering events, respectively, based on the corresponding p-values and mean-squared errors. These preservation rates would increase the chances of detecting the harmonically-related note at 440 Hz in the next iteration, while rejecting the spurious event at 660 Hz. Assigning the maximum preservation rates to interferers at 880 Hz and at 1320 Hz does not represent a significant problem, since they are both related to the real note at 440 Hz, meaning that its subsequent extraction on the following iteration would also remove most of the spectral energy at those frequencies, reducing the risk of false detections in later iterations.

It is important to appreciate the difficulty of distinguishing real interfering events from false positives; the proposed rules are a partial solution which has been designed after extensive experimentation on several groups of integer-related notes played by different musical instruments.

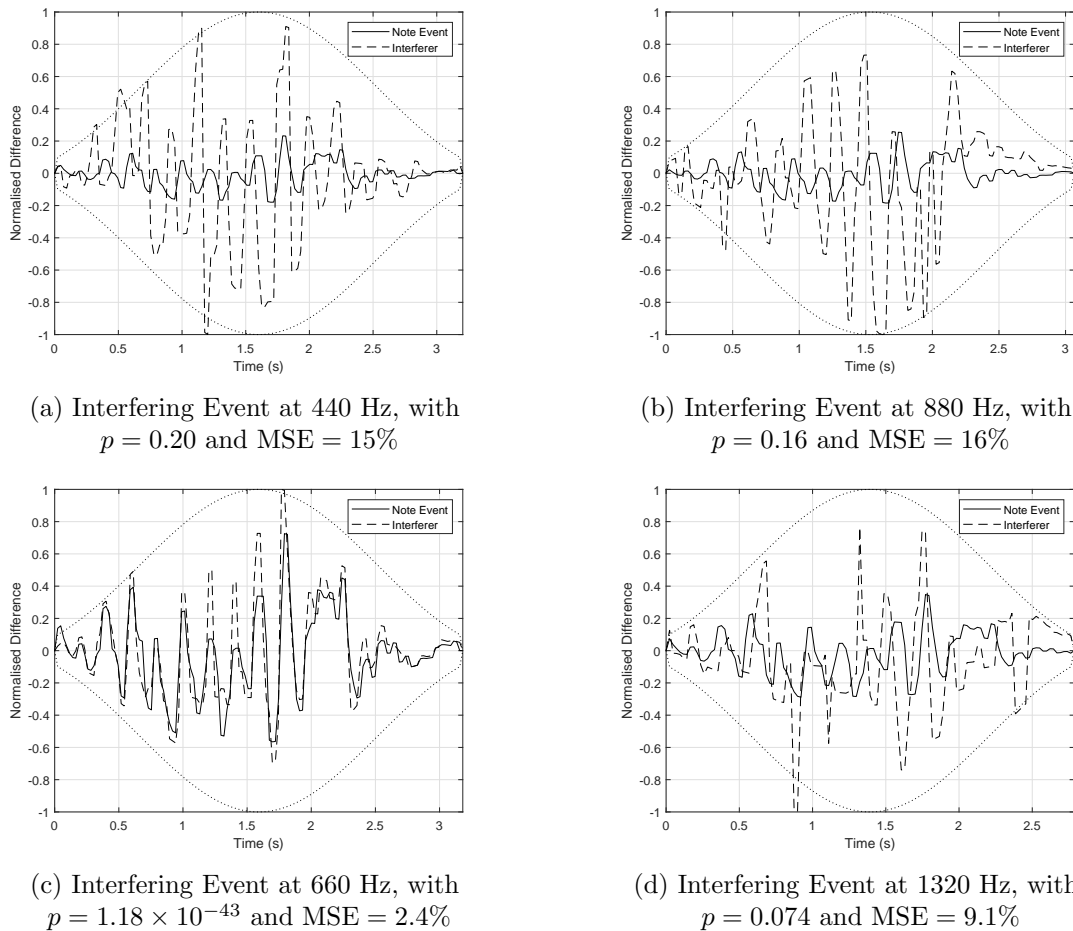


Figure 4.7: Comparison between the pitch-scaled contours of four potentially interfering events detected during the first iteration of the system on a mixture of two octave-related violin notes.

The parameters and ranges presented in this section have been empirically selected based on these experiments, in order to favour the detection of real harmonically-related note events and hence increase the accuracy of the system. Moreover, distortion affecting pitch contours increases with every iteration and makes the p-value more unreliable with each extraction. However, the proposed strategy has been shown to be useful in detecting up to three real notes played in near-harmonic relation.

4.3.7 Interpolation and Boundary Correction

Pitch trajectories are initially estimated using 50% overlap between frames, mostly to reduce computational costs, but also to lower the risk of detecting incomplete note events due to undetected estimates in some of the frames. Considering that the baseline algorithm [118] can also be affected by masking and distortion associated with phase interaction, it is possible that some real pitch estimates will not be detected in some of the frames, which could force the

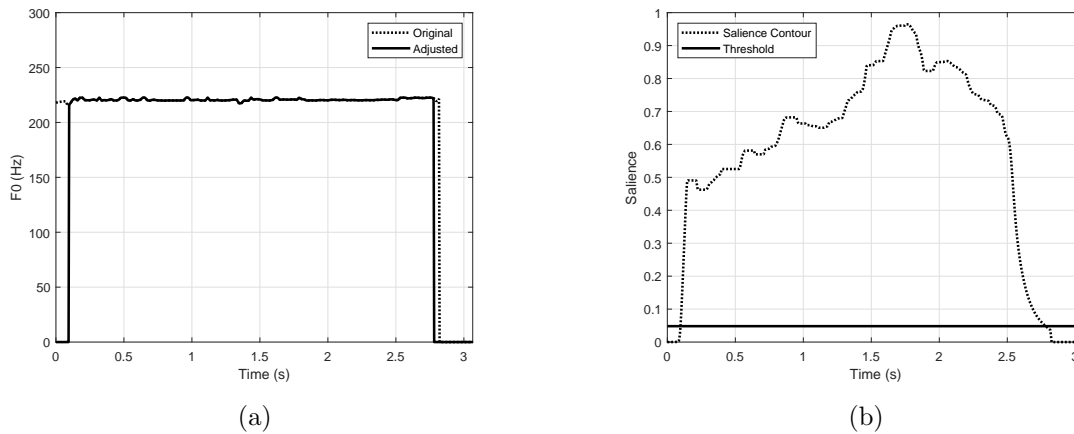


Figure 4.8: Boundary correction. (a) Original and adjusted pitch trajectory of the predominant note event in a mixture of two violin notes. (b) Saliency contour and significance threshold at 5% of the maximum for the same mixture of two violin notes.

proposed algorithm to detect only sections of a particular real note event. However, if the same overlap rate is used to separate the predominant note event, it can lead to significant distortion in the reconstructed signals. Hence, to keep the quality of the separation high, pitch trajectories are linearly interpolated so that an overlap of 87.5% can be used during separation of the predominant note event.

At the same time, the boundaries of each trajectory are rechecked to guarantee that each of the estimates is targeting meaningful energy inside the current input signal. The saliency of each estimate is recalculated and arranged into a saliency contour, which is normalised according to its maximum, and which represents an approximation to the intensity of the note event across time. A significance threshold is empirically defined at 5% and if the saliency of a pitch estimate is lower than the threshold, it is considered to be irrelevant and therefore, the estimate is removed from the predominant trajectory. Figure 4.8 illustrates the boundary adjustment of the predominant note event from Figure 4.6(a). A small number of estimates, located at the start and at the end of the trajectory, are rejected since the levels of saliency associated with them are below the significance threshold. Before starting the separation stage, interfering events also undergo boundary correction, using the same process and significance threshold.

4.3.8 Separation of Spectral Content

The spectral content of the predominant note event is separated frame by frame. Its pitch trajectory indicates the predominant fundamental frequencies, while interferers contain other concurrent pitches that might be present and must be considered. In every frame, all harmonic

frequencies associated with the predominant and potential interfering pitches are computed up to the Nyquist frequency, and a set of spectral peaks is identified in the magnitude spectrum by finding the local maximum in the neighbourhood of each predominant harmonic frequency. The distance between any of these peaks and any possible interfering frequency is measured and used to detect potential overlapping peaks. When the minimum distance between the spectral peak and any interferer frequency is greater than a threshold (set as 30 Hz corresponding to the frame size of 2048 used in this study, with bin spacing of 21.5 Hz at $f_s = 44.1$ kHz), the peak is classified as a semi-overlapping partial, otherwise it is assumed to be a fully-overlapping partial.

Considering the importance of the separation stage within the proposed system, it is presented as a separate chapter in this thesis (Chapter 5), where its main characteristics and internal algorithms are discussed in detail. Chapter 5 also introduces an additional clustering stage, in which the end-user is allowed to group a set of separated note events, based on their sounds and pitch trajectories, to form individual sources. The separation performance of this semi-supervised approach is evaluated on several audio mixtures, while results are compared with a similar method.

So far, the discussion of the proposed multipitch estimator has focused on its iterative stage, where note events are detected and extracted from within an input mixture. Section 4.3.9 describes the convergence criteria that stop the iterative stage, while Section 4.3.10 presents the post-processing stage, which takes place after the iterative stage is complete, with the aim of removing spurious events from the final set of pitch trajectories.

4.3.9 Convergence of the Iterative Stage

The iterative estimation and extraction of note events stops when a maximum number of iterations, hereafter denoted as Θ , is reached or when the energy in the residual is below a significance threshold T_C , which is calculated as follows.

$$T_C = \lambda E_{mix} \quad (4.17)$$

where E_{mix} is the energy of the original input mixture and λ is a system parameter in the range $]0, 1[$ that controls the amount of energy to be extracted from the original mixture. Parameter Θ can be adjusted by the end-user to match the number of notes in the mixture, if that information is available. An appropriate value for λ can be selected depending on the complexity of the audio mixture and the melodies being played by the underlying sources. If the polyphony of

the mixture is high and several notes with short durations are present, then a low λ is preferred ($0 < \lambda < 0.05$). Otherwise, a higher λ can be used to reduce computation times.

4.3.10 Power-based Revision

After the iterative stage has been completed, all extracted note events and their pitch trajectories are further analysed to detect and remove outliers. The proposed algorithm can generate spurious detections which are mostly associated with non-harmonic sources present in the mixture, noisy content, or other interfering events that were not properly classified. The power of each extracted note event is used as a criterion to distinguish real events from outliers. Considering the q -th note event, with average pitch $f_{(0,q)}$, its power can be defined as follows.

$$\mathcal{P}_q = \begin{cases} \frac{1}{L_q} \sum_{n=1}^N |\text{NE}_q(n)|^2 & \text{if } f_{(0,q)} > 200\text{Hz} \\ \frac{10}{L_q} \sum_{n=1}^N |\text{NE}_q(n)|^2 & \text{if } f_{(0,q)} \leq 200\text{Hz} \end{cases} \quad (4.18a)$$

$$\quad (4.18b)$$

where L_q is the number of non-zero fundamental frequency estimates within the q -th pitch contour, and $\text{NE}_q(n)$ is the separated time-domain signal associated with it. The energy of low-pitched events is deliberately amplified (10 times according to Equation 4.18b) to avoid rejecting real notes that lost most of their overtones during the previous extraction of harmonically-related notes in the same region. In these particular cases, the only surviving partial of a low-pitched note might be just the fundamental, while the energy associated with its higher harmonics has already been extracted in previous iterations. If this fundamental belongs to a real note event, the amplification might compensate for the loss of its overtones and allow the detection of other heavily masked note events.

The energy content of all detected note events is normalised according to the most energetic one. If the number of note events detected in a particular iteration is Θ , then the power of each detected note event is normalised as follows.

$$\mathcal{P}_q^n = \frac{\mathcal{P}_q}{\max_{i \rightarrow 1:\Theta} \{\mathcal{P}_i\}} \quad (4.19)$$

Finally, the set of accepted pitch trajectories is generated by identifying all events with normalised power above a significance threshold T_E . While it is possible to define T_E to be an adaptive threshold, it has been found that a fixed value is more effective and less dependent

on the actual distribution of the musical notes in the mixture or the order in which they are extracted.

4.4 Evaluation of Performance

4.4.1 Methodology

The performance of the proposed algorithm is evaluated in three different experiments, which have been designed to assess three key aspects of the process. The first experiment uses several single-note sources to evaluate accuracy for different levels of polyphony. The second introduces musical notes in integer frequency relationships to assess the algorithm’s capacity to distinguish real events from false detections. The third experiment uses the well-established Bach10 dataset [118] to evaluate the accuracy of the algorithm on real polyphonic music. Results for each experiment are compared with the original unsupervised multipitch estimator presented in [118], hereafter referred as DUAN, with frame size of 2048 samples, hop size of 256 samples, silence threshold $\xi = \sqrt{5} \times 10^{-2}$, and a Hanning window function. The following two variations of the proposed iterative method are defined, based on the strategy used to extract the predominant event from within the mixture.

- **IES-TFM:** Iterative system that uses the extraction algorithm described in Section 5.6.1, which is based on time-frequency masking.
- **IES-TDS:** Iterative system that uses the extraction algorithm described in Section 5.6.2, which is based on time-domain subtraction.

Both of them are applied using the following general settings: $\Gamma = 5$, $A_{max} = 0.25$, $\Theta = 45$, $\lambda = 0.0016$, $T_E = 0.01$, $H_{max} = 30$ and $H_{min} = 10$. Parameters H_{max} and H_{min} are defined in Section 5.6.4. The sampling frequency of all audio excerpts used in this section is 44.1 kHz.

In every experiment, a set of independent ground-truth pitch trajectories was obtained by applying the proposed system to each source in isolation. For the Bach10 dataset, the resulting contours were further examined and adjusted whenever necessary so that the resulting contours could be as similar as possible to the original ground-truth trajectories supplied with the dataset. Multipitch estimation performance is evaluated here by means of the F-Score, in which the accuracy (A), precision (P) and recall (R) are computed as defined in [111], with a tolerance of 3% or quarter-tone (in accordance with MIREX), and reported as a percentage (%).

Table 4.1: Simultaneous violin notes contained in each test mixture.

| Polyphony | Combined Violin Notes |
|-----------|----------------------------------|
| 2 | F5 and Ab5 |
| 3 | F5, Ab5 and A5 |
| 4 | F5, Ab5, A5 and B5 |
| 5 | F5, Ab5, A5, B5 and Db6 |
| 6 | F5, Ab5, A5, B5, Db6 and E6 |
| 7 | F5, Ab5, A5, B5, Db6, E6 and Gb6 |

Table 4.2: F-Scores for multipitch estimation on seven mixtures of simultaneous sustained single-note violin sources. (A) Accuracy, (P) Precision and (R) Recall.

| Polyphony | IES-TFM | | | IES-TDS | | | DUAN | | |
|-----------|---------|-------|------|---------|-------|------|------|------|------|
| | A | P | R | A | P | R | A | P | R |
| 2 | 98.5 | 100.0 | 98.5 | 98.5 | 100.0 | 98.5 | 70.0 | 70.5 | 99.0 |
| 3 | 98.8 | 100.0 | 98.8 | 98.7 | 100.0 | 98.7 | 80.5 | 83.3 | 96.0 |
| 4 | 98.4 | 100.0 | 98.4 | 98.5 | 100.0 | 98.5 | 76.5 | 81.0 | 93.3 |
| 5 | 98.5 | 99.9 | 98.6 | 98.4 | 99.9 | 98.6 | 79.4 | 86.9 | 90.1 |
| 6 | 98.3 | 99.8 | 98.5 | 98.1 | 99.8 | 98.4 | 66.5 | 83.9 | 76.3 |
| 7 | 98.0 | 99.7 | 98.3 | 98.6 | 99.8 | 98.7 | 61.1 | 83.8 | 69.2 |

4.4.2 Seven Simultaneous Violins

In this experiment, the proposed algorithm is tested on a set of seven audio mixtures containing different numbers of single-note sources playing simultaneously³. These sources are violin notes, recorded in anechoic conditions, with fundamental frequencies in the range 700 Hz to 1600 Hz. With ascending order, the notes involved are F5, Ab5, A5, B5, Db6, E6 and Gb6. The violin notes involved in each of the seven test mixtures are shown in Table 4.1.

Performance measures obtained by each of the methods are presented in Table 4.2, for polyphonies two to seven, while the estimated pitch contours for the highest polyphony are presented in Figure 4.9, where numbers indicate the order of extraction.

Results show two important advantages of the proposed system. First, the iterative estimation of note events reduces the complexity of the audio mixture in every iteration, allowing the detection of additional sources. Second, note events are continuous sets of pitch estimates, which is a useful feature for note tracking. The high levels of accuracy observed in Table 4.2 are due to the high fundamental frequencies of the notes involved, so that each of them has widely spaced partials, which can be well separated using the proposed strategies. Duan’s joint algorithm, on the other hand, struggles to detect all real pitches due to the spectral complexity as the polyphony of the mixture increases. Also in Figure 4.9, it is important to notice that

³<https://doi.org/10.5281/zenodo.3478442>

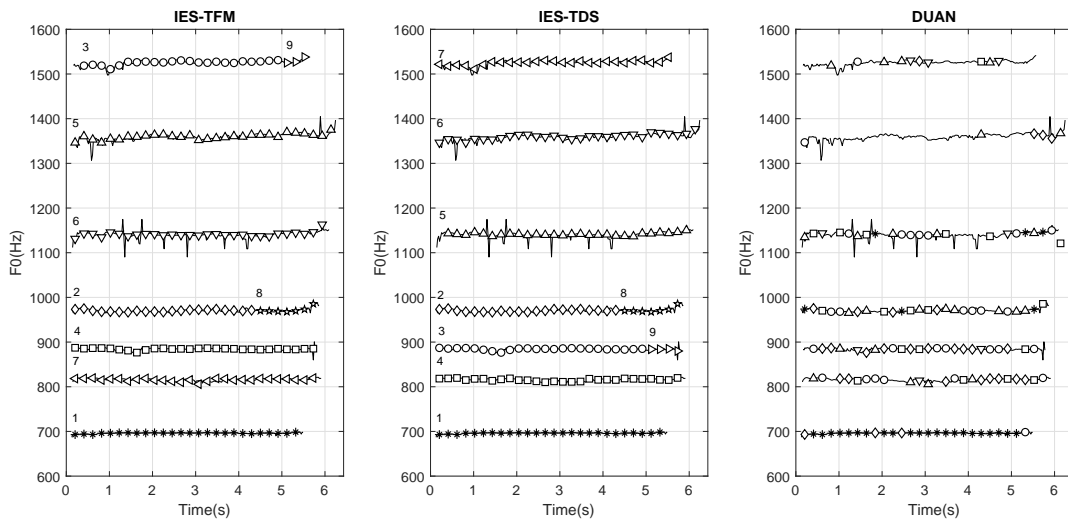


Figure 4.9: Estimated pitch trajectories for a test mixture consisting of seven simultaneous violin notes. Markers identify pitch estimates of a particular note event, while numbers (IES-TFM and IES-TDS) indicate the order of extraction. Solid lines represent the ground-truth pitch contours.

both variations of the proposed system delivered two of the real pitch contours as two different note events (2-8 and 3-9 in both cases). The reason for this is that source interaction was still very high during iterations 2 and 3, and several estimates in the release sections were heavily masked. However, it was possible to detect these additional estimates in later iterations when the rest of the musical notes were already removed.

Figure 4.9 also shows the way in which the strategy used to extract the spectral content of note events alters the order in which their pitch trajectories are estimated. Considering the mixture of seven violin notes, both variations of the proposed system started the process by first detecting and extracting the note F5, then the note B5 was selected and partially extracted in both cases. However, during the third iteration, IES-TFM selected the note G \flat 6 as the next one, while IES-TDS chose the note A5, subsequently changing the extraction order for the rest of the notes. The reason for this depends on the fact that each method extracts energy from the original input mixture in slightly different ways, influencing the salience measurements for later iterations.

4.4.3 Notes in Harmonic Relation

This experiment is designed to evaluate the performance of the proposed system when different numbers of harmonically-related musical notes are simultaneously playing in an audio mixture. Here, fifteen groups of musical notes approximately in harmonic relation were chosen

Table 4.3: Details of selected notes in harmonic relation.

| Set | Average f_0 (Hz) | Instrument |
|-----|----------------------|-------------|
| 1 | 220, 440, 660, 880 | Viola |
| 2 | 196, 393, 587, 788 | Violin |
| 3 | 147, 294, 443, 587 | Cello |
| 4 | 124, 248, 373, 499 | Piano |
| 5 | 210, 411, 620, 830 | Soprano Sax |
| 6 | 110, 221, 331, 443 | Bassoon |
| 7 | 262, 527, 794, 1066 | Flute |
| 8 | 333, 665, 1000, 1330 | Recorder |
| 9 | 230, 465, 698, 926 | Oboe |
| 10 | 99, 198, 298, 398 | Horn |
| 11 | 141, 279, 417, 555 | Alto Sax |
| 12 | 148, 297, 444, 592 | Pipe Organ |
| 13 | 104, 208, 310, 415 | Harp |
| 14 | 166, 333, 502, 672 | Trumpet |
| 15 | 198, 397, 596, 792 | Clarinet |

Table 4.4: F-Scores for multipitch estimation on 15 mixtures of simultaneous harmonically-related notes. (A) Accuracy, (P) Precision and (R) Recall.

| Polyphony | IES-TFM | | | IES-TDS | | | DUAN | | |
|-----------|---------|-------|-------|---------|-------|-------|-------|-------|-------|
| | A | P | R | A | P | R | A | P | R |
| 2 | 66.18 | 74.76 | 86.76 | 68.29 | 76.77 | 88.04 | 69.20 | 76.09 | 90.99 |
| 3 | 67.11 | 79.32 | 81.63 | 73.45 | 84.91 | 84.92 | 68.73 | 83.42 | 81.66 |
| 4 | 69.59 | 89.69 | 75.95 | 71.53 | 92.64 | 77.04 | 66.20 | 87.67 | 74.13 |

from the RWC instrument database [129] (Table 4.3). The selected test recordings are available online⁴. In every group, three audio mixtures were created by combining the lowest-pitched note with one or more higher ones following the same pattern used in the previous experiment (Table 4.1), in order to produce polyphonies 2, 3 and 4. Performance measures after multipitch estimation on these audio mixtures, corresponding to both methods, are presented in Table 4.4 for each level of polyphony.

The results show that the overall performance exhibited by the proposed systems is similar to that of Duan’s algorithm, despite the high complexity of the task, with the additional advantage of delivering each trajectory as a group of continuous note events. Figure 4.10 shows the estimated pitch trajectories corresponding to the third mixture in set seven, which consists of four flute notes (C4, C5, G5, and C6).

When note events are in harmonic relation, the order in which they are extracted from within the mixture has a significant impact on the final accuracy of the estimation. Extracting the lower

⁴<https://doi.org/10.5281/zenodo.3478465>

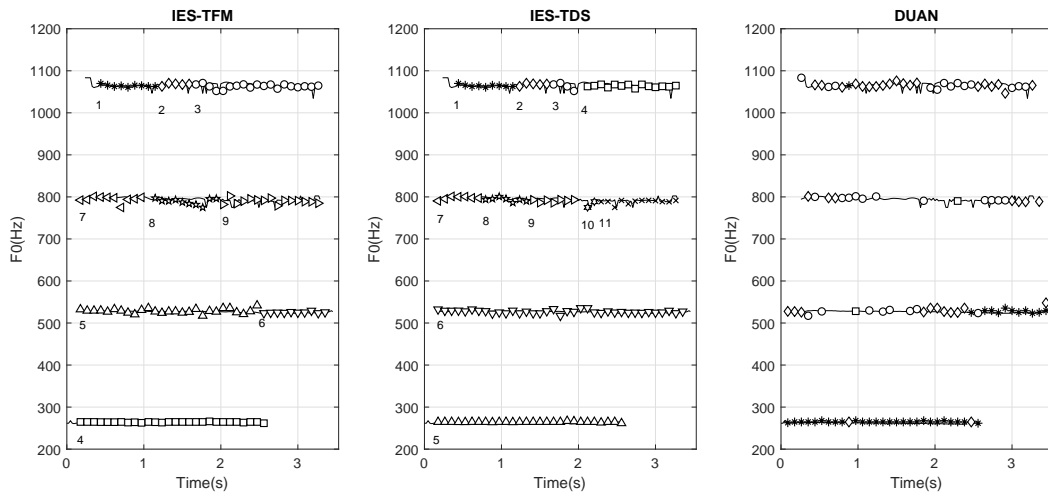


Figure 4.10: Estimated pitch trajectories for a test mixture from set 7, consisting of four simultaneous flute notes (C4, C5, G5, and C6). Markers identify pitch estimates of a particular note event, while numbers (IES-TFM and IES-TDS) indicate the order of extraction.

pitched note first is usually more difficult than otherwise. In this situation, the correct detection of any additional note relies on identifying the right set of interferers and on the appropriate setting of their preservation rates, which is quite demanding even in low polyphonies.

The observed levels of accuracy also show some correlation with the type of instrument under analysis, as shown in Table 4.5, where bowed string instruments perform better than plucked ones, and results for woodwind instruments are more stable than those for brass instruments. High levels of inharmonicity, which are common in piano notes, represent an important source of error for the proposed algorithm, since they make it very difficult to identify the harmonic structure associated with the corresponding note in every frame, leading to an incomplete extraction of its energy. Additionally, plucked notes are usually detected as note events which are significantly shorter than the duration of the real notes, due to their long decays which are not completely tracked by the algorithm.

4.4.4 Real Music

Performance evaluation on real music is conducted using the Bach10 dataset [118]⁵, which contains ten excerpts from four-part chorales by J. S. Bach, performed by an ensemble of four instruments: violin, clarinet, tenor saxophone and bassoon. All tracks included are single-channel recordings, containing musical notes in the range F2 to A5, with an overall duration of 5 minutes and 30 seconds. For simplicity, the whole dataset is divided into 50 sections, with

⁵<http://www2.ece.rochester.edu/projects/air/resource.html>

Table 4.5: F-Scores for multipitch estimation on several audio mixtures consisting of different numbers of harmonically-related notes played by four types of instruments.

| Type | Poly. | IES-TFM | | | IES-TDS | | | DUAN | | |
|----------------|-------|---------|-------|-------|---------|-------|-------|-------|-------|-------|
| | | A | P | R | A | P | R | A | P | R |
| Plucked String | 2 | 24.58 | 72.41 | 32.19 | 27.35 | 75.63 | 35.10 | 57.30 | 82.54 | 73.75 |
| | 3 | 34.15 | 84.87 | 41.56 | 36.06 | 88.77 | 41.33 | 62.76 | 91.43 | 70.32 |
| | 4 | 24.64 | 84.19 | 28.49 | 30.19 | 94.02 | 31.89 | 56.39 | 93.54 | 61.13 |
| Bowed String | 2 | 80.41 | 81.62 | 98.62 | 80.37 | 81.56 | 98.66 | 69.32 | 69.91 | 98.66 |
| | 3 | 88.20 | 90.19 | 97.87 | 88.45 | 90.43 | 97.87 | 74.02 | 78.79 | 92.41 |
| | 4 | 87.62 | 97.50 | 89.58 | 81.75 | 91.10 | 88.93 | 76.10 | 86.46 | 86.13 |
| Woodwind | 2 | 85.17 | 89.25 | 92.68 | 86.68 | 90.07 | 95.52 | 86.01 | 89.93 | 94.89 |
| | 3 | 80.66 | 89.44 | 86.34 | 80.47 | 89.52 | 86.07 | 74.68 | 91.69 | 81.59 |
| | 4 | 75.03 | 99.05 | 75.64 | 73.12 | 99.29 | 73.62 | 70.75 | 93.59 | 75.69 |
| Brass | 2 | 55.29 | 57.09 | 95.57 | 59.03 | 61.04 | 95.36 | 56.98 | 63.39 | 89.39 |
| | 3 | 54.08 | 60.47 | 83.20 | 72.38 | 75.44 | 93.44 | 61.99 | 74.73 | 79.82 |
| | 4 | 71.32 | 77.83 | 87.06 | 80.35 | 86.36 | 91.38 | 59.64 | 80.14 | 70.58 |

similar durations, while performance is evaluated here in three different levels of polyphony: 2, 3, and 4. Results obtained for each case are presented in Table 4.6, and examples of estimated pitch trajectories are presented in Figure 4.11, for three different excerpts with polyphony 4.

The accuracy levels reported show that both variations of the IES algorithm outperformed DUAN in all polyphony levels investigated, evidencing that the proposed iterative estimation/separation strategy, in which DUAN is used as a starting point, allows the detection of additional true pitches that could not be detected during a single-pass DUAN estimation, reducing at the same time the number of outliers. When the musical notes involved exhibit relatively high pitches (above 200 Hz or notes higher than G3), the IES system delivered very high quality trajectories in most cases.

However, the introduction of low-pitched notes increased the spectral complexity, and consequently reduced the quality of estimation, given the difficulties associated with the separation of harmonic partials with very little distance between their centre frequencies. The influence of low-pitched notes on the overall accuracy of the estimation can be seen in Table 4.6. In polyphony 2, the highest accuracy was obtained on a mixture of violin and clarinet, since these two instruments are playing almost always above G3, while the lowest accuracy was reported in a mixture of saxophone and bassoon where a significant number of notes below G3 can be found. A similar pattern can also be observed in polyphony 3, where the incorporation of the saxophone and the bassoon represented a notorious reduction of the overall accuracy of the estimation.

Table 4.6 also shows an interesting case in polyphony 2, where the accuracy of the proposed algorithms, in a mixture of saxophone and bassoon, was even lower than the accuracy of the

Table 4.6: F-Scores for multipitch estimation on the Bach10 dataset. Instruments: (V) violin, (C) clarinet, (S) saxophone, and (B) bassoon.

| Poly. | Instru. | IES-TFM | | | IES-TDS | | | DUAN | | |
|-------|---------|---------|-------|-------|---------|-------|-------|-------|-------|-------|
| | | A | P | R | A | P | R | A | P | R |
| 2 | V-C | 89.07 | 94.94 | 93.39 | 89.76 | 95.73 | 93.42 | 70.38 | 74.18 | 92.90 |
| | V-S | 84.75 | 88.42 | 95.44 | 85.51 | 89.28 | 95.37 | 73.90 | 75.57 | 97.09 |
| | V-B | 75.23 | 79.49 | 93.24 | 77.24 | 81.36 | 93.88 | 67.90 | 70.84 | 94.12 |
| | C-S | 87.87 | 91.25 | 95.98 | 88.76 | 92.21 | 95.92 | 79.62 | 85.26 | 92.10 |
| | C-B | 81.34 | 88.58 | 90.66 | 83.11 | 90.47 | 90.96 | 77.74 | 94.01 | 81.75 |
| | S-B | 73.43 | 77.72 | 92.57 | 74.75 | 79.09 | 92.98 | 70.25 | 78.56 | 86.26 |
| 3 | V-C-S | 87.55 | 92.65 | 94.07 | 87.62 | 92.89 | 93.90 | 76.83 | 84.69 | 89.06 |
| | C-S-B | 77.37 | 86.98 | 87.45 | 77.58 | 87.14 | 87.58 | 66.48 | 87.30 | 73.56 |
| | V-C-B | 80.39 | 87.29 | 90.88 | 80.56 | 87.62 | 90.84 | 70.88 | 83.56 | 82.23 |
| | V-S-B | 74.72 | 81.26 | 90.02 | 75.68 | 81.98 | 90.56 | 68.52 | 79.71 | 82.75 |
| 4 | V-C-S-B | 76.79 | 87.45 | 86.22 | 78.19 | 88.73 | 86.79 | 64.61 | 86.16 | 72.04 |

systems in polyphony 4. In this particular case, the presence of the saxophone and bassoon, both playing a large number of notes below G3, triggered the detection of a large number of interfering events, many of which were false positives. Since the preservation rates were not always appropriately assigned, many of these interfering events were extracted as real note events, increasing the total number of outliers and hence, reducing the accuracy of estimation.

Despite the limitations, the accuracy of the proposed systems is still high and comparable with that reported for an informed multipitch estimator [127], where the highest accuracy on the same dataset is 80% for polyphony 4. The method in [116] reported an accuracy of 85.5% for the same mixture with polyphony 4, but without the IES note tracking capabilities.

Results from these experiments also evidence a number of limitations that have been classified into five groups, as follows.

- **Incorrect Selection of Note Event Boundaries.** The presence of a high number of simultaneous musical notes and transitions increases the difficulties of detecting their boundaries correctly, resulting in some note events being longer or shorter than the real ones. Some examples are presented in Figure 4.11(a), where note events 1, 2, 3 and 22 are clearly shorter than the true notes, while note event 21 is slightly longer than the real one.
- **Same Pitch Concurrent Notes.** Two different sources playing the same note at the same time are detected as a single-source event. For example, in Figure 4.11(c), two notes played by the violin and clarinet have been detected as note event number 4, between $t = 4$ s and $t = 4.5$ s, but the algorithm cannot recognise the presence of two simultaneous notes with the same pitch, hence, only one trajectory is presented.

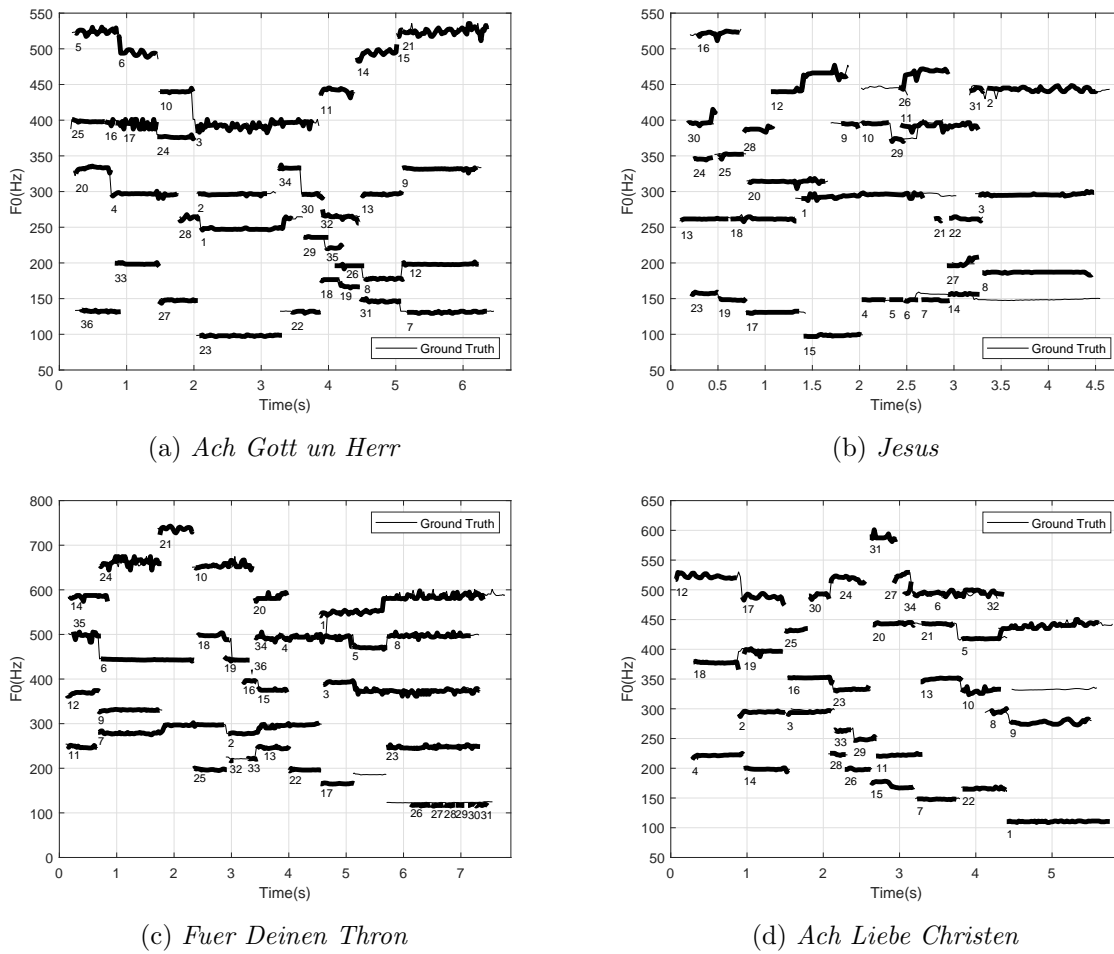


Figure 4.11: Pitch trajectories of the note events detected with the IES-TDS algorithm on four excerpts with polyphony 4 of the Bach10 database. Numbers indicate the extraction order while the ground-truth pitch trajectories are marked with solid lines.

- Note Events Missed by Duan’s Algorithm.** Real note events cannot be detected if Duan’s multipitch estimator is unable to deliver at least some of its estimates during the iterative stage. An example can be observed in Figure 4.11(b), between $t = 3.7$ s and $t = 4.5$ s. Here, the note at 150 Hz was not detected in any of the iterations, even after removing the higher notes (events 2, 3 and 8).
- Inadequate Extraction Order.** The order of extraction significantly influences the final accuracy of the system. An interesting case is presented in Figure 4.11(c), between $t = 5.8$ s and $t = 7.3$ s. Note event 23 is not real; it is a spurious candidate generated by the second harmonic of the real note at 120 Hz. Since the fundamental of the real note is weak, its second harmonic was chosen as a predominant event before the real note was extracted, appearing at the end as a real event with significant energy content. If the real

note event at 120 Hz is extracted first, then it would extract the energy of the spurious event at 240 Hz, avoiding its detection in later iterations. The way in which the extraction order is determined relies on the total salience and duration of the candidates in any given iteration, and the risk of extraction is not considered before selecting the predominant note event. Hence, this type of error cannot be completely avoided.

- **Inadequate Separation of Spectral Content.** The separation of note events can introduce damage to other harmonic structures nearby, making it harder to detect them in later iterations. Figure 4.11(b) shows an interesting case between $t = 2.5$ s and $t = 3$ s. Note event 7 has been misleadingly estimated as having fundamental frequency at 150 Hz, while the real note is slightly higher. The extraction of the incorrect note damaged the tail of a higher note at 300 Hz, which could not be detected during later iterations.
- **Harmonically-related Note Events:** It has been observed that note events in harmonic relation can be missed if the corresponding interferers are not properly detected and handled. In Figure 4.11(b), between $t = 2$ s and $t = 2.5$ s, it can be seen that the note at 450 Hz was not detected. During the separation of several previous events at 150 Hz, the preservation rate for the interferer at 450 Hz was set too low and most of its energy was removed with the events at 150 Hz, leaving almost no traces to be found in later iterations. A different case is presented in Figure 4.11(d), between $t = 4.5$ s and $t = 5.6$ s, where a set of three harmonically-related notes are present with fundamental frequencies 110 Hz, 330 Hz and 440 Hz. During the first iteration, the note at 110 Hz is extracted while the other two are detected as interferers and the maximum preservation rate is assigned to them. The next one to be extracted is the note at 440 Hz, during the fifth iteration. When the note at 330 Hz is finally detected, most of its energy has already been misallocated into the two previously extracted notes. Hence, it appears as a very weak note event and the algorithm rejects it assuming that it represents a spurious detection.

4.5 Summary

In this chapter, an iterative approach for multipitch estimation was proposed, based on identification and separation of note events. In every iteration, the method considers an initial set of fundamental frequency estimates, from which the pitch contour of the predominant note event is selected, and used to estimate its spectral content. Then, the predominant event is separated from within the current input mixture, following two different extraction methods,

while the residual is used as the input for the following iteration. Convergence occurs when no significant energy is left in the residual or when a predefined maximum number of iterations is reached.

This approach is fully unsupervised and does not require any prior knowledge of the input signal, so it can be applied to a wide range of music containing harmonic or nearly-harmonic sources. The fact that note events are detected as continuous segments of estimates suggests the potential use of the algorithm as the basis for note tracking and multipitch streaming systems.

Evaluation of performance was conducted in three different scenarios, and the proposed algorithm outperformed a well-known probabilistic approach, showing significant advantages in terms of detecting heavily masked notes while reducing the number of spurious estimates. The results also revealed a correlation between the fundamental frequency of the underlying notes and the final accuracy of the estimation. Pitch contours of high-pitched notes were detected with high levels of accuracy, even in audio mixtures with high polyphonies. However, low-pitched notes faced complications that reduced the quality of estimation, in particular for audio mixtures with polyphony 4. Comparable results were observed in harmonically-related notes, in which the type of instrument, the relative volume of each note, and the playing style influenced the final accuracy of the process.

The complete description of the separation strategy for note events is presented and discussed separately in the following chapter, where the proposed multipitch estimator is used as the initial stage of a broader semi-supervised source separation system.

Chapter 5

Semi-supervised Source Separation based on Clustering of Note Events

5.1 Preamble

The work in this thesis concentrates on the decomposition of a single-channel audio mixture into a number of note events, which can be clustered to form individual audio sources. Each note event is characterised by a continuous pitch trajectory, and can be seen as a harmonic sound representing either a single musical note or a group of consecutive notes coming from the same source.

Note events are automatically estimated from the original audio mixture following an iterative approach. In every iteration, the pitch trajectory of the predominant note event is selected from a set of fundamental frequency estimates and used to separate its spectral content from the mixed spectrogram. The extraction of the predominant note event generates a residual, which is used as the input signal in the next iteration. This process repeats until the energy in the residual is below a significance threshold, or until a predefined maximum number of iterations is reached. The detection of pitch trajectories for note events was addressed in the previous chapter, while the current one deals with the estimation and separation of their associated harmonic structures, and with the clustering of note events into separated sources.

Assuming that the pitch trajectory of the predominant note event is known, the separation process consists of detecting a set of relevant spectral peaks in every frame, and partitioning their energy among the predominant note event and the residual. This process is particularly critical for those peaks where the selected note event overlaps with other sources, where additional processing is required to achieve a proper separation.

Section 5.2 describes the peak-picking stage, where the set of spectral peaks associated with harmonic partials of the predominant note event is identified, while Section 5.3 presents the estimation of their parameters. Tracking harmonic partials is discussed in Section 5.4, considering a variety of real musical instruments and playing styles, which provides an insight into the adaptability of the proposed framework. Separation of the harmonic content associated with the predominant note event is addressed in Section 5.5, while two different techniques are considered for its extraction from the current audio mixture, namely, time-frequency masking in Section 5.6.1, and time-domain subtraction in Section 5.6.2. Note event clustering is discussed in Section 5.7 while evaluation of performance is conducted in Section 5.8. Section 5.9 summarises the contributions of this chapter.

5.2 Spectral Peak Picking

An initial task in the separation stage is to detect spectral peaks in each time frame of the mixed spectrogram that are likely to have been produced by harmonic partials of the predominant note event, whilst ideally minimising the number of detected peaks produced by other sources, noise or artifacts of the time-frequency representation.

The pitch trajectory $\mathcal{P}_{\mathcal{T}}(m)$ of the predominant note event is known and it contains the fundamental frequencies of the harmonic structure that has to be separated. In every frame, the peak-picking method computes the ideal centre frequencies of each harmonic partial associated with the corresponding fundamental frequency of the frame, and identifies the most significant spectral peaks closest to those harmonic frequencies. The peak-picking method receives the mixed complex spectrogram \mathbf{X} , the pitch trajectory $\mathcal{P}_{\mathcal{T}}(m)$, the maximum number of harmonic partials to consider H_q , and returns an array \mathbf{C} , with dimensions $H \times M$, where H is the number of harmonic partials detected in every frame, and M is the number of significant frames in the pitch trajectory, containing the index of the central frequency bin of each selected spectral peak. Details of the proposed method are presented in Algorithm 1.

Algorithm 1 Peak-Picking Algorithm

```

1: function PEAK-PICKING
   Input: Mixed Spectrogram  $\mathbf{X}_{(K \times M)}$ , Pitch Trajectory  $\{\mathcal{P}_{\mathcal{T}}\}_{(1 \times M)}$ , Sampling Frequency  $f_s$ ,
   Maximum Number of Harmonics to Extract  $H_q$ .
   Output: Array of Centre Frequency Bins of Selected Spectral Peaks  $\mathbf{C}_{(H \times M)}$ 
2:    $m_a \leftarrow \arg[\mathcal{P}_{\mathcal{T}}(\text{first})]$  ▷ Frame index of the first pitch estimate
3:    $m_b \leftarrow \arg[\mathcal{P}_{\mathcal{T}}(\text{last})]$  ▷ Frame index of the last pitch estimate
4:   for  $m = m_a$  to  $m_b$  do
5:      $f_0 \leftarrow \mathcal{P}_{\mathcal{T}}(m)$  ▷ Current pitch estimate
6:      $h \leftarrow 1$  ▷ Harmonic counter
7:      $f_h \leftarrow f_0$  ▷ Harmonic frequency
8:     while  $(h \leq H_q)$  AND  $(f_h < \frac{f_s}{2})$  do
9:        $k \leftarrow \left\lfloor \frac{2f_h K}{f_s} \right\rfloor$  ▷ Closest frequency bin to  $f_h$ 
10:      if  $k$  is a local minimum then ▷ Local minimum condition
11:         $k \leftarrow$  Adjacent bin closest to  $f_h$ 
12:      end if
13:      for  $j = 1$  to  $3$  do ▷ Centre bin refinement
14:         $k \leftarrow \operatorname{argmax}_{\tau \in [k-1:k+1]} |\mathbf{X}(\tau, m)|$ 
15:      end for
16:       $\mathbf{C}(h, m) \leftarrow k$  ▷ Storing centre bin of  $h$ -th peak in frame  $m$ 
17:       $h \leftarrow h + 1$  ▷ Updating harmonic counter
18:       $f_h \leftarrow h f_0$  ▷ Updating harmonic frequency
19:    end while
20:  end for
21:  return  $\mathbf{C}$  ▷ Centre bins of spectral peaks
22: end function

```

The refinement of the centre bin position of every spectral peak consists of two stages. First, the algorithm checks whether the initial centre bin position, which is obtained from the ideal centre frequency of the corresponding harmonic partial, is a local minimum or not (line 10). If it is a local minimum, the centre bin position is moved to the adjacent frequency bin that is closer to the ideal harmonic frequency, in order to reduce the risk of misleadingly tracking a different partial during the second stage of the refinement. Secondly, a series of three further refinements are conducted on the centre bin position with the aim of finding the local maximum in the vicinity of the initial centre bin position (line 13). Here, each relevant partial is assumed to be the result of convolving a sinusoidal component with a Hanning window. Hence, most of its energy concentrates around five frequency bins, namely, the middle one (closest to the frequency of the sinusoidal component) and two bins on each side of the central one.

This refinement strategy is designed to provide a cautious level of flexibility to the assumption of harmonicity, in order to cope with inharmonic instruments and with artifacts of the STFT. Although the algorithm starts looking at the ideal centre frequencies of the harmonic partials,

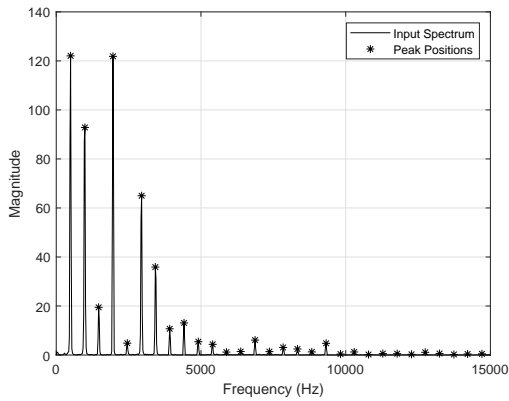
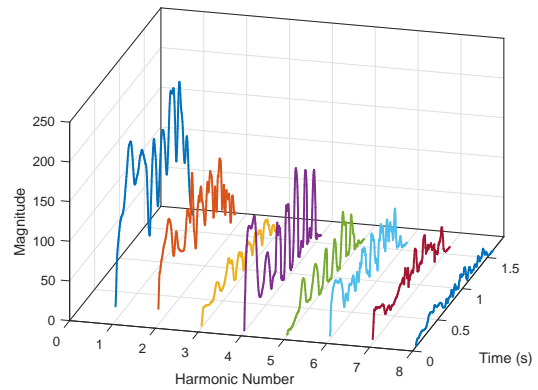
their corresponding spectral peaks are assumed to be centred within a maximum deviation of 3 frequency bins from those ideal positions. Other alternatives such as measuring the degree of inharmonicity of the selected harmonic structure proved unreliable, in particular for low-pitched note events or in those cases where the polyphony of the input mixture is higher than 2.

An additional advantage of the refinement process is related to the proposed separation strategy of overlapping harmonics, where the position of the highest spectral peak in the vicinity of the ideal harmonic frequency is required to estimate the boundaries of the overlapping range and to initialise a dual-peak model for shared partials. Further details of the dual-peak model and other approaches to the separation of overlapping harmonics are provided in Section 5.5.

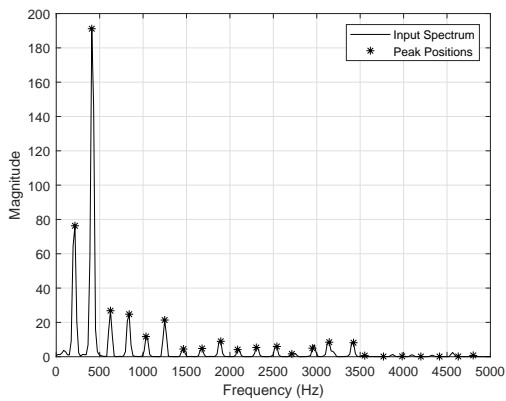
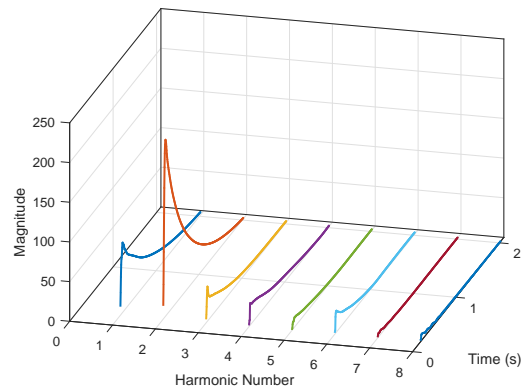
The application of the peak-picking algorithm is illustrated in three different examples, shown in Figure 5.1, considering three musical notes in isolation and with different pitches. The first is a violin note with fundamental frequency $f_0 = 490$ Hz, which benefits the peak-picking strategy by providing sufficient separation in between harmonic partials. Hence, detecting spectral peaks is possible even up to the 30-th harmonic partial, as presented in Figure 5.1(a). Figure 5.1(b) also shows how the playing style is captured by the harmonic magnitude contours.

The second example is a piano note with $f_0 = 210$ Hz, in which the detection of spectral peaks is more difficult because of the inharmonicity of its partials. However, Figure 5.1(c-d) show a successful detection of at least the first 15 harmonic partials, comprising most of the note energy, while the missed harmonics exhibit very low absolute magnitudes at relatively high frequencies.

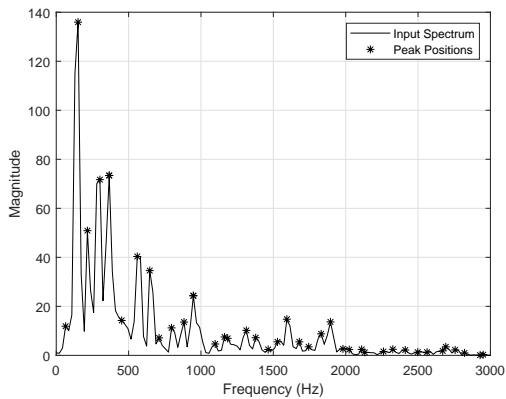
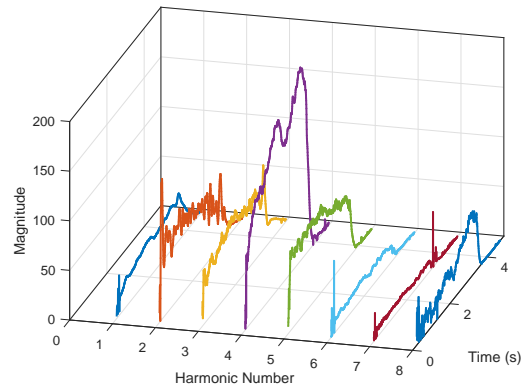
Figure 5.1(e-f) show the selected spectral peaks for a cello note with $f_0 = 73$ Hz. In this case, the low fundamental frequency reduces the separation between harmonic partials, leading to more overlapping partials and more phase interactions associated with these overlapping partials, potentially generating higher distortion in a short-framed magnitude spectrum. An example is shown in Figure 5.1(e) where the sixth harmonic partial ($f_6 = 452$ Hz) does not seem to have a clear spectral peak associated with it, and may not be detected by other approaches based on frequency dependent thresholds. The proposed refinement process, on the other hand, has been designed to cope with some of these complexities and ensures that at least one spectral peak is selected in the vicinity of each harmonic frequency, even if it is very weak.


 (a) Spectrum at $t = 0.18$ s


(b) Harmonic Magnitude Contours


 (c) Spectrum at $t = 0.28$ s


(d) Harmonic Magnitude Contours


 (e) Spectrum at $t = 0.33$ s


(f) Harmonic Magnitude Contours

Figure 5.1: Peak-picking method applied to three different notes in isolation. (a-b) Violin B4 ($f_0 = 490$ Hz), (c-d) Piano A3 ($f_0 = 210$ Hz), and (e-f) Cello D2 ($f_0 = 73$ Hz).

5.3 Spectral Peak Parameters

In the previous section, an array \mathbf{C} was obtained following the proposed peak-picking method and it contains the centre frequency bins of the spectral peaks that best associate with the desired

harmonic partials in every frame. This information is used here to estimate a set of parameters (centre frequency, amplitude and phase angle) for each detected peak.

Considering the m -th frame of the input spectrogram, let $k_v = \mathbf{C}(v, m)$ denote the frequency bin in which the v -th spectral peak occurs. The rough peak magnitude and phase angle are obtained by sampling the complex spectrum at this frequency bin, giving $r(k_v) = |\mathbf{X}(k_v, m)|$ and $\varphi(k_v) = \angle \mathbf{X}(k_v, m)$, respectively. Since the Fourier transform of the input signal is convolved by that of the window function in every time frame, parameters $r(k_v)$ and $\varphi(k_v)$ are partly determined by the shape of the Fourier transform of the window function centred at the true partial frequency, which in general will not be situated exactly at the frequency bin k_v . For this reason, the initial values of $r(k_v)$ and $\varphi(k_v)$ have to be refined in order to obtain a better estimate of the peak parameters.

The refined centre frequency of the underlying partial is obtained by means of quadratic interpolation, where the three ordered pairs $[k_{v-1}, r(k_{v-1})]$, $[k_v, r(k_v)]$ and $[k_{v+1}, r(k_{v+1})]$ are used as the reference points to fit a second order polynomial of the form $r = ak^2 + bk + c$, whose constants are computed as follows.

$$a = \frac{[r(k_{v+1}) - r(k_v)](k_v - k_{v-1}) - (k_{v+1} - k_v)[r(k_v) - r(k_{v-1})]}{(k_{v+1}^2 - k_v^2)(k_v - k_{v-1}) - (k_{v+1} - k_v)(k_v^2 - k_{v-1}^2)} \quad (5.1)$$

$$b = \frac{[r(k_v) - r(k_{v-1})] - a(k_v^2 - k_{v-1}^2)}{(k_v - k_{v-1})} \quad (5.2)$$

$$c = r(k_v) - ak_v^2 - bk_v \quad (5.3)$$

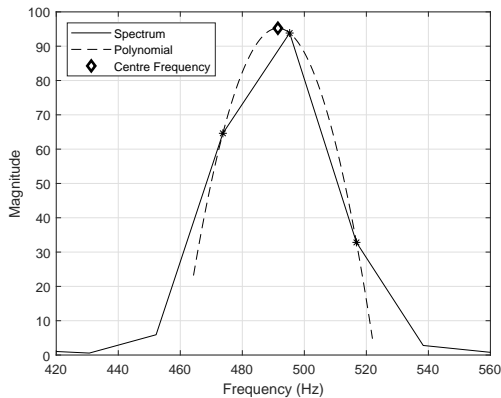
Then the refined centre frequency of the v -th harmonic partial is obtained as follows.

$$f_v = \frac{f_s}{F_{\text{SIZE}}} \left(\frac{-b}{2a} \right) \quad (5.4)$$

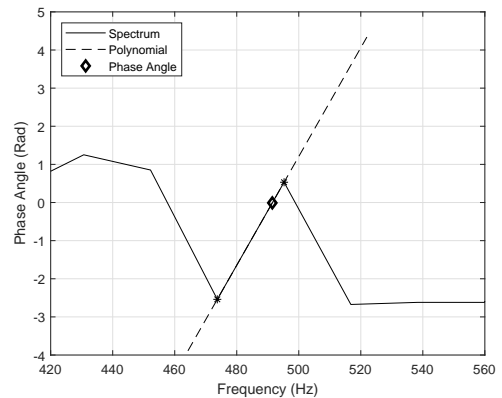
where F_{SIZE} is the frame size. The centre frequency f_v lies somewhere in between two adjacent frequency bins of the input spectrum, denoted by k' and k'' , where $k' < -b/2a < k''$. These two frequency bins also define a segment of the phase spectrum on which the phase angle of the v -th partial is located, which can be estimated by linear interpolation as follows.

$$\phi_v = \frac{2\pi}{F_{\text{SIZE}}} \left[\left(\frac{\varphi(k'') - \varphi(k')}{k'' - k'} \right) \left(\frac{-b}{2a} - k' \right) + \varphi(k') \right] \quad (5.5)$$

Figure 5.2 shows an example of the quadratic interpolation of the magnitude and the linear interpolation of the phase angle, for a harmonic partial associated with a real violin note.

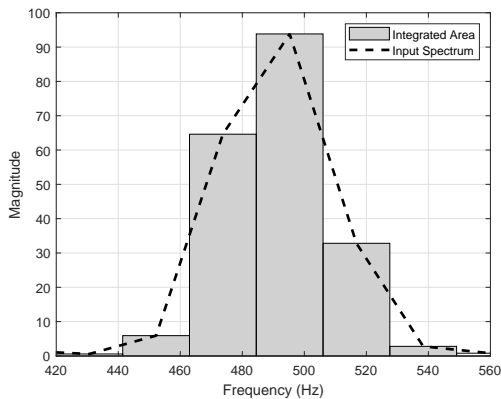


(a) Centre Frequency Interpolation

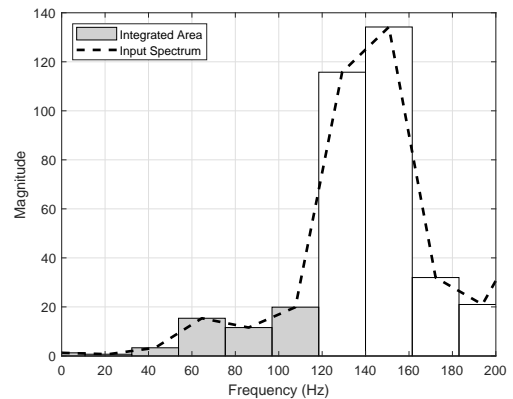


(b) Phase Angle Interpolation

Figure 5.2: Parameter estimation for a harmonic partial associated with a real violin note. Reference points are marked with asterisks while estimated parameters are marked with diamonds.



(a) Violin Note B4



(b) Cello Note D2

Figure 5.3: Estimation of the integrated partial amplitude for two different notes, considering five frequency bins to define the significance area of each partial.

The third parameter to compute is the partial amplitude, which is obtained by integrating the area associated with the selected peak. Considering five frequency bins as the integration range (two bins on either side of the centre one), the estimated amplitude of the v -th partial is defined as follows.

$$A_v = \frac{2}{F_{\text{SIZE}}} \sum_{\tau=k_v-2}^{k_v+2} |\mathbf{X}(\tau, m)| \quad (5.6)$$

This strategy is appropriate for harmonic partials with no other frequency component nearby. However, the quality of the estimated amplitude degrades as the selected partial gets closer to other frequency components. An example is presented in Figure 5.3 where two different peaks are presented. The fundamental partial in Figure 5.3(a) corresponds to a violin note in isolation

($f_0 = 490$ Hz), in which the distance between harmonics allows an accurate estimation of the integrated amplitude. Figure 5.3(b), on the other hand, shows the fundamental partial of a cello note ($f_0 = 73$ Hz), where the distance between harmonic partials is less than two frequency bins. In this case, equation (5.6) leads to an estimated amplitude which is excessively high due to the overlap with the second harmonic. To cope with this limitation, a variation of the aforementioned strategy is presented in Section 5.5.

5.4 Tracking Harmonics

Finding the correct set of spectral peaks and their parameters is crucial to achieve an effective separation of the spectral content associated with the predominant note event. However, the accurate tracking of harmonic partials depends on the characteristics of each note event, in particular, the type of instrument involved, its fundamental frequency, the degree of inharmonicity and the playing style. Among these potential sources of error, the level of inharmonicity and playing style are the most significant ones since they modify the centre frequencies of partials and introduce deviations from the assumption of harmonicity. In this section, the effects of inharmonicity and the playing style on the accuracy of the peak-picking and harmonic tracking algorithm are explored using a set of synthetic harmonic signals and real musical notes in isolation.

5.4.1 Effects of Inharmonicity

To evaluate the accuracy of the system under different levels of inharmonicity, an array of four fundamental frequencies is selected as $p = \{110 \text{ Hz}, 220 \text{ Hz}, 440 \text{ Hz}, 880 \text{ Hz}\}$, and for each of them, five artificial notes are synthesised considering five different inharmonicity coefficients ranging from zero to a pitch-dependent maximum. Each synthetic signal, with a duration of 1.5 s, consists of 30 partials with exponentially decaying amplitudes. The maximum inharmonicity coefficients are chosen as $B_{max} = \{0.00018, 0.00041, 0.00124, 0.0047\}$, based on experimental measurements conducted by Hendry [25] on six strings of a grand piano (Steinway Concert Grand Model D), which included the four notes in p . The proposed harmonic tracking framework was applied to the synthetic signals using ground-truth and automatically estimated pitch trajectories. Results in Figure 5.4 show the normalised Mean-Squared Error (MSE) between the real and estimated centre frequencies for the 30 partials included in each of the input signals.

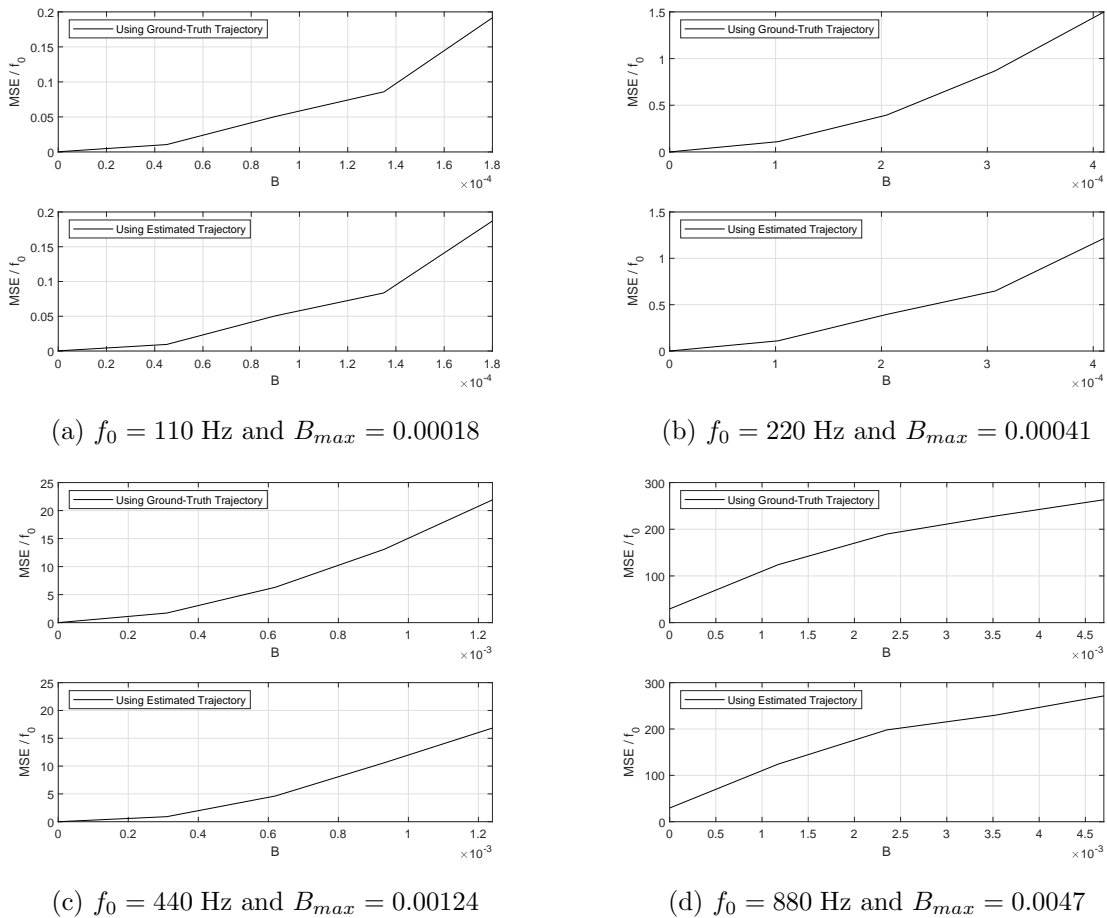


Figure 5.4: Normalised mean-squared error between the real and estimated harmonic frequencies for a set of synthetic notes with different pitches and levels of inharmonicity. (Top) Using a ground-truth pitch trajectory. (Bottom) Using an automatically estimated pitch trajectory.

As the inharmonicity coefficient increases, the centre frequency of the n -th partial deviates more and more from its ideal harmonic frequency, as modelled in a piano string by the relationship $f_n = nf_0\sqrt{1+Bn^2}$. If this deviation is higher than three frequency bins, the algorithm will not be able to identify the correct spectral peak, which explains the increment in error as the pitch of the note increases. Providing more flexibility to the system might reduce the error in this particular test, but it might also increase the error in other polyphonic cases, in which the additional flexibility would incorrectly guide the algorithm towards another partial from a nearby source. If the deviation is small and local, affecting a small number of partials, the algorithm is able to find the right set of spectral peaks, but if it is a progressive deviation (as in piano notes) the risk of missing a high number of partials significantly increases, in particular for high-pitched notes.

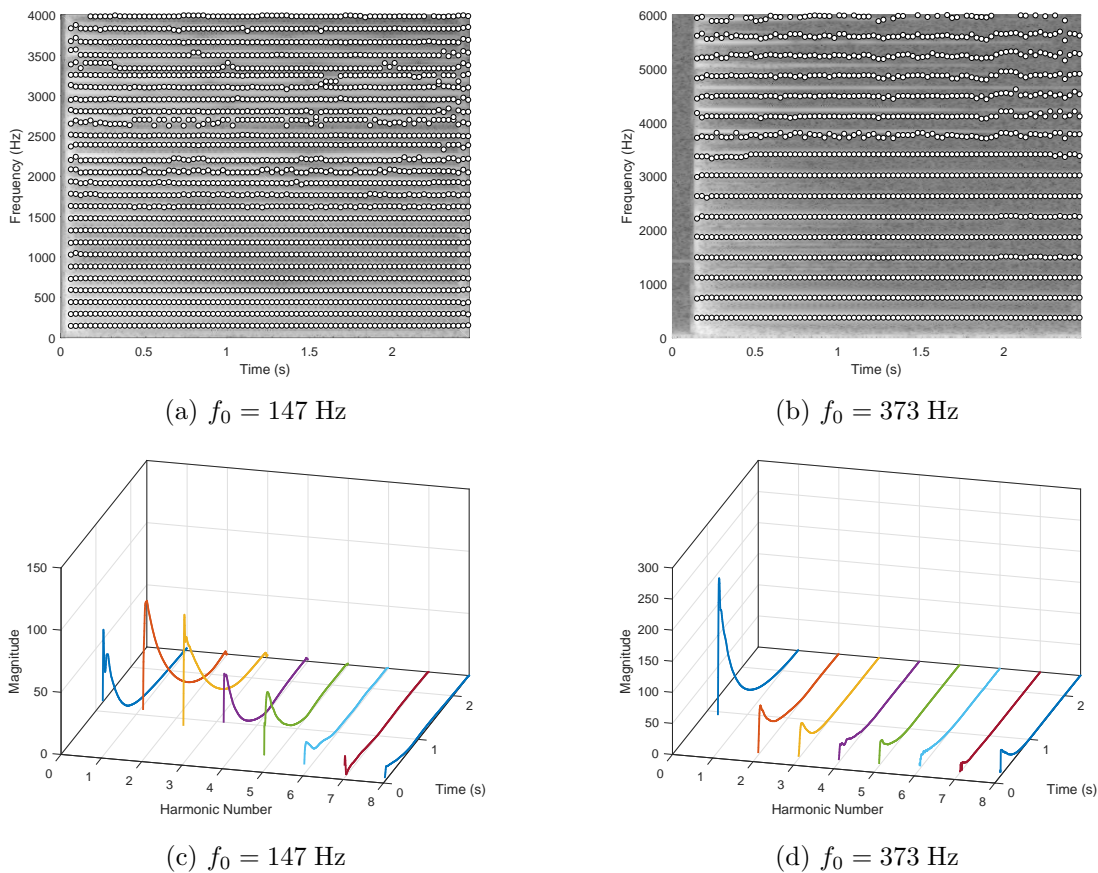


Figure 5.5: Spectrograms and harmonic magnitude contours of two real piano notes in isolation. Estimated harmonic trajectories in each frame are shown with circles in (a) and (b). These results were obtained using automatically estimated pitch trajectories.

Figure 5.4 also shows very little difference between the error rates obtained by using the ground-truth pitch trajectories and the automatically estimated ones. In this case, the algorithm is able to cope more effectively with deviations in fundamental frequency, given that the amount of deviation does not increase with the harmonic number. As deviations in fundamental frequency estimates are more common than the occurrence of highly inharmonic sources, the two-bins flexibility criterion is going to be considered within the peak-picking framework in all further tests. Two additional examples are presented in Figure 5.5 for two real piano notes from the RWC database [129], where peak-detection errors are more evident for the note with the higher pitch.

5.4.2 Effects of Vibrato

It is also worth investigating the behaviour of the system under highly dynamic spectral conditions, which are usually associated with some physical properties of instruments and how

musicians perform with them. Good examples of these dynamic conditions are the ones associated with audio effects, such as vibrato and tremolo, in which the performer introduces time-varying changes in the frequencies and amplitudes of the harmonic partials forming the notes. Under these circumstances, each frame of the time-frequency representation becomes more non-stationary.

As the frequency and amplitude of each harmonic partial are not constant within a single frame, the peak shape in the magnitude spectrum also changes. The usual consequence of this phenomenon is the broadening of spectral peaks, but if the rate of change in frequency is significantly fast, a single frequency component can eventually look like two spectral peaks very close to each other. The harmonic tracking algorithm should be able to follow these spectral dynamics in order to extract most of the energy associated with the note event, without losing the character and intentions of music performers.

To illustrate the extent to which the harmonic tracking strategy is able to follow some of these dynamic features of music, two examples are shown in Figure 5.6. The two violin notes presented show some degree of vibrato, evidenced by the oscillatory nature of their harmonic partials. Unlike the case of inharmonic notes, audio effects such as vibrato have the advantage of introducing regular deviations in the centre frequency of harmonic partials, which are in many cases captured by the estimated pitch trajectory. Hence, the corresponding spectral peaks can be accurately located in every frame even for high order partials, as shown in Figure 5.6(a-b). The harmonic magnitude contours associated with the first eight partials are presented in Figure 5.6(c-d) for both musical notes, respectively.

5.5 Separation of Spectral Content

At this point, the set of relevant spectral peaks associated with the predominant note event have already been identified in every frame, and rough estimates of their parameters have also been computed. Some of these peaks might correspond to isolated harmonic partials of the predominant note event, with very little or no interaction with other sources. However, other peaks might be associated with spectral regions where partials of the predominant note event overlap with components of other concurrent sources. Hence, spectral content separation is defined as the task of identifying the correct amount of energy within every spectral peak that corresponds to the exact contribution of the predominant note event to the input mixture.

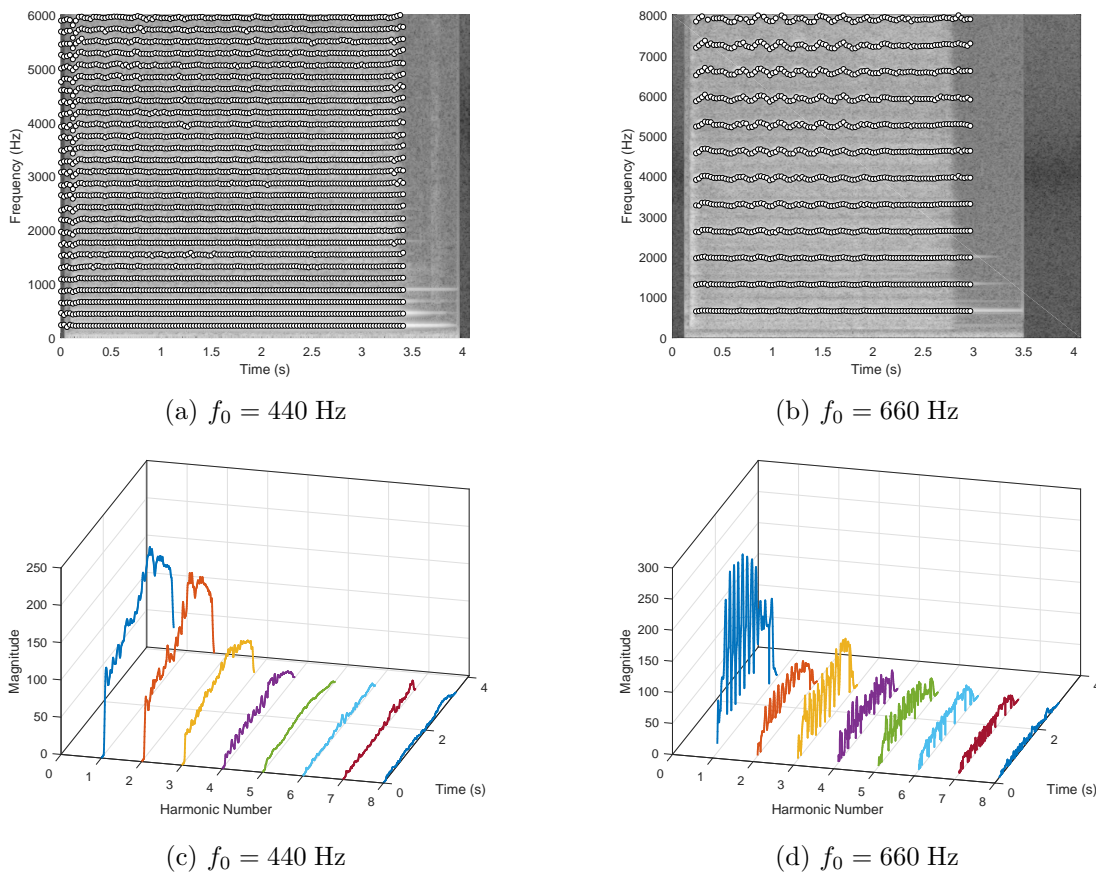


Figure 5.6: Spectrograms and harmonic magnitude contours of two real violin notes in isolation. Estimated harmonic trajectories in each frame are shown with circles in (a) and (b). These results were obtained using automatically estimated pitch trajectories.

The amount of energy extracted from within the spectral peak has to be decided depending on the amount of overlap that it exhibits. In some of the cases, when harmonic partials overlap, the shape of the shared peak allows an approximate estimation of the underlying partials, usually when their centre frequencies are not very close. However, when the involved components are too close, the shared peak adopts the same shape as a single-component partial, revealing no information about the constituent components. For these reasons, two spectral separation strategies are presented in the following sections to account for semi-overlapping and fully-overlapping partials.

5.5.1 Detection and Classification of Overlapping Partial

Several strategies have been presented to detect overlapping harmonics in single-channel audio mixtures. Most of them require the knowledge of the fundamental frequencies associated with the concurrent sources. Parsons proposes the use of spectral peak symmetry, their phase

behaviour, and the distance from adjacent peaks, as criteria for detecting overlapping partials [81]. Systems in [23, 26, 65, 70, 82] use built-in multipitch detectors to obtain estimates of the harmonic frequencies, while different proximity criteria and harmonic masks were proposed to detect overlapping regions. Every's method [69] was also based on proximity measures, but pitch information was obtained from a time-aligned MIDI score. Carabias-Orti et al. combined matching pursuit and harmonic dictionaries as a way to handle overlapping harmonics, while the use of convolutional recurrent neural networks was explored by Adavanne et al. in [130].

In this work, the detection of overlapping partials takes place during the separation of the spectral peaks associated with the predominant note event, using a combination of peak shape-based and proximity-based criteria. Selected spectral peaks are classified as fully-overlapping or semi-overlapping partials, where peaks in the first category are identified first by computing their proximity to the closest interfering component, which might indicate the presence of another spectral component associated with a note event in harmonic relationship. Potential interfering events are detected along with the predominant pitch trajectory, as presented in the previous chapter (Section 4.3.6), while preservation rates are assigned to them depending on whether these interferers are likely to be real notes in harmonic relationship. However, if the frequency distance between the estimated underlying components is greater than a predefined minimum, the corresponding spectral peak is labelled as a combination of semi-overlapping partials. Different separation methods are proposed to handle overlapping partials in each category, which are discussed further later in this section.

The minimum frequency distance used here to distinguish between fully-overlapping and semi-overlapping partials has been empirically selected based on experiments conducted on pure sinusoidal components. In each experiment, a number of single-frame synthetic signals were constructed by combining two sinusoidal components with different parameters, as presented in equation (5.7).

$$y(t) = A_1 \cos(2\pi f_1 t + \phi_1) + A_2 \cos(2\pi f_2 t + \phi_2) \quad (5.7)$$

While the amplitudes A_1 and A_2 , and phase angles ϕ_1 and ϕ_2 of the components remained fixed during each experiment, the distance between their centre frequencies f_1 and f_2 was gradually reduced from 60 Hz to 20 Hz, in steps of 5 Hz. Then, parameters of two potentially overlapping partials, termed the dominant and secondary components (defined in Section 5.5.2), were estimated directly from the shared peak using the algorithm in [81], and their average deviations from the real parameters were computed in each case. Results obtained from these experiments

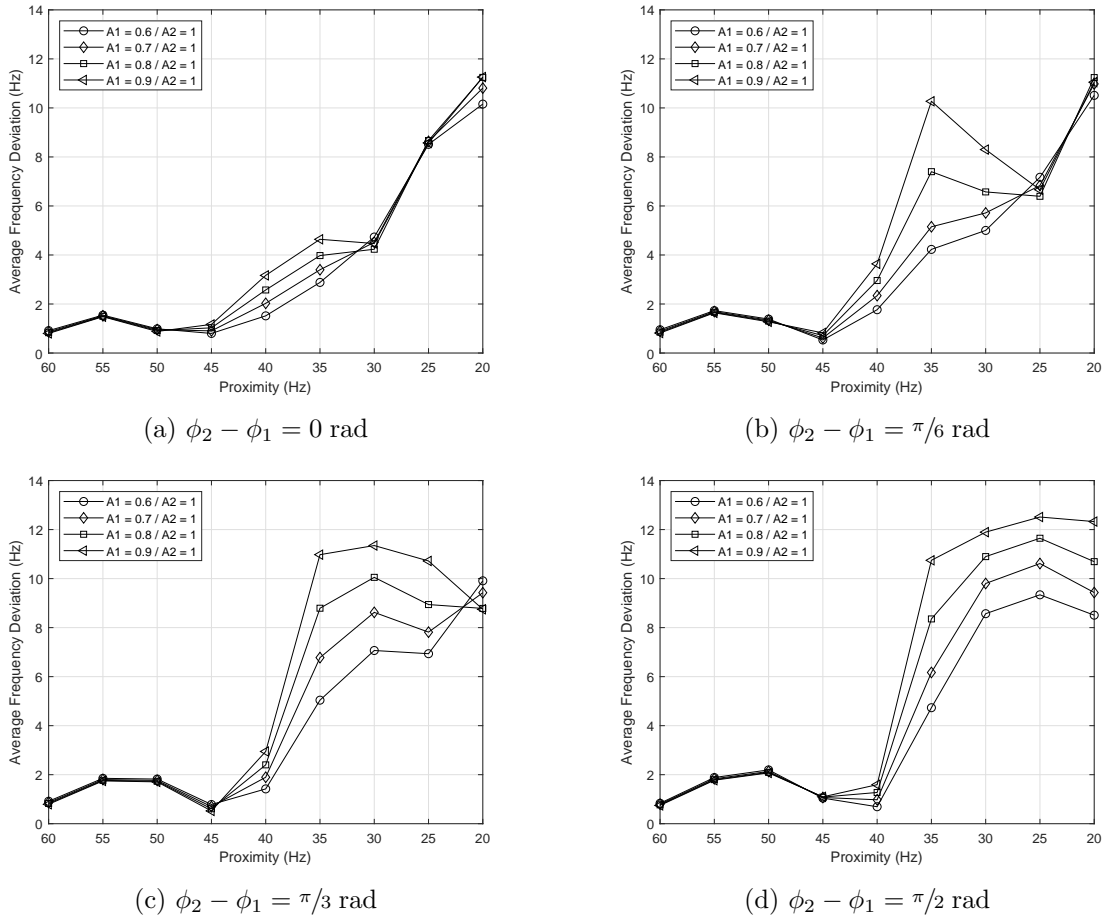
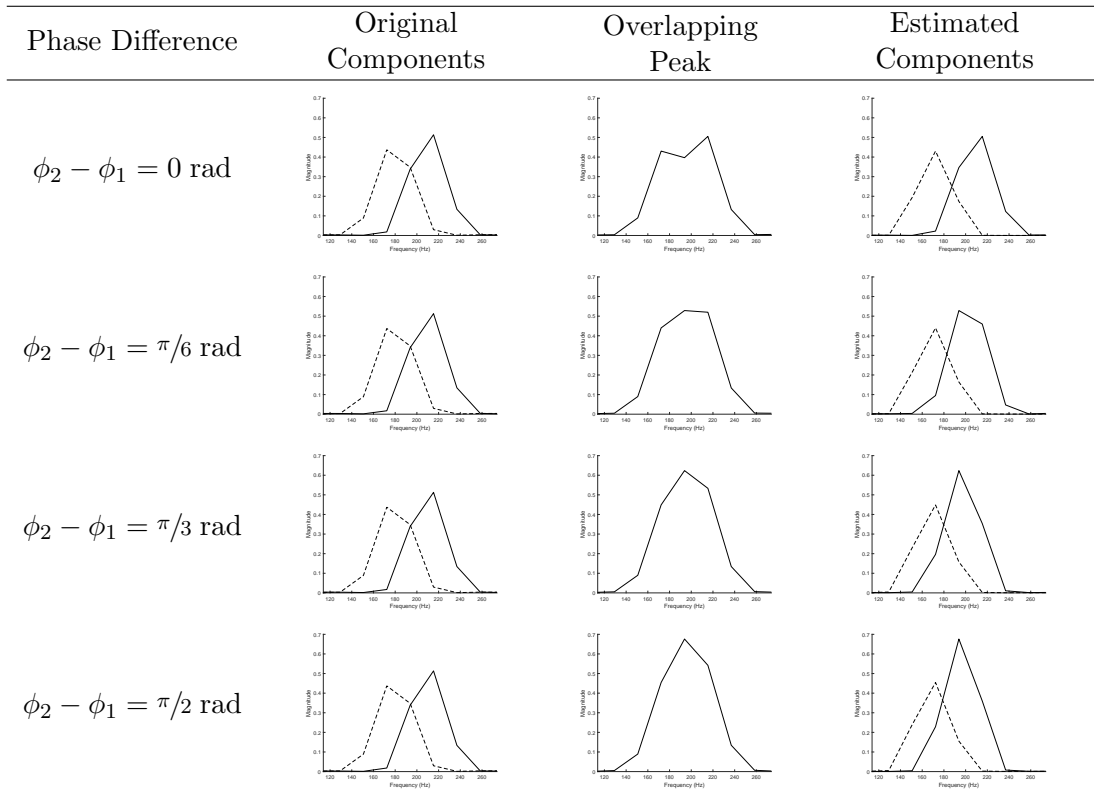


Figure 5.7: Average frequency deviation between estimated components during the estimation of their centre frequencies in a number of synthetic mixtures consisting of two sinusoidal components with different phase differences.

are presented in Figure 5.7, considering $0.6 \leq A_1 \leq 1$, $A_2 = 1$, $150 \text{ Hz} \leq f_1, f_2 \leq 210 \text{ Hz}$ and four phase differences, where frequency deviations below 2 Hz were obtained for overlapping partials separated by more than 45 Hz.

As the underlying components get closer, the quality of the estimated parameters degrades, particularly the amplitudes and phase angles. The deviation of the estimated centre frequencies from their original values increases at different rates depending on the phase difference, reaching an average deviation of 10 Hz when the distance between the centre frequencies of the underlying components is 20 Hz. Though this increment is not linear, as Figure 5.7 shows; it seems to have a turning point somewhere in between 40 Hz and 30 Hz. Hence, the shape of the resulting estimated partials, which are synthesised from these potentially deviated parameters, may significantly differ from that of the original overlapping partials, reducing the chances of

Table 5.1: Original and estimated components from a set of four overlapping peaks consisting of two sinusoidal tones with frequencies $f_1 = 180$ Hz, $f_2 = 210$ Hz, and different phase angles.

obtaining an effective estimation and extraction of the spectral content of the predominant note event.

Additional tests have also shown that a decomposition of the shared peak into a dominant and a secondary peak is still feasible even when the overlapping partials are just 30 Hz apart, and despite the deviation in their estimated parameters, they normally exhibit a spectral shape that is not completely different from that of the real partials. An example is presented in Table 5.1, where four overlapping partials are created from two sinusoidal tones with frequencies $f_1 = 180$ Hz and $f_2 = 210$ Hz, amplitudes $A_1 = 0.6$ and $A_2 = 1$, and phase differences from 0 to $\pi/2$ rad. When both components are in phase, the overlapping peak clearly shows the presence of both components and therefore the estimated parameters suffer little deviation from the real values, and the shape of the estimated components is still close to the original partials. In the rest of the cases, the phase difference between the underlying partials produces a broadened overlapping peak where the actual position of its components is less clear. Nevertheless, the overall shape of the estimated partials is still close to the original ones, and the separation of these shared peaks should still be possible by means of the algorithm in [81], hereafter referred to as Parsons' method.

However, when the distance between components is reduced even more, the detection of a secondary peak with a significant magnitude cannot be ensured, since the resulting shared peak would start looking as a single-component partial. For this reason, the frequency distance of 30 Hz has been found to be the minimum safe threshold that ensures the detection of an appropriate secondary peak that allows the shared peak to be treated as a set of semi-overlapping partials. Below that point, shared peaks are handled differently as a set of fully-overlapping partials.

5.5.2 Semi-Overlapping Partial

When the distance between centre frequencies in a group of overlapping partials is higher than 30 Hz, the shape of the resulting shared peak differs from the typical peak shape produced by the convolution of a pure sinusoidal component and the window function. Components of such a group are considered to be semi-overlapping partials, in which one of the components is assumed to be a harmonic partial associated with the predominant note event.

Spectral peaks associated with semi-overlapping partials are decomposed into a number of frequency components, using Parsons' method, which exploits the additive nature of overlapping harmonics. In Parsons' algorithm, a pure sinusoidal component is first generated using the estimated parameters of the shared peak, and then convolved with the window function to generate an equivalent single-component spectral peak, referred to here as the dominant component. Then, the shared peak is isolated from the original spectrum and the dominant component is subtracted from it in order to reveal potential overlapping components. If a significant peak appears in the resulting difference vector, it is treated as a secondary component associated with the potential contribution of another source, and its parameters are also estimated. Separation of the shared peak is then achieved by taking the component closest to the ideal centre frequency of the desired harmonic partial. Any uncovered peaks with magnitude more than 20 dB below the principal component are rejected as either spurious or inaudible.

Parsons' method was originally conceived in the context of separation of vocalic speech of two competing talkers, and it was not applied to polyphonic music, where the number of concurrent sources is typically higher. But its simplicity makes it suitable for the separation of semi-overlapping partials within the proposed framework, given that the ideal centre frequencies of the desired harmonic partials can be easily computed from the pitch trajectory, and the iterative nature of the system in which only one note event is extracted at a time. However, Parsons' method is used here under the assumption that the shared peak is the result of two underlying components, one of which corresponds to the target partial. But, instead of this

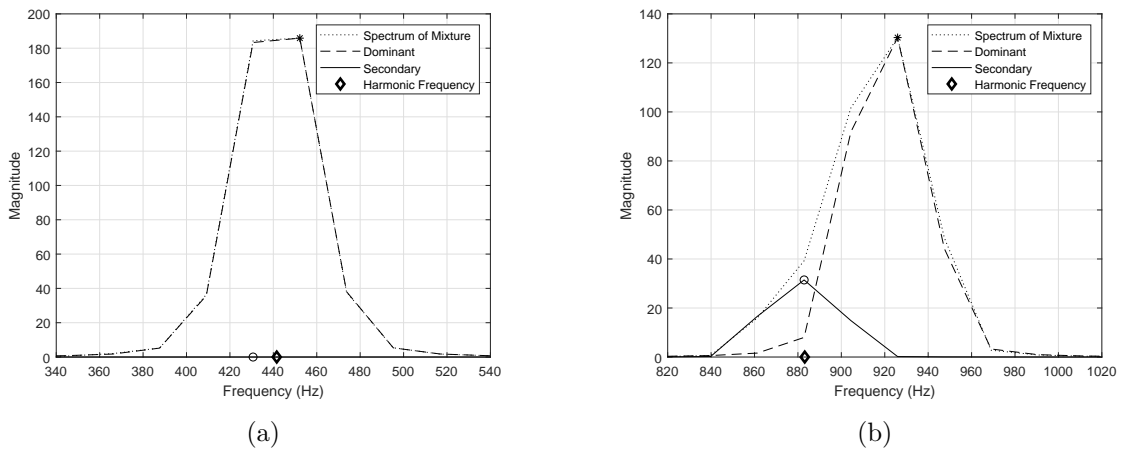


Figure 5.8: Decomposition of two different spectral peaks using Parsons' method. (a) Spectral peak associated with a single harmonic of a real viola note. (b) Shared peak associated with two semi-overlapping harmonics in a mixture of viola and clarinet. In both cases, an asterisk (*) and a circle (o) are used to mark the peak positions of the dominant and the secondary components, respectively. Notice the magnitude of the secondary component in (a) is very low.

being a limitation, Parsons' method is considered as a way to separate the desired partial from a more complex spectral structure.

A decomposition based on Parsons' method is demonstrated for two different circumstances in Figure 5.8. First, a spectral peak corresponding to the fundamental harmonic of a viola note is analysed in Figure 5.8(a), in which the estimated dominant and secondary components are presented. Since the observed spectral peak does not correspond to an overlapping partial, the shape of the dominant component is very close to that of the observed peak, while the secondary component shows a very low magnitude and therefore, it is labelled as a spurious component.

Figure 5.8(b), on the other hand, shows a spectral peak associated with two overlapping partials in a mixture of viola and clarinet. The fundamental harmonic of the clarinet (centred at 923 Hz) overlaps the second harmonic of the viola note (centred at 883 Hz), which is the one associated with the note event being extracted in this example. Since the distance between their centre frequencies is approximately 40 Hz, the decomposition of the resulting shared partial clearly shows the presence of two overlapping components with significant magnitudes, where the secondary component is the one closest to the ideal frequency of the viola's second harmonic. A comparison between the estimated components and the original harmonic partials is presented in Figure 5.9 for the same semi-overlapping peak consisting of viola and clarinet. It is important to mention that the reconstructed partials were adjusted to match the magnitude of the mixed spectrum at their corresponding centre frequency bins, in order to cope with the already

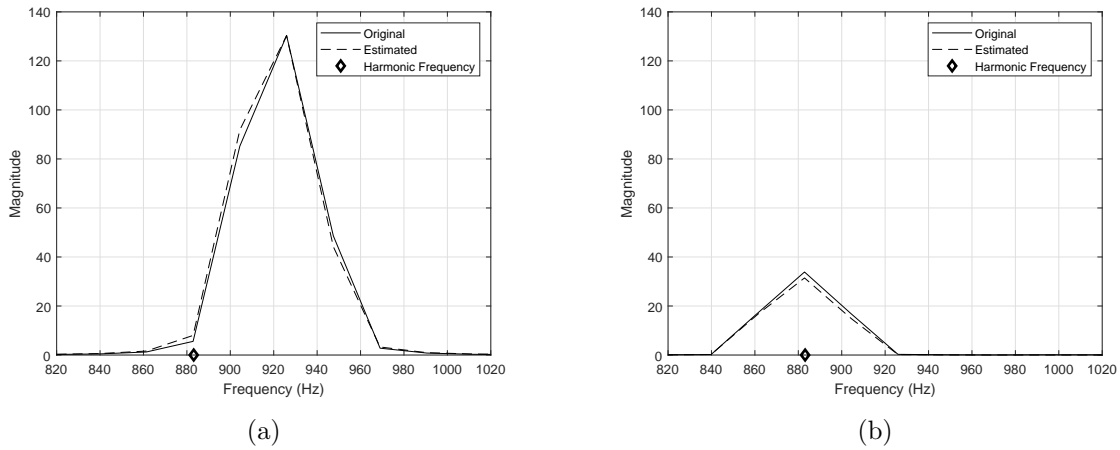


Figure 5.9: Original and estimated components associated with the semi-overlapping peak shown in Figure 5.8(b), taken from a mixture of viola and clarinet. (a) Fundamental partial of the clarinet note $A\sharp_5$. (b) Second harmonic of the viola note A_4 .

discussed deviations in the estimated amplitudes and reduce the risk of artificially increasing the energy of the separated note event.

The separation of the semi-overlapping peak is achieved by taking the estimated component closest to the ideal centre frequency of the target partial. If the secondary component is chosen, the new centre bin position is updated in array \mathbf{C} . The selected component is the one that will be considered during the extraction of the predominant note event, following one of the two alternatives presented in this work, which are addressed in Section 5.6.

5.5.3 Fully-Overlapping Partial

These are overlapping partials in which the underlying components are very close. The distance between their centre frequencies is usually less than a frequency bin, and full separation of the original components cannot be obtained. The proposed strategy does not attempt an exact separation, but focuses on extracting some percentage of the total energy in the shared partial, so that the remainder can be detected as part of a different note event in later iterations.

In this case, only the dominant component can be estimated. Then, an incomplete separation of the overlapping partial is achieved by extracting an attenuated version of its dominant component, in which the absolute magnitude is reduced according to the preservation rates assigned to its interfering events. If the n -th harmonic partial of the predominant note event collides with frequency components from K interfering events, with preservation rates $\alpha_1, \alpha_2, \dots, \alpha_K$, then the dominant component is attenuated by multiplying its absolute magnitude by the following gain.

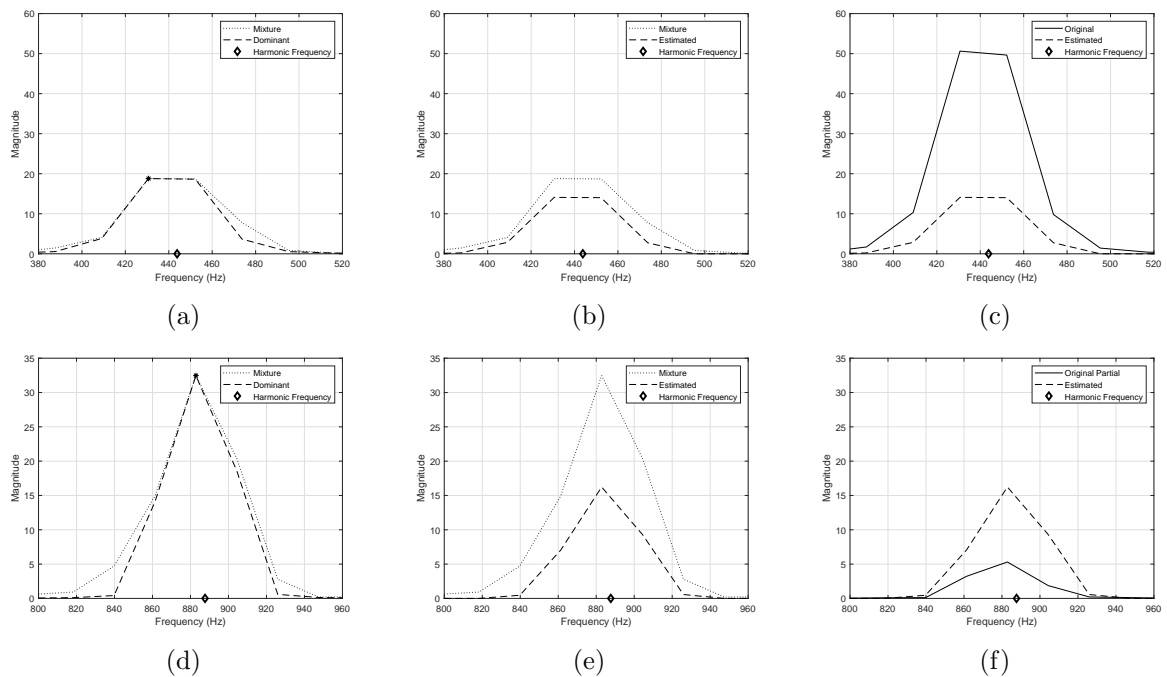


Figure 5.10: Incomplete separation of two fully-overlapping partials in a mixture involving two violins playing the notes A3 ($f_0 = 220$ Hz) and A4, where note A3 is the predominant one and two interferers are centred at 440 Hz and 880 Hz. (Top) Separation of the second harmonic of note A3, centred at 440 Hz, which collides with the first harmonic of the interferer at 440 Hz, thus the gain $G_2 = 0.75$ is applied. (Bottom) Separation of the fourth harmonic of note A3, centred at 880 Hz, which simultaneously collides with the second harmonic of the interferer at 440 Hz and with the first harmonic of the interferer at 880 Hz, hence the gain $G_4 = 0.5$ is applied. (a,d) Observed overlapping partial, (b,e) Attenuated dominant component, (c,f) Comparison between the original harmonic and the attenuated dominant component.

$$G_n = 1 - \sum_{k=1}^K \alpha_k \quad (5.8)$$

The second term in Equation 5.8 cannot be greater than $2A_{max}$ to avoid leaving excessive energy in the residual. This attenuated component is the one taken as the target partial for separation. Figure 5.10 shows two examples of fully-overlapping partials taken from a mixture of two violin notes in an octave relationship (A3 and A4). In this examples, the violin note A3 ($f_0 = 220$ Hz) has been chosen as the predominant note event, while two potential interferers have also been found at 440 Hz and 880 Hz, which are allocated preservation rates of $\alpha_1 = 0.25$ and $\alpha_2 = 0.25$, respectively, despite the fact that the 880 Hz interferer does not actually exist. Figure 5.10(a-c) shows the separation of the second harmonic of the predominant note event. In this case, the underlying components are just 3.62Hz apart, so the apparent single peak at 440 Hz is due to strongly overlapping harmonics and appears to be a single component partial.

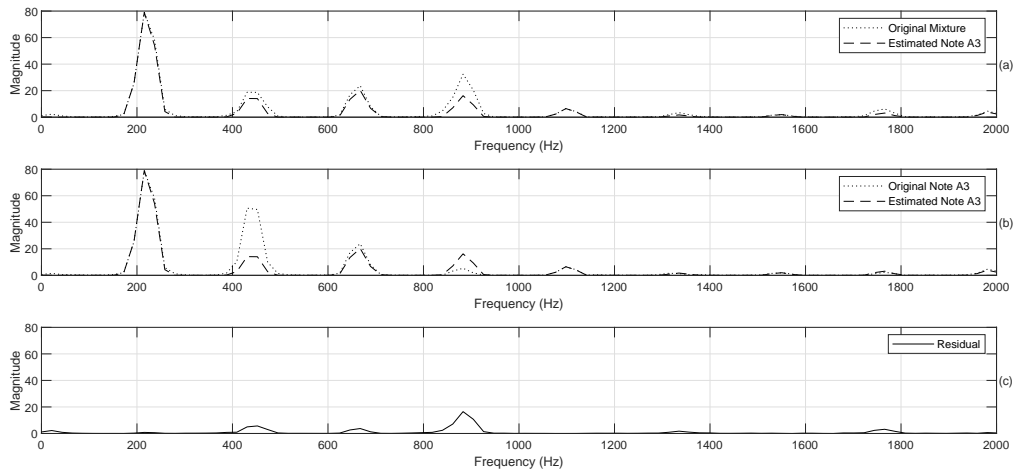


Figure 5.11: Estimated spectral content of the violin note A3 in a mixture containing the violin notes A3 and A4. (a) Comparison between the estimated spectral content and the original mixture. (b) Comparison between the estimated spectral content and the original note A3. (c) Residual.

When the dominant component is generated (a), its resulting shape ends up being very similar to that of the shared peak. Consequently, in order to extract an appropriate proportion of the peak's energy (b), the dominant component is taken and attenuated by the gain $G_2 = 0.25$. A comparison between the selected partial and the original second harmonic of the predominant event clearly shows the magnitude mismatch (c), which cannot be avoided for fully-overlapping partials, but it also shows the effects of phase interactions since the original magnitude of the second harmonic is actually much higher than that of the shared partial. A similar case is presented in Figure 5.10(d-f), where the separation of the fourth harmonic of the predominant note event is presented. In this second example, the underlying components are 5.64 Hz apart and the dominant component is attenuated by the gain $G_4 = 0.5$, due to the presence of two potential interfering frequencies at 880 Hz.

The estimated spectral content associated with the violin note A3 is presented in Figure 5.11 for the complete frame, following the previous example, and it is compared against the spectrum of the input mixture and with the spectrum of the original violin note A3. The remaining energy in the residual, which in this iteration is presented in Figure 5.11(c), should allow the detection of the second violin note (A4) in later iterations. However, in those cases where the real note in harmonic relation has not been properly detected as an interfering event, or if its preservation rate has been set too low, its detection during a later iteration might not be possible.

Separated events that originate from note events in a harmonic relationship usually exhibit high levels of interference, since the partitioning of their spectral energy was conducted in an arbitrary way that only benefits the detection of their pitch trajectories. In order to achieve a proper separation of these events an additional stage has to be implemented for the estimation of the true parameters associated with their harmonic partials, which should allow an optimal separation of their spectral content. An approach based on optimisation is an attractive option that could deliver high quality parameters for fully-overlapping harmonics, in which phase information can also be incorporated to improve the quality of the estimation. However, the success of this technique would rely on finding a more effective way of initialising the optimisation algorithm within a small region where the true parameters might be located. Otherwise the algorithm could be driven towards another incorrect local minimum, or would take a great deal of time to converge. Additional thoughts on this matter can be found in Chapter 7.

5.6 Note Event Extraction

In the previous section, a set of spectral peaks in every frame was carefully analysed, classified and separated according to their shapes and proximity to potentially interfering components. The result is the estimated magnitude spectrogram of the separated predominant note event, with centre frequency bins for all selected harmonic partials in every frame indicated in array **C**. The next stage consists of extracting the spectral energy associated with the predominant note event from the input mixture. For this purpose, two different methods are presented and discussed in the upcoming sections.

5.6.1 Time-Frequency Masking

This method consists of fitting a non-binary time-frequency mask to extract the energy of all selected harmonic partials associated with the predominant note event in every frame. This mask is constructed for every harmonic partial, using a dual-peak model as a reference. Considering the h -th harmonic partial in the m -th frame, centred at frequency bin k^h , the boundaries of the spectral peak have to be determined based on its shape, and on the assumption that the energy within a stationary sinusoidal partial spans over a maximum of five frequency bins with centre at bin k^h (convolution of the original component with a Hanning window). Hence, the boundaries of the h -th harmonic partial, as illustrated in Figure 5.3, are defined as follows.

$$k_{min}^h = k^h - 2 \tag{5.9}$$

$$k_{max}^h = k^h + 2 \quad (5.10)$$

If the original spectral peak $|\mathbf{X}(k_{min}^h : k_{max}^h, m)|$ has been decomposed into its dominant and secondary components, denoted here as $F_D(k)$ and $F_S(k)$, respectively, then the dual-peak model is used to generate an isolated approximation of the original spectral peak, hereafter referred to as the observed peak $O_T(k)$, defined by the following equation.

$$O_T(k) = \begin{cases} F_D(k) + F_S(k) & \text{if } |f_d - f_s| > 30\text{Hz} \\ |\mathbf{X}(k, m)| & \text{if } |f_d - f_s| < 30\text{Hz} \end{cases} \quad (5.11a)$$

$$(5.11b)$$

where f_d and f_s are the centre frequencies of the dominant and secondary components, respectively. Since the shape of the frequency component to be extracted is known from the previous separation stage, the notation $F_T(k)$ is used to indicate the estimated component of the spectral peak closest to the ideal centre frequency of the h -th harmonic partial, which can be either $F_D(k)$ or $F_S(k)$. Then, the corresponding section of the time-frequency mask is computed as follows.

$$\mathbf{M}_E(k_{min}^h : k_{max}^h, m) = \frac{F_T(k_{min}^h : k_{max}^h)}{O_T(k_{min}^h : k_{max}^h)} \quad (5.12)$$

where $\mathbf{M}_E(k, m)$ is the m -th frame of the time-frequency mask used to extract the predominant note event. When spectral peaks are well-spaced, the dual-peak approximation is very similar in shape to the observed spectral peak. But, when peaks are very close to each other, the dual-peak model provides a better fit of the time-frequency mask, in particular near the partial boundaries, and reduces the risk of errors by forcing the mask to be always within the range from 0 to 1.

Figure 5.12 shows the estimated extraction masks for two harmonic partials in a mixture involving viola and clarinet notes, previously presented in Figure 5.8, where the viola note is the one being extracted first. In this particular frame, Figure 5.12(a) shows the extraction of the fundamental harmonic of the viola, which is not interacting with any other partial, so the mask is designed to take all its energy out from the mixture. In Figure 5.12(b), the fundamental of the clarinet overlaps the second harmonic of the viola, which is detected as the secondary component of the spectral peak. Hence, the shape of the mask adapts to extract only the energy associated with the secondary component, while leaving the dominant one in the residual to allow the detection of the clarinet note in the following iteration. Considering the same example, the

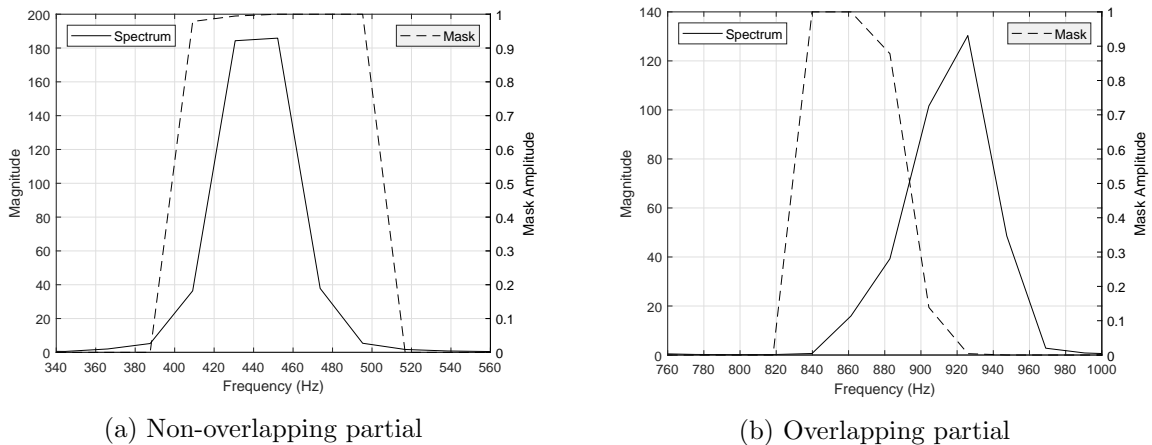


Figure 5.12: Extraction masks for two different frequency components from a mixture of viola (A4) and clarinet (A#5), where the viola has been selected as the predominant note event.

extraction mask of the first eight harmonic partials of the viola note are presented in Figure 5.13 while the extracted partials are also compared with the original harmonics of the viola note. A spurious interferer, detected at an average fundamental frequency of 880 Hz and having $\alpha = 0.25$, is causing amplitude changes in the extraction mask of the fourth and sixth harmonics in Figure 5.13(a). However, since the magnitudes of the affected harmonics are relatively low compared with the first two partials, it is very unlikely that this energy will trigger the detection of an unreal event at 880 Hz.

The iterative extraction of note events using time-frequency masks is demonstrated in Figures 5.14 and 5.15, for a mixture of two real notes: saxophone E3 and bassoon A2. Since both notes have relatively low fundamental frequencies (165 Hz and 110 Hz, respectively), the difficulty of fitting extraction masks for these note events is much higher in this case, due to the close proximity of harmonic partials and the low magnitude of the fundamental and second harmonics of the bassoon note. During the first iteration, the saxophone note is chosen as the predominant one, while two spurious interferers are detected at 330 Hz and 430 Hz, which are allocated preservation rates $\alpha_1 = \alpha_2 = 0.25$. The extraction mask is then generated for all selected peaks in every frame without explicitly knowing that the lower bassoon note is present. Figure 5.14(a) shows an example frame of the extraction mask, while Figure 5.14(b) compares the extracted partials with the original saxophone harmonics. The dual-peak model is used when the frequency components are not too close, while the preservation rates are used otherwise. Despite the increased complexity of the input spectrum, the extracted partials look very close to the original harmonics of the saxophone note. After removing the first note event from the mixture, the lower bassoon note is detected and its extraction mask is also generated, as

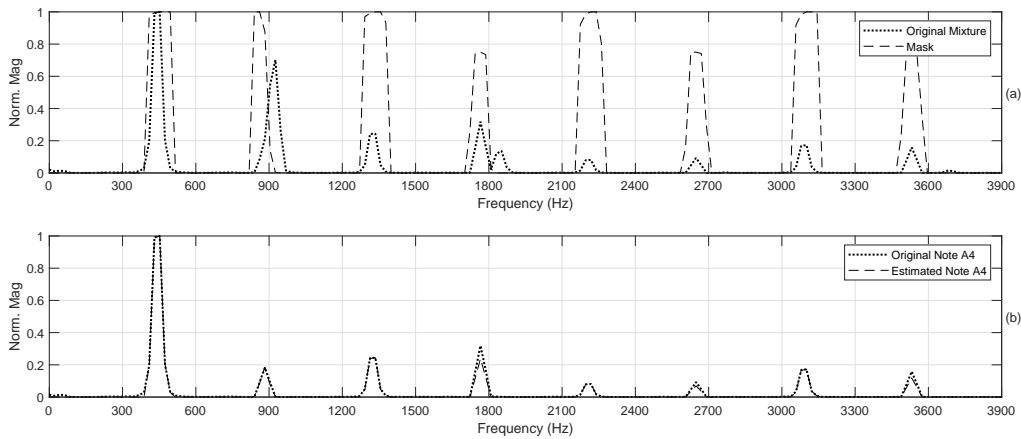


Figure 5.13: A single frame of the time-frequency mask used to extract a viola note A4 from a mixture in which the clarinet note A \sharp 5 is also present. (a) Spectrum of the input mixture and extraction mask. (b) Original and estimated spectra of the viola note A4.

presented in Figure 5.15(a), while the extracted partials in Figure 5.15(b) also show significant similarity with the original harmonics of the bassoon note.

In the same example, the principal differences between the original and estimated events come from the separation of fully-overlapping partials, where a set of arbitrary preservation rates are used to partition their energies. However, as these differences might affect the timbre quality of the separated events, the proposed strategy ensures that at least a small amount of energy is left at the right frequencies to enforce the detection of any other real note present in the mixture.

Since the shape of every spectral peak is affected by the phase interaction with other components, significant changes in shape are expected to happen when moving from one frame to the next one, making the processing of some frames harder than it is for others. For this reason, the algorithm is expected to produce erroneous mask shapes for some harmonic partials in some particular frames, which tend to occur in bursts of no more than two or three consecutive frames for one particular partial. However, these problems can be significantly reduced if the final time-frequency mask is median-filtered using a short window. In this work, a median filter of length 5 has been selected and applied to smooth out some of the errors. The estimation of the time-frequency mask used for extraction, is summarised in Algorithm 2.

In order to separate the predominant note event from within the mixture, the input magnitude spectrogram $\mathbf{X}(k, m)$ is multiplied by the extraction mask $\mathbf{M}_E(k, m)$, and the result is converted back to the time domain by means of the ISTFT, where the phase information is taken directly from the input mixture, as stated in the following equation.

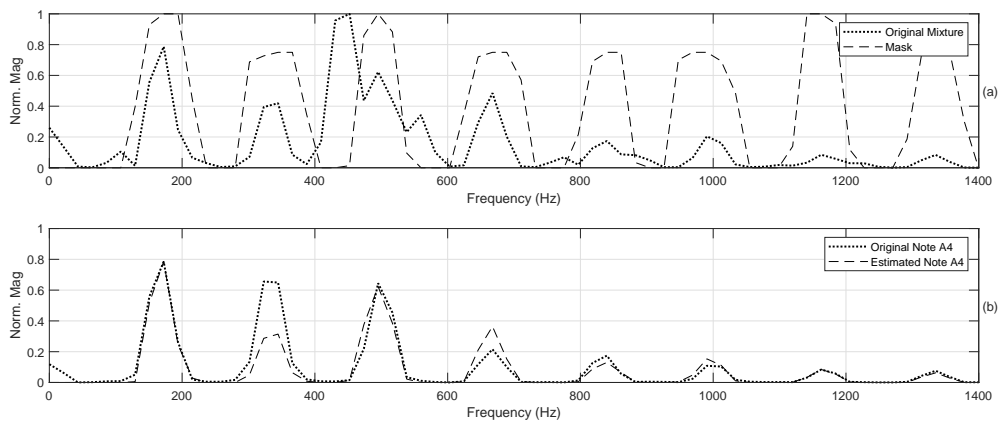


Figure 5.14: First note event extraction from a mixture of saxophone and bassoon. (a) Spectrum of the mixture and extraction mask for the saxophone note E3. (b) Original and estimated spectra of the saxophone note. Spurious interferers are present at 330 Hz and 430 Hz (with preservation rates $\alpha_1 = \alpha_2 = 0.25$).

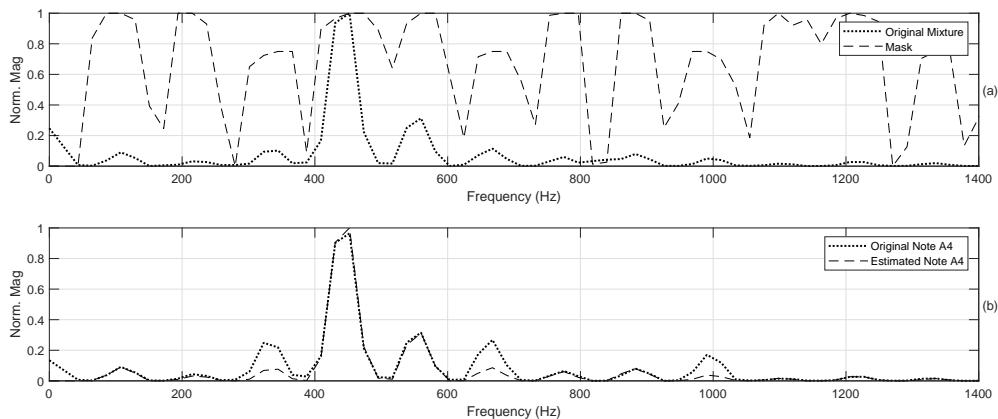


Figure 5.15: Second note event extraction from a mixture of saxophone and bassoon, after removing the saxophone note. (a) Input spectrum and extraction mask for the bassoon note A2. (b) Original and estimated spectra of the bassoon note. A spurious interferer is present at 330 Hz (with preservation rate $\alpha = 0.25$).

$$\mathbf{Ne}_v(t) = \text{ISTFT}\{|\mathbf{X}(k, m)| \cdot \mathbf{M}_E(k, m), \angle \mathbf{X}(k, m)\} \quad (5.13)$$

Where $\mathbf{Ne}_v(t)$ is the time-domain signal associated with the predominant note event detected in the v -th iteration of the system, which is kept in a memory structure during the rest of the iterative stage, along with other note events extracted in previous iterations. The residual energy of the input mixture that is not assigned to the current note event is also separated with a different time-frequency mask, which can be obtained from the current extraction mask as follows.

Algorithm 2 Time-Frequency Masking Algorithm

```

1: function TIME-FREQUENCY MASKING
   Input: Mixed Spectrogram  $\mathbf{X}_{(K \times M)}$ , Centre Frequency Bins of Selected Peaks  $\mathbf{C}_{(H \times M)}$ ,
   Pitch Trajectory  $\{\mathcal{P}_{\mathcal{T}}\}_{(1 \times M)}$ , Maximum Number of Harmonics to Extract  $H_q$ , Preservation
   Rates  $A_{(H \times M)}$ , and Sampling Frequency  $f_s$ .
   Output: Time-Frequency Extraction Mask  $\{\mathbf{M}_E\}_{(K \times M)}$ 
2:    $\mathbf{M}_E \leftarrow \text{zeros}(K, M)$  ▷ Initializing extraction mask
3:    $[m_a, m_b] \leftarrow \text{Find-Limits}(\mathcal{P}_{\mathcal{T}})$  ▷ Start and end of pitch trajectory
4:   for  $m = m_a$  to  $m_b$  do
5:      $f_0 \leftarrow \mathcal{P}_{\mathcal{T}}(m)$  ▷ Current pitch estimate
6:      $h \leftarrow 1$  ▷ Harmonic counter
7:      $f_h \leftarrow f_0$  ▷ Harmonic frequency
8:     while  $(h \leq H_q)$  AND  $(f_h < \frac{f_s}{2})$  do
9:        $c_b \leftarrow \mathbf{C}(h, m)$  ▷ Centre bin of selected peak
10:       $[F_D(:), F_S(:), Dis] \leftarrow \text{Parsons-Method}(|\mathbf{X}(c_b, m)|)$  ▷ Peak decomposition
11:      if  $Dis < 30$  Hz then ▷ Peak classification
12:         $F_T(:) \leftarrow A(h, m) \times F_D(:)$  ▷ Fully-Overlapping
13:         $O_T(:) \leftarrow \mathbf{X}(:, m)$ 
14:      else
15:         $F_T(:) \leftarrow \text{Find-Closest}([F_D(:), F_S(:)], f_h)$  ▷ Semi-Overlapping
16:         $O_T(:) \leftarrow F_D(:) + F_S(:)$ 
17:      end if
18:       $c_b \leftarrow \text{Peak-Position}(F_T(:))$  ▷ Updating centre bin
19:       $[k_{min}^h, k_{max}^h] \leftarrow [c_b - 2, c_b + 2]$  ▷ Current peak limits
20:       $\mathbf{M}_E(k_{min}^h : k_{max}^h, m) \leftarrow \frac{F_T(k_{min}^h : k_{max}^h)}{O_T(k_{min}^h : k_{max}^h)}$  ▷ Extraction mask of current peak
21:       $h \leftarrow h + 1$ 
22:       $f_h \leftarrow h \times f_0$ 
23:    end while
24:  end for
25:   $\mathbf{M}_E \leftarrow \text{Median-Filtering}(\mathbf{M}_E, [m_a : m_b], 5)$  ▷ Smoothing extraction mask
26:  return  $\mathbf{M}_E$  ▷ Extraction mask
27: end function
    
```

$$\mathbf{M}_R(k, m) = 1 - \mathbf{M}_E(k, m) \quad (5.14)$$

After extracting the predominant note event, the residual signal in the v -th iteration, denoted as $\text{Re}_v(t)$, is computed as in Equation 5.15.

$$\text{Re}_v(t) = \text{ISTFT}\{|\mathbf{X}(k, m)| \cdot \mathbf{M}_R(k, m), \angle \mathbf{X}(k, m)\} \quad (5.15)$$

Within the proposed iterative framework, the residual signal is used as the input for the next iteration, from which a new predominant note event might be detected and extracted. As stated in Chapter 4, this cycle continues until a predefined maximum number of iterations is reached, or until the energy in the residual signal is below a significance threshold.

5.6.2 Time-Domain Subtraction

The extraction method presented in the previous section is particularly effective when harmonic partials are well spaced. However, if the average fundamental frequency of the predominant note event is low (usually below 200 Hz), the distance between centre frequencies of its harmonic partials is so small that any other component associated with a different source might produce complex shared peaks, as a result of more than two heavily overlapping partials. Finding the limits of these complex peaks is difficult and the extraction mask cannot always be obtained accurately.

When the shape of the extraction mask is inadequate for a particular harmonic partial, the extracted peak might show significant distortion that could affect the overall quality of the reconstructed event. The method presented in this section, which is based on time-domain subtraction, replaces the extraction mask with a different approach, in which the separated magnitude spectrogram of the predominant note event is estimated directly from individual synthetic partials, chosen just after decomposing each selected peak with Parsons' method.

The outcome of the process is a magnitude spectrogram that can be converted back to the time domain in order to recover the separated note event, while the residual is generated by subtracting this note event from the input mixture in the time domain. The main advantage of the process is that each component in the separated spectrogram should preserve its Hanning-windowed shape throughout the whole separation process, reducing the levels of distortion in the separated signals, independently of the frequency distance to other nearby components.

This alternative approach is very similar to the one discussed in Section 5.6.1, where every selected peak in array \mathbf{C} is decomposed into its dominant and secondary components, following Parsons' method. Depending on the distance between their centre frequencies, a decision is made about whether to handle them as a set of semi-overlapping or fully-overlapping partials. If the peak is due to a group of semi-overlapping partials, the component closest to the ideal harmonic frequency is chosen as the target partial, otherwise an attenuated version of the dominant component is selected. However, instead of using the target partial to construct an extraction mask, it is directly incorporated as an element of the separated magnitude spectrogram of the predominant note event, denoted as $|\mathbf{Y}(k, m)|$, which avoids having to compute suitable boundaries for the observed shared peak. Algorithm 3 summarises this alternative method.

Algorithm 3 Time-Domain Subtraction Algorithm

```

1: function TIME-DOMAIN SUBTRACTION
   Input: Mixed Spectrogram  $\mathbf{X}_{(K \times M)}$ , Centre Frequency Bins of Selected Peaks  $\mathbf{C}_{(H \times M)}$ ,
   Pitch Trajectory  $\{\mathcal{P}_T\}_{(1 \times M)}$ , Maximum Number of Harmonics to Extract  $H_q$ , Preservation
   Rates  $A_{(H \times M)}$ , and Sampling Frequency  $f_s$ .
   Output: Separated Magnitude Spectrogram of Note Event  $|\mathbf{Y}|_{(K \times M)}$ 
2:    $\mathbf{Y} \leftarrow \text{zeros}(K, M)$  ▷ Initializing output spectrogram
3:    $[m_a, m_b] \leftarrow \text{Find-Limits}(\mathcal{P}_T)$  ▷ Start and end of pitch trajectory
4:   for  $m = m_a$  to  $m_b$  do
5:      $f_0 \leftarrow \mathcal{P}_T(m)$  ▷ Current pitch estimate
6:      $h \leftarrow 1$  ▷ Harmonic counter
7:      $f_h \leftarrow f_0$  ▷ Harmonic frequency
8:     while  $(h \leq H_q)$  AND  $(f_h < \frac{f_s}{2})$  do
9:        $c_b \leftarrow \mathbf{C}(h, m)$  ▷ Centre bin of selected peak
10:       $[F_D(\cdot), F_S(\cdot), Dis] \leftarrow \text{Parsons-Method}(|\mathbf{X}(c_b, m)|)$  ▷ Peak decomposition
11:      if  $Dis < 30$  Hz then ▷ Peak classification
12:         $F_T(\cdot) \leftarrow A(h, m) \times F_D(\cdot)$  ▷ Fully-Overlapping
13:      else
14:         $F_T(\cdot) \leftarrow \text{Find-Closest}([F_D(\cdot), F_S(\cdot)], f_h)$  ▷ Semi-Overlapping
15:      end if
16:       $|\mathbf{Y}(:, m)| \leftarrow |\mathbf{Y}(:, m)| + F_T(\cdot)$  ▷ Updating output spectrogram
17:       $h \leftarrow h + 1$ 
18:       $f_h \leftarrow h \times f_0$ 
19:    end while
20:  end for
21:  return  $\mathbf{Y}$  ▷ Separated spectrogram of note event
22: end function

```

A time-domain signal $\text{Ne}_v(t)$, corresponding to the predominant note event in the v -th iteration, is obtained by inverse transforming the separated magnitude spectrogram $|\mathbf{Y}(k, m)|$ using the phase information of the input mixture, as presented in Equation 5.16.

$$\text{Ne}_v(t) = \text{ISTFT}\{|\mathbf{Y}(k, m)|, \angle \mathbf{X}(k, m)\} \quad (5.16)$$

The residual signal in the v -th iteration is obtained by subtracting the estimated predominant note event $\text{Ne}_v(t)$ from the current input mixture $x_v(t)$, as defined in Equation 5.17.

$$\text{Re}_v(t) = x_v(t) - \text{Ne}_v(t) \quad (5.17)$$

As the phase spectrum of the mixture is used to reconstruct the separated note event, a risk of producing a distorted waveform in the time domain always exists. However, it is believed that the quality of the reconstruction depends more on the accuracy of the centre frequencies and magnitudes of the separated harmonic partials than on the phase spectrum. Hence, any deviation of the phase information should not be critical if the accuracy of the estimated centre

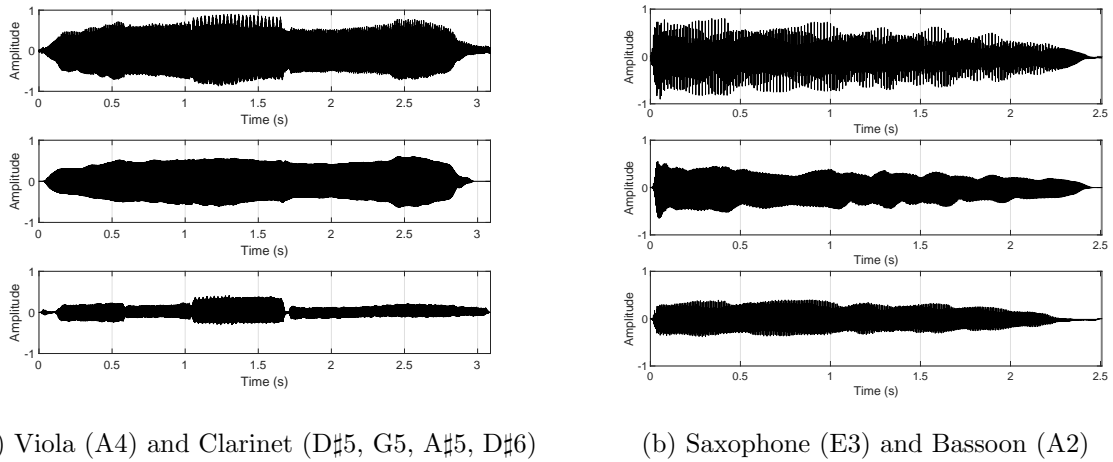


Figure 5.16: Time domain subtraction applied to a set of audio mixtures with polyphony two. (Top) Input mixture. (Middle) Predominant note event. (Bottom) Residual of the first iteration.

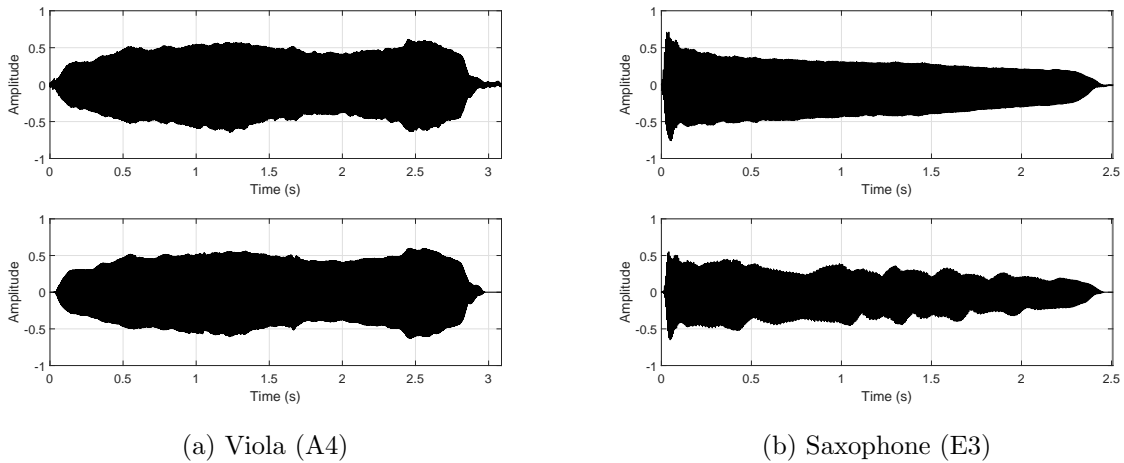


Figure 5.17: Comparison between the extracted note events in Figure 5.16 and the original ones. (Top) Original note. (Bottom) Extracted note event.

frequencies and magnitudes is acceptable. As in Section 5.6.1, the residual signal is used as the input mixture in the next iteration.

Two examples are presented in Figures 5.16 and 5.17, in which the time domain subtraction is used to extract the predominant note event from within two different audio mixtures. First, a mixture of viola and clarinet is analysed in Figure 5.16(a), where the long viola note A4 is selected as the predominant note event. As it is extracted, the residual now shows four underlying clarinet notes which are extracted individually in later iterations. In Figure 5.16(b) the mixture consists of a saxophone note E3 and a bassoon note A2, where the saxophone is chosen as the predominant event. After extraction, the residual also shows the underlying note A2. In each case, the extracted note events are compared with the original notes in Figure 5.17, where the

extracted saxophone note shows more distortion than the extracted viola. But, the reason for this is mainly due to the fundamental frequencies of the notes present in every mixture, and not on the inaccuracy of the phase spectra used to reconstruct the events. Separating the spectral content of low-pitched notes is significantly more difficult, since the quality of the estimated centre frequencies and magnitudes deteriorates as partials get closer. In the first mixture, the distance between harmonics allows a sharp separation of overlapping partials, while in the second one, shared peaks involving more than two components are more likely. However, the quality of the separation is still acceptable and the algorithm is much faster than the time-frequency masking algorithm.

5.6.3 Time-Frequency Masking vs. Time-Domain Subtraction

The differences between the aforementioned extraction methods are very subtle, but their accumulated effects can introduce variations in separation performance. A specific problem of time-frequency masks is that they usually exhibit rapid transitions in between peak and non-peak regions and hence may introduce discontinuities in the magnitude spectrogram of the residual signal. During the next iteration, these discontinuities increase the chances of detecting spurious note events, which in reality are associated with previously extracted events. When time-domain subtraction is used, separated partials are not clipped as they are incorporated into the estimated magnitude spectrogram, which tends to create smoother residuals.

A comparison between the two extraction methods is presented in Figure 5.18, where the harmonic content of a cello note D2 in isolation is extracted using both alternatives. A single frame of the separated spectra is presented in Figure 5.18(a) and compared with the same frame of the original note, while the spectra of the corresponding residuals are shown in Figure 5.18(b). The amount of energy extracted is approximately the same for both methods, but the residual obtained with time-domain subtraction looks smoother than the one produced with time-frequency masking, which exhibits lower minima and sharper peaks. While time-frequency masking can be seen as a harder separation process, time-domain subtraction represents a softer alternative similar to Wiener filtering, but it does not require information about all components in the mixture. Further comparison of these separation strategies is presented in Section 5.8.

5.6.4 Number of Harmonics Extracted per Frame

The number of harmonic partials extracted in every frame depends on the average fundamental frequency of the predominant note event. When the predominant pitch is above

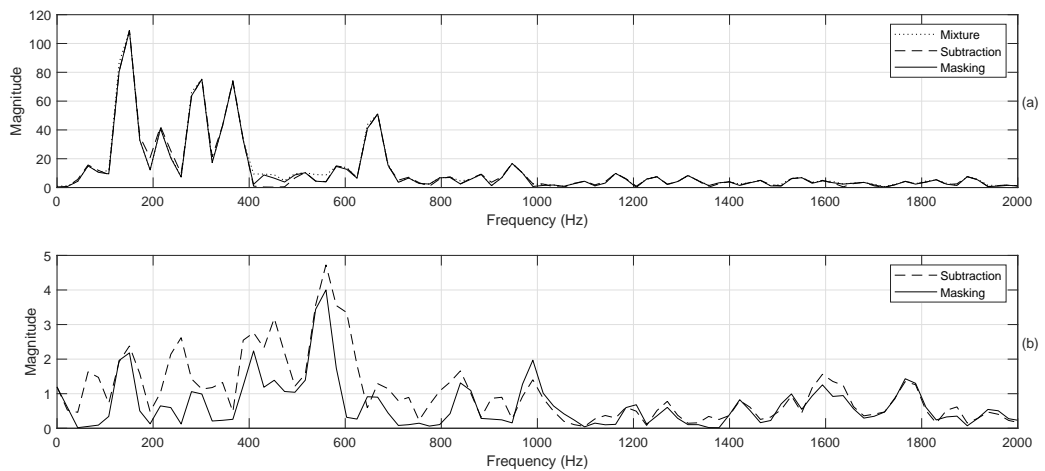


Figure 5.18: Extraction of the spectral content associated with a cello note (D2) in isolation using time-domain subtraction and time-frequency masking. (a) Spectra of the original and estimated cello notes (using both extraction methods). (b) Spectra of the corresponding residual signals.

200 Hz, the distance between harmonics is usually larger than five frequency bins, which provides enough room to identify the relevant set of spectral peaks, and harmonic partials can be tracked up to a maximum number H_q , or up to the Nyquist frequency.

However, if the note event has fundamental frequency below 200 Hz, its harmonic partials suffer significant overlap, complicating the identification of relevant peaks. There is also a high probability of observing distortion in the magnitude spectrum due to phase interaction between harmonics of the selected note event and other partials associated with concurrent sources. In this case, to reduce the risk of severely damaging other frequency components during the extraction of a low-pitched note event, the separation is conducted by extracting a reduced set of harmonics. Hence, during the separation of the q -th note event, with average fundamental frequency $f_{(0,q)}$, the maximum number of harmonic partials to be extracted in every frame is defined by the following rule.

$$H_q = \begin{cases} H_{max} & \text{if } f_{(0,q)} > 200\text{Hz} \\ H_{min} & \text{if } f_{(0,q)} \leq 200\text{Hz} \end{cases} \quad (5.18a)$$

$$(5.18b)$$

where H_{max} and H_{min} are parameters of the system that can be defined by the end-user. After extensive experimentation, the values $H_{max} = 30$ and $H_{min} = 10$ have been found to be effective in terms of the amount of energy extracted in every iteration, and the amount of damage introduced when handling low-pitched note events.

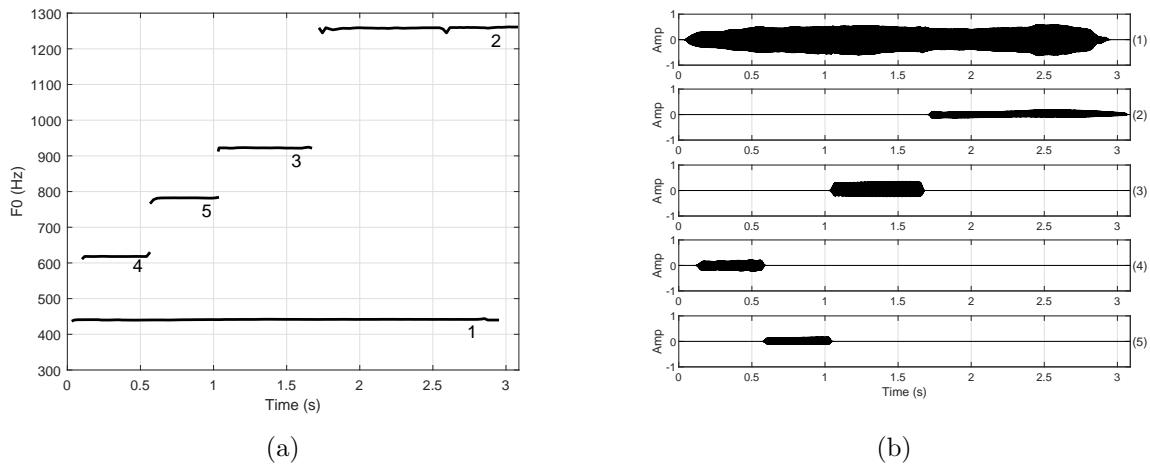


Figure 5.19: Clustering of estimated note events from a mixture of viola and clarinet. (a) Estimated pitch trajectories of note events. (b) Extracted note events: (1) Viola A4, (2) Clarinet D \sharp 6, (3) Clarinet A \sharp 5, (4) Clarinet D \sharp 5, and (5) Clarinet G5. The extraction order is shown with numbers.

5.7 Clustering of Note Events

At the end of the iterative stage, most of the energy contained in the original mixture should have been allocated within a set of note events, which can be clustered to form individual sources by the end-user, who may use the pitch trajectories of the separated note events as a hint to find an appropriate clustering of the events. The end-user can also listen to each individual note event in order to obtain further guidance. Grouping or instrument identification algorithms could be used at this stage to remove the need for user input, but are not the emphasis of this research.

Using the same example mixture of viola and clarinet, previously presented in Figure 5.16, the final set of estimated pitch trajectories is presented in Figure 5.19(a), and their corresponding extracted note events, obtained in this case by means of time-domain subtraction, are shown in Figure 5.19(b). The final residual signal is shown in Figure 5.20, whose energy content is mostly associated with transients or with any other harmonic structure that was not extracted during the iterative stage. For example, the energy content between $t = 1$ s and $t = 1.3$ s relates to an interferer that was erroneously labelled as a real note during the extraction of note event 3. However, the energy associated with this spurious interferer was not significant enough to trigger a sixth iteration and, hence, the energy ended up in the final residual.

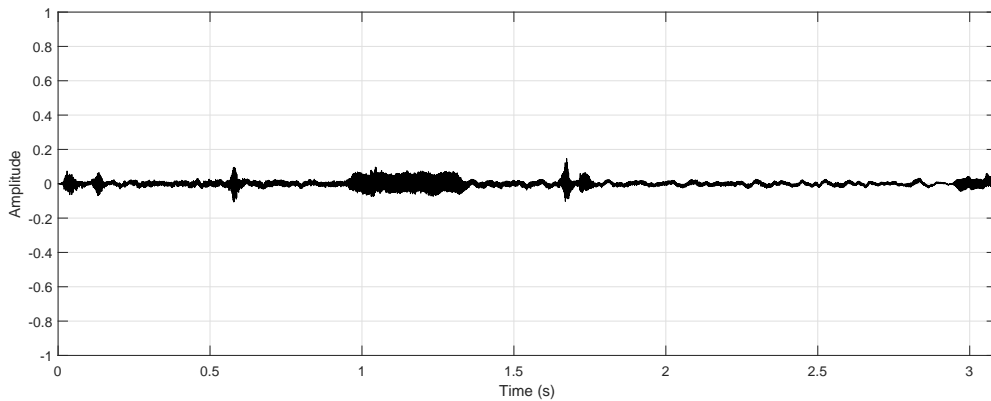


Figure 5.20: Residual signal obtained after extracting all five note events presented in Figure 5.19(b).

The end-user is now able to cluster note events 2, 3, 4 and 5 in order to form the separated clarinet track, while note event 1 is used to form the separated viola track. A comparison between the original and estimated sources is presented in Figure 5.21.

5.8 Evaluation of Performance

To evaluate the relative performance of the proposed semi-automatic separation systems, they are applied in turn to the task of separating the underlying sources from within a number of audio mixtures. In every experiment, performance is measured in accordance with the evaluation method presented in Section 3.7.1, which delivers the overall source-to-distortion ratio (SDR), source-to-interference ratio (SIR), and source-to-artifact ratio (SAR). In every test mixture, performance measures are calculated using the BSS Eval toolbox [98], then averaged across the estimated sources, and finally reported in decibels (dB). Box plots are used to present the results in every experiment, in which the minimum, first quartile (25% of the data), median, third quartile (75% of the data) and maximum are displayed in a box-and-whisker configuration. A marker (+) is used to represent any outliers in the data.

In every case, note events are automatically detected using the pitch tracking system presented in Chapter 4, while their extraction from within the mixture is conducted using the two methods presented in Section 5.6, namely, Time-Frequency Masking (IES-TFM) and Time-Domain Subtraction (IES-TDS). The clustering of note events is conducted by the end-user, as described in the previous section, who also determines the maximum number of note events (Θ) to be extracted for any particular input mixture. All test recordings are sampled at 44.1 kHz and analysed using a 46 ms Hanning window with 5.8 ms hop-size during the separation stage.

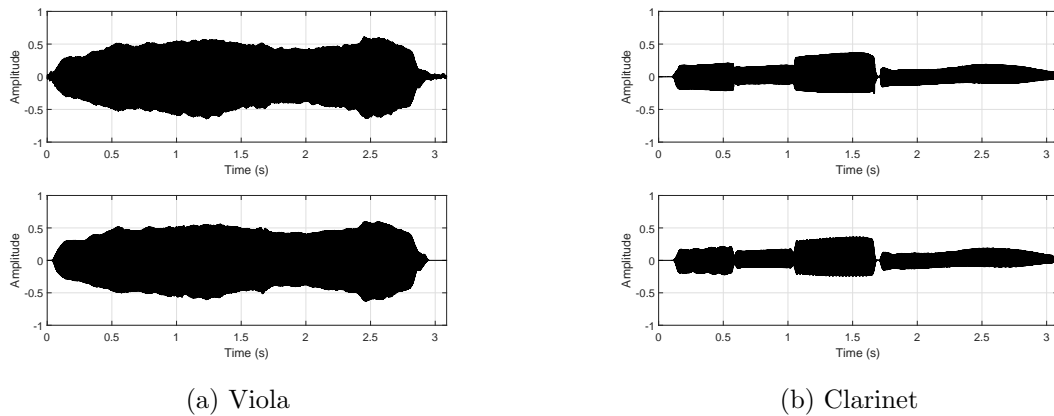


Figure 5.21: Comparison between the original and estimated sources from a mixture of viola and clarinet. (Top) Original source, (Bottom) Estimated source.

In each frame, the maximum number of harmonic partials to be extracted (H_q) has been set according to Equations (5.18a) and (5.18b), while other parameters associated with the multipitch estimator have been set as in Chapter 4.

5.8.1 Seven Simultaneous Violin Notes

In this experiment, a set of seven violin notes were anechoically recorded and combined to form six mixtures with polyphonies from 2 to 7. With ascending order, the notes involved are F5, Ab5, A5, B5, Db6, E6 and Gb6. These recordings are available online¹.

Although the selected notes are not related to each other by harmonic intervals, such as major thirds, fourths, fifths or octaves, the level of complexity exhibited by each of the mixtures gradually increases as the the number of simultaneous violin notes goes from 2 to 7, which also increases the difficulty of detecting continuous pitch trajectories and separating their corresponding harmonic structures. Results for this experiment are presented in Table 5.2.

For comparison, Oracle separation results have also been calculated for each of the mixtures using the BSS Oracle toolbox, developed by Vincent et al. [131], which is available online². In particular, the function used to generate these metrics has the name *bss_nearopt_monomask*, which generates near-optimal time-frequency masks for single-channel source separation using the STFT with a sine window [132]. Oracle results are, theoretically, the highest achievable separation results that can be obtained using time-frequency masking-based methods, but can only be obtained when the reference sources are available, serving as an upper bound in performance evaluation [26].

¹<https://doi.org/10.5281/zenodo.3478442>

²http://bass-db.gforge.inria.fr/bss_oracle/

Table 5.2: Performance metrics for the separation of 2-7 synchronous violin notes.

| Polyphony | IES-TFM | | | IES-TDS | | | Oracle | | |
|-----------|---------|-------|-------|---------|-------|-------|--------|-------|-------|
| | SDR | SIR | SAR | SDR | SIR | SAR | SDR | SIR | SAR |
| 2 | 21.80 | 33.10 | 24.04 | 23.76 | 32.51 | 26.20 | 27.67 | 32.79 | 29.44 |
| 3 | 21.22 | 28.81 | 23.11 | 22.11 | 29.14 | 24.09 | 26.76 | 32.76 | 28.12 |
| 4 | 24.10 | 30.25 | 25.53 | 22.37 | 29.14 | 23.81 | 25.59 | 32.45 | 26.62 |
| 5 | 23.21 | 28.45 | 25.35 | 23.25 | 28.67 | 25.14 | 24.48 | 30.54 | 25.82 |
| 6 | 18.03 | 21.89 | 21.14 | 17.82 | 21.44 | 20.93 | 20.28 | 27.20 | 21.56 |
| 7 | 18.30 | 22.28 | 21.10 | 17.61 | 21.74 | 20.41 | 19.92 | 26.85 | 21.10 |

From the data in Table 5.2, it can be seen that both algorithms exhibit slightly different levels of performance and, in both cases, these metrics are very close to the Oracle results, which provide evidence for the overall quality of the separated sources. An effective separation of the spectral content in both cases can also be noticed by the high SIRs obtained, while the corresponding SARs indicate that only small artefacts are being introduced by the process. Since the proposed method is not a binary masking strategy, where the energy of a single time-frequency tile is allocated to only one source, it is possible for the proposed system even to outperform the Oracle results, as in polyphony 2, where the IES-TFM method outperforms the Oracle results in terms of SIR.

While the IES-TFM process shows better performance levels at high polyphonies, the IES-TDS variation is more effective at lower levels of polyphony. However, their overall performance does not decrease monotonically as the polyphony increases, which is a trend that the Oracle results do exhibit. The reason for this is due to the order in which note events are extracted from within the mixture, which is defined automatically based on an accumulated measure of salience that depends on the shape of the input spectrum. Since the extraction methods are not the same, the generated residuals present subtle differences that change the salience measures during the following iteration, where these residuals are used as the new set of input mixtures, and consequently change the selection of the next predominant note event.

Considering the mixture of seven simultaneous violin notes, results obtained by the IES-TFM and IES-TDS systems are also compared with two previously presented algorithms in Figure 5.22. The first alternative process, denoted as MIDI, corresponds to the MIDI-informed separation algorithm by Every [28], while the second one is the iterative residual-based process by Siamantas [23], denoted as IDG, in which the automatic multipitch estimator by Klapuri [113] is used. In this comparison, the IES-TFM and IES-TDS systems exhibit higher separation performances than the MIDI process, not only in terms of the median values but also in terms

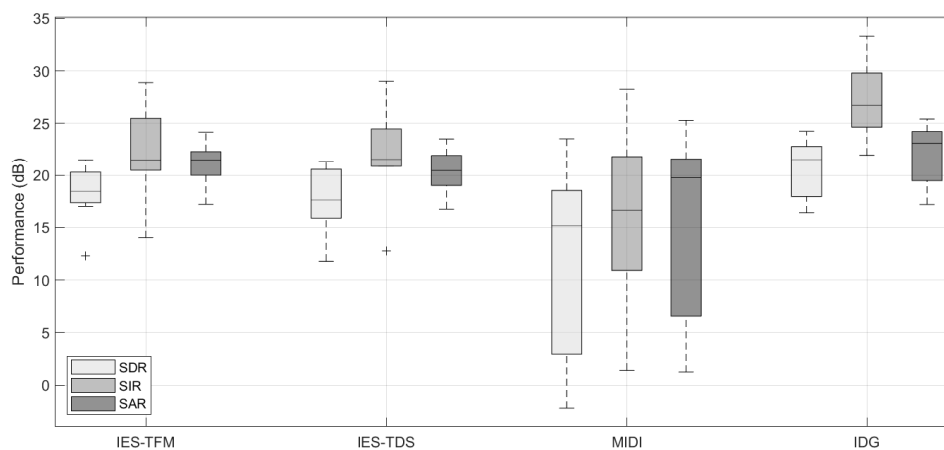


Figure 5.22: Separation performance for a mixture of seven synchronous violin notes of the same duration using four different algorithms: the proposed system with time-frequency masking (IES-TFM), the proposed system with time-domain subtraction (IES-TDS), the MIDI-informed system in [28] (MIDI), and the iterative residual-based system in [23] (IDG).

of the consistency of these results. The IDG system, on the other hand, presents slightly higher levels of performance in this experiment, due to its specific design which allows it to cope particularly well with mixtures of individual sustained notes. Considering that the proposed algorithm is specifically designed to process audio mixtures where the underlying sources are allowed to play more complicated melodies, an average difference in this experiment of less than 2dB in SDR represents a very positive outcome.

5.8.2 Influence of the Extraction Order

The experiment presented in the previous section suggests that the order in which note events are extracted influences the separation performance. Therefore, an additional experiment is conducted here to further explore the impact of the extraction order, using a set of 12 audio mixtures, each consisting of 3 simultaneous musical notes³. Each set represents a different instrument, such as violin, double bass, flute, piccolo and bassoon, while the notes in every mixture are played by the same instrument. These test recordings were selected from the RWC musical instrument sound database [129], considering a normal-forte playing style in every case, while their pitches were selected to avoid harmonically-related notes within the same mixture.

Three different orders of extraction are defined and denoted as A, B and C. Order A forces the system to extract note events from the lowest pitch to the highest. Order B does exactly the opposite, it forces the extraction to be from the highest pitch to the lowest one. Finally, order

³<https://doi.org/10.5281/zenodo.3478455>

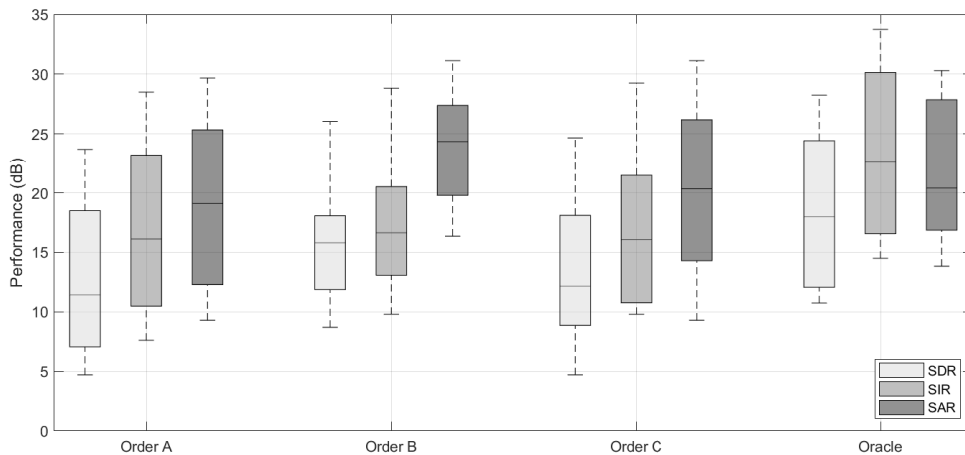


Figure 5.23: Separation performance for 12 mixtures of single musical notes using the proposed time-frequency masking approach and three note-event extraction orders.

C represents the extraction order that the system would automatically select based on salience measurements. The proposed system is applied to the separation of the underlying sources considering both extraction methods (time-frequency masking and time-domain subtraction), a fixed maximum of harmonic partials extracted in every frame, and ground-truth pitch trajectories, which were previously estimated for each musical note in isolation by means of the same pitch tracking system presented in Chapter 4. The corresponding separation results are shown in Figure 5.23 for the system that uses time-frequency masking for extraction, and in Figure 5.24 for the system that uses time-domain subtraction. In both cases, Oracle estimates are also included for comparison.

Results in Figures 5.23 and 5.24 show that the separation performance tends to exhibit higher variations when low-pitched note events are extracted first, while starting with the highest ones seems to produce better average results. When the system was allowed to select the extraction order, it chose Order A four times, Order B four times, and for the rest of the mixtures, it decided to extract the events in a different order. In those particular cases, the separation performance was higher than Order A and B on two occasions, lower than Order A and B on one occasion, and in between Order A and B on one occasion. In terms of the method used to extract note events, time-domain subtraction performs slightly better than time-frequency masking in all three extraction orders, while the average performance of the systems is approximately 5.5 dB below the Oracle estimates in terms of SDR.

Choosing the order of extraction might be possible only in test cases where the number of simultaneous notes and their pitch contours are known in advance. In other circumstances, the

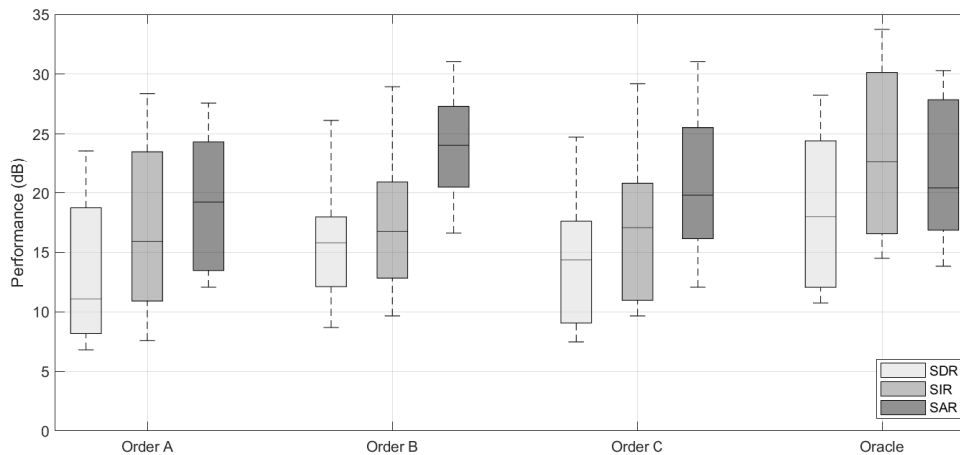


Figure 5.24: Separation performance on the same 12 mixtures of single musical notes using the proposed time-domain subtraction approach and three note-event extraction orders.

system has to deal with incomplete or partially incorrect information provided by multipitch detectors and operate in blind conditions. Hence, using salience measurements is possibly the safest way to determine which event to extract next. When the iterative estimation stage is complete, the process could be started again and the separation could be attempted in a different order to increase the quality. However, the main objective of this section is to show that the order of extraction does not have a high impact on the separation performance, thus, an automatic order selection scheme based on salience measurements constitutes an effective strategy for an initial separation framework such as the one presented in this work.

5.8.3 Influence of the Automatic Pitch Tracking Stage

Separation performance was measured in the previous section using ground-truth pitch trajectories to guide the extraction of note events from within 12 mixtures with polyphony 3. It is also important to evaluate the separation performance when the system has to estimate these pitch contours automatically. Performance measures in Figure 5.25 correspond to the separation of the underlying sources from the same set of audio mixtures considered in Section 5.8.2, but using a combination of the proposed iterative pitch detection stage and each of the extraction methods. In this case, the system chooses the predominant note event automatically in every iteration and estimates its pitch contour without any previous knowledge of the corresponding audio mixtures.

Moving from the ground-truth trajectories to the automatically detected ones represents an average reduction of 1 dB in SDR for both extraction methods, as shown in Figure 5.25. This

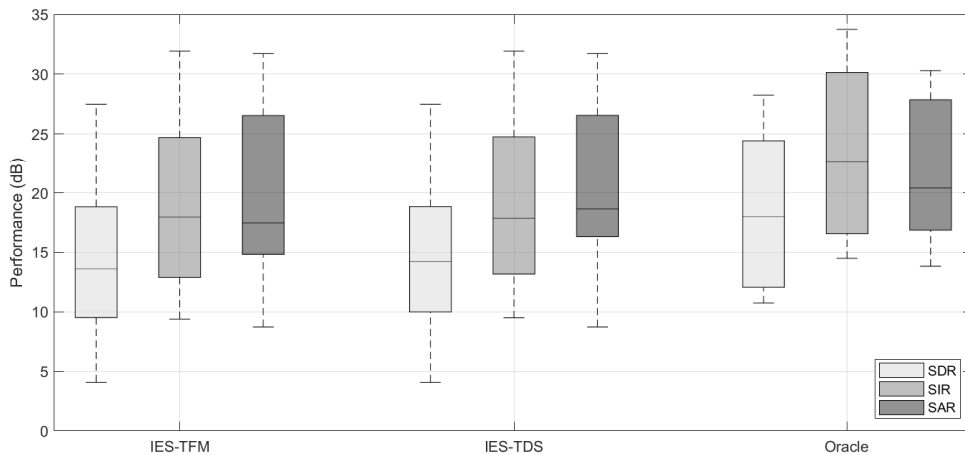


Figure 5.25: Separation performance on 12 mixtures of single notes using a combination of the proposed iterative pitch detection stage and each of the extraction approaches: time-frequency masking (IES-TFM) and time-domain subtraction (IES-TDS).

reduction is considered small for the significantly increased complexity of the task. Another effect associated with the use of automatic pitch tracking is that some of the original notes are extracted as groups of smaller note events, that have to be clustered by the end-user to reconstruct the estimated note. Usually, this situation does not represent a significant degradation of the separation quality, but in order to keep an acceptable performance, the system is required to deliver all sections of the original note, requiring longer processing times due to the additional iterations that have to be executed.

5.8.4 Influence of the Number of Harmonics Extracted per Frame

The final part of this experiment evaluates the impact of varying the maximum number of harmonic partials extracted per frame in accordance with the average pitch of the predominant note event. All previous results in this chapter were generated by considering a maximum of 30 harmonic partials to be extracted in every frame. However, if the same maximum is used during the extraction of note events with average fundamental frequency below 200 Hz, the risk of damaging other harmonic structures nearby is high. Hence, a pitch-dependent maximum (Section 5.6.4) is used as a way to cope with this situation and to improve the detection of other note events in later iterations.

The impact of using a pitch-dependent maximum is evaluated on the same set of audio mixtures used in Sections 5.8.2 and 5.8.3, while results are presented in Figures 5.26 and 5.27, for both extraction approaches. The new performance measures are compared with previously

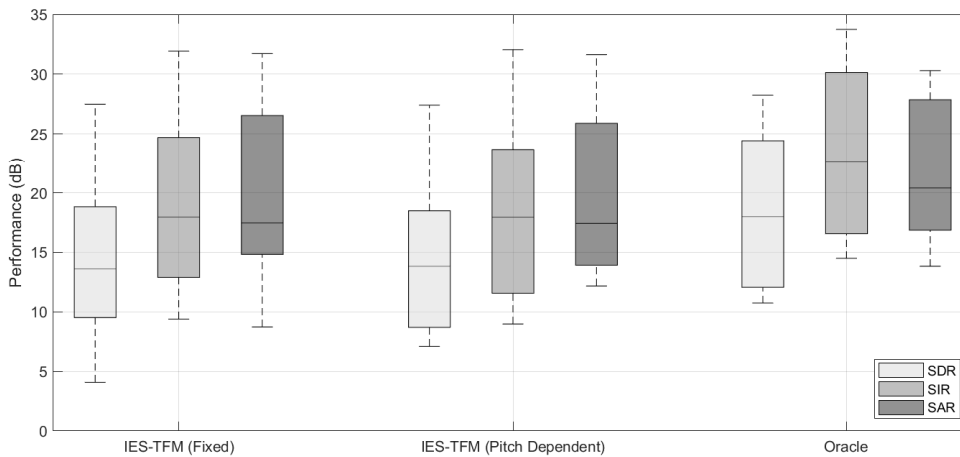


Figure 5.26: Separation performance exhibited by the proposed system on 12 mixtures of single notes, using time-frequency masking for extraction and two different ways to define the maximum number of harmonics extracted in every frame: Fixed and Pitch Dependent.

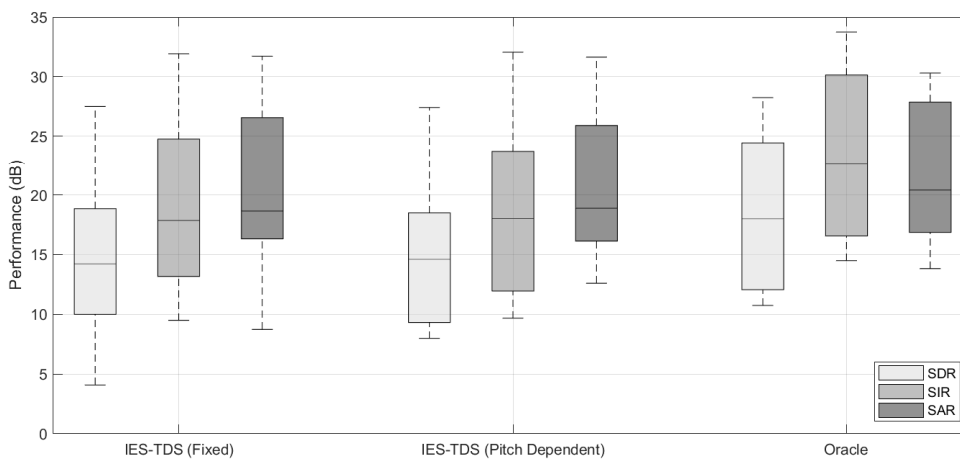


Figure 5.27: Separation performance exhibited by the proposed system on 12 mixtures of single notes, using time-domain subtraction for extraction and two different ways to define the maximum number of harmonics extracted in every frame: Fixed and Pitch Dependent.

obtained results using the fixed-maximum approach. The principal effect of using a pitch-dependent maximum is the reduction in the amount of artifacts that are introduced by the extraction process, which represents an average improvement of 0.8 dB in SAR for both variations of the system.

5.8.5 Source Separation in Real Music

In this section, separation performance is evaluated on a set of test recordings consisting of real music, where the underlying sources are different musical instruments which are allowed to

Table 5.3: Characteristics of the selected test recordings used in Section 5.8.5.

| Group | Details |
|-------|--|
| 1 | 12 mixtures with polyphony 2 involving violin and clarinet, with fundamental frequencies in the range from 225 Hz to 750 Hz. A total of 254 musical notes are present. |
| 2 | 12 mixtures with polyphony 3 involving violin, clarinet and tenor saxophone, with fundamental frequencies in the range from 175 Hz to 750 Hz. A total of 386 musical notes are present, including harmonically-related notes. |
| 3 | 12 mixtures with polyphony 3 involving violin, clarinet and percussion, with fundamental frequencies in the range from 225 Hz to 750 Hz. A total of 254 musical notes are present, as well as several hundred percussive events. |
| 4 | 12 mixtures with polyphony 4 involving violin, clarinet, saxophone and bassoon, with fundamental frequencies in the range from 86 Hz to 750 Hz. A total of 546 musical notes are present, including harmonically-related notes. |

play more elaborate melodies. A number of tests recordings were obtained from excerpts of the Bach10 database [118], which involves four pitched instruments, namely, violin, clarinet, tenor saxophone and bassoon. Additionally, a percussive source consisting of a synthesised sequence of snare drums and cymbals is also considered. These recordings are arranged in four groups according to their polyphony levels and the pitches of the musical notes present, as described in Table 5.3. The original sources and mixtures used in this experiment are available online⁴.

The proposed iterative separation approach is applied to each audio mixture using the pitch tracking strategy, where the system automatically detects and selects the predominant note event in every iteration. The proposed semi-supervised systems are evaluated independently and the results are compared with a similar approach, known as the Interactive Sound Source Separation Editor (ISSE) [133], where the end-user is required to provide annotations in order to constrain, regularise, or otherwise inform the algorithm. These annotations are introduced by highlighting relevant sections of the input spectrogram, while the separation of the sources is obtained by an implementation of the NMF algorithm. In each case, Oracle separation results are included as well for comparison.

Figure 5.28 shows the separation performance of the systems for the test mixtures in Group 1, where IES-TFM and IES-TDS exhibit a slightly higher overall separation quality than ISSE, in terms of SDR. While the proposed algorithms deliver separated sources with significantly less interference among them, ISSE introduces less artifacts in the separated tracks. This behaviour might be related to the initialisation of ISSE, which depends entirely on the annotations provided and where harmonicity is not an assumption. Even when these annotations provide the

⁴<https://doi.org/10.5281/zenodo.3468471>

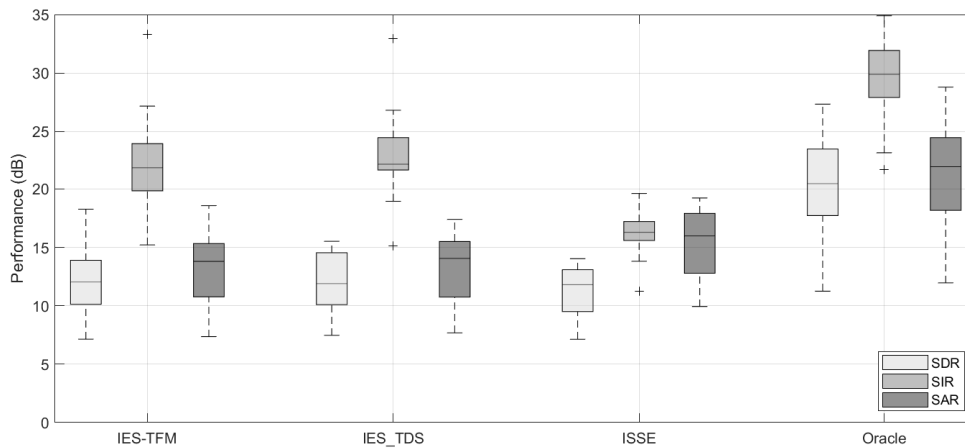


Figure 5.28: Separation performance on 12 test mixtures with polyphony 2 in Group 1 (two harmonic sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle).

localisation of the fundamental partial of every note, the algorithm struggles to identify some of their overtones and delivers in some cases an incomplete separation of the notes.

The incorporation of a third harmonic source within the test mixtures in Group 2 results in a reduction of the separation quality, as can be observed in Figure 5.29. A higher number of simultaneous sources means additional difficulties in providing good annotations for the sources, reducing the overall performance of ISSE, but it also means additional problems during the separation of overlapping harmonics, which affects the separation quality of the proposed systems. However, the higher number of note events in the mixture and the proximity of their frequency components are causing a greater reduction in the separation performance on ISSE, compared with Figure 5.28.

Harmonically-related notes, which are present in some of the mixtures in Group 2, introduce an additional challenge for both algorithms and affect their separation performance. The IES systems are able to detect the pitch trajectories of many harmonically-related notes, however, an accurate separation of the original note events is not possible, since the amplitudes of their harmonic partials cannot be correctly estimated from the mixed spectrogram. Similarly, the ISSE system also has problems interpreting overlaps between the provided annotations, associated with different sources, and therefore tends to allocate most of the shared energy to only one of the sources.

Results in Figure 5.30 correspond to the separation of the underlying sources from within the test mixtures in Group 3. Considering that the proposed methods are designed to detect

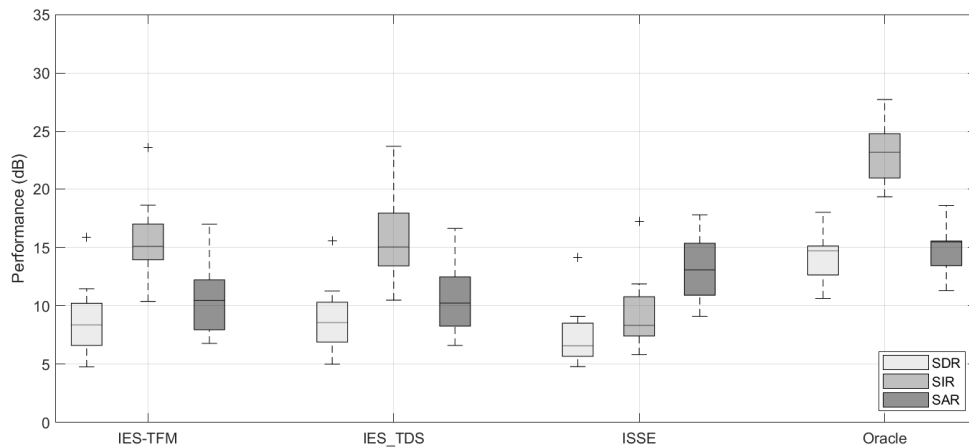


Figure 5.29: Separation performance on 12 test mixtures with polyphony 3 in Group 2 (three harmonic sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle).

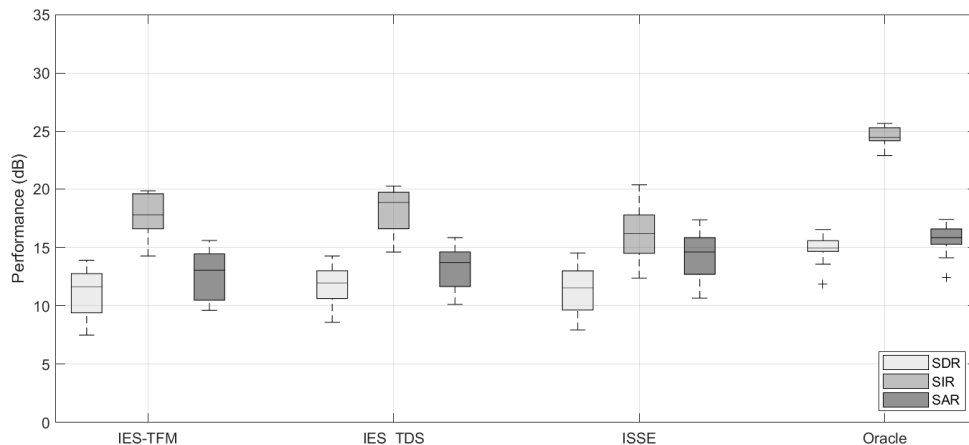


Figure 5.30: Separation performance on 12 test mixtures with polyphony 3 in Group 3 (two harmonic and one percussive sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle).

harmonic content, the percussive output is consequently contained in a residual signal together with other non-harmonic content. In the case of ISSE, the percussion is instead extracted first by exploiting additional user-provided annotations of solo percussive regions of the spectrogram. In this case, the algorithms show similar separation quality, with the IES variations still showing slightly less interference in the separated sources, while the ISSE approach introduces slightly less artifacts. In this specific experiment, the percussive source does not affect the detection of

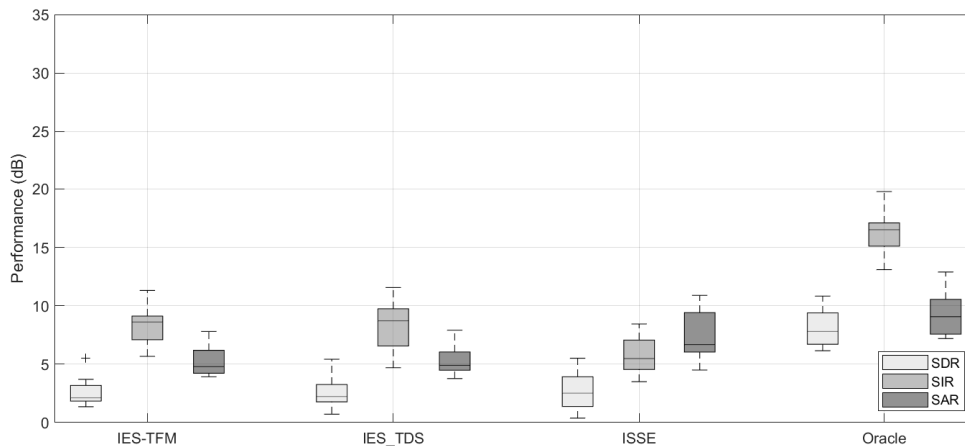


Figure 5.31: Separation performance on 12 test mixtures with polyphony 4 in Group 4 (four harmonic sources). Two variations of the proposed system (IES-TFM and IES-TDS) are compared with a similar separation process (ISSE) and with the Oracle estimates (Oracle).

note events within the IES automatic framework but, more generally, similar percussive effects might impact on the detection of musical notes with a fundamental frequency below 200 Hz.

Separation performances associated with the test mixtures in Group 4 are presented in Figure 5.31. In this case, four sources are playing simultaneously in every mixture and the range of fundamental frequencies has been expanded to include low-pitched notes (86 Hz to 200 Hz), which represent a significant challenge for each algorithm due to the very short spacing in between their harmonics and the increased likelihood of observing overlapping partials.

Figure 5.31 shows a significant reduction in separation quality delivered by each of the methods due to the increased polyphony of the mixtures. Although the ISSE still seems to generate less artifacts, the separated sources also exhibit higher levels of interference, suggesting that the annotations are not providing enough information to completely characterise each individual source. This problem is partially solved in the IES variations by assuming that the underlying sources are harmonic, which provides a simple but effective way to identify their frequency components based on the knowledge of their fundamental frequencies. The proposed dual-peak model provides a sharper separation of semi-overlapping harmonics, which also reduces interference among the separated sources. Problems associated with the separation of harmonically-related notes is another important factor that contributes to lowering the separation quality of all three methods investigated. When harmonically-related note events are detected by the proposed algorithms, their spectral content is arbitrarily partitioned using the strategy described in Section 5.5.3, which does not constitute a true separation of the real note

events. Similarly, the ISSE method also struggles to identify harmonically-related notes and the information provided by the annotations is not enough to identify the actual characteristics of the underlying musical notes and their separation is also incomplete.

In general, an important advantage of IES over ISSE is that it allows end-user interaction during the final stage of the process (clustering of note events), which seems to be more effective than using it at the beginning of the separation, as in the case of the ISSE process. From the user perspective, listening to separated events and grouping them into individual sources is far easier than recognising harmonic structures and estimating frequencies from within the spectrogram of a complicated audio mixture.

Finally, four representative separation examples are presented in Figures 5.32 to 5.35, for test mixtures in Groups 1, 2, 3 and 4, respectively. The original unmixed sources are presented in the time domain and compared with estimated sources obtained with the proposed system (IES-TFM and IES-TDS) and with the alternative process (ISSE).

5.8.6 Potential Sources of Error

Experiments conducted so far have been used to evaluate the overall performance of the proposed separation system and the influence of its principal stages. However, differences between the median performance of the methods presented here and the Oracle estimates still exist. Hence, a discussion of potential sources of error is presented in this section as a way to complete the analysis of the results. These potential sources of error can be summarised as follows.

- Incorrect pitch trajectories are a major source of separation errors, involving note events that are completely missed and pitch trajectories in which some of the estimates deviate from the real pitches present in the mixture. The separation stage is designed with a certain amount of flexibility to cope with small deviations in the pitch estimates, but when the estimated pitch is more than two frequency bins away from the real pitch, the system will not be able to deliver an effective separation of the spectral content associated with the estimated fundamental frequency.
- Interfering events misleadingly labelled as real harmonically-related notes prevent the complete extraction of the energy associated with the predominant note event, and reduce the quality of the separation. However, distinguishing real harmonically-related notes from false positives is difficult and usually depends on how stationary the signal is in any particular frame of the input spectrogram.

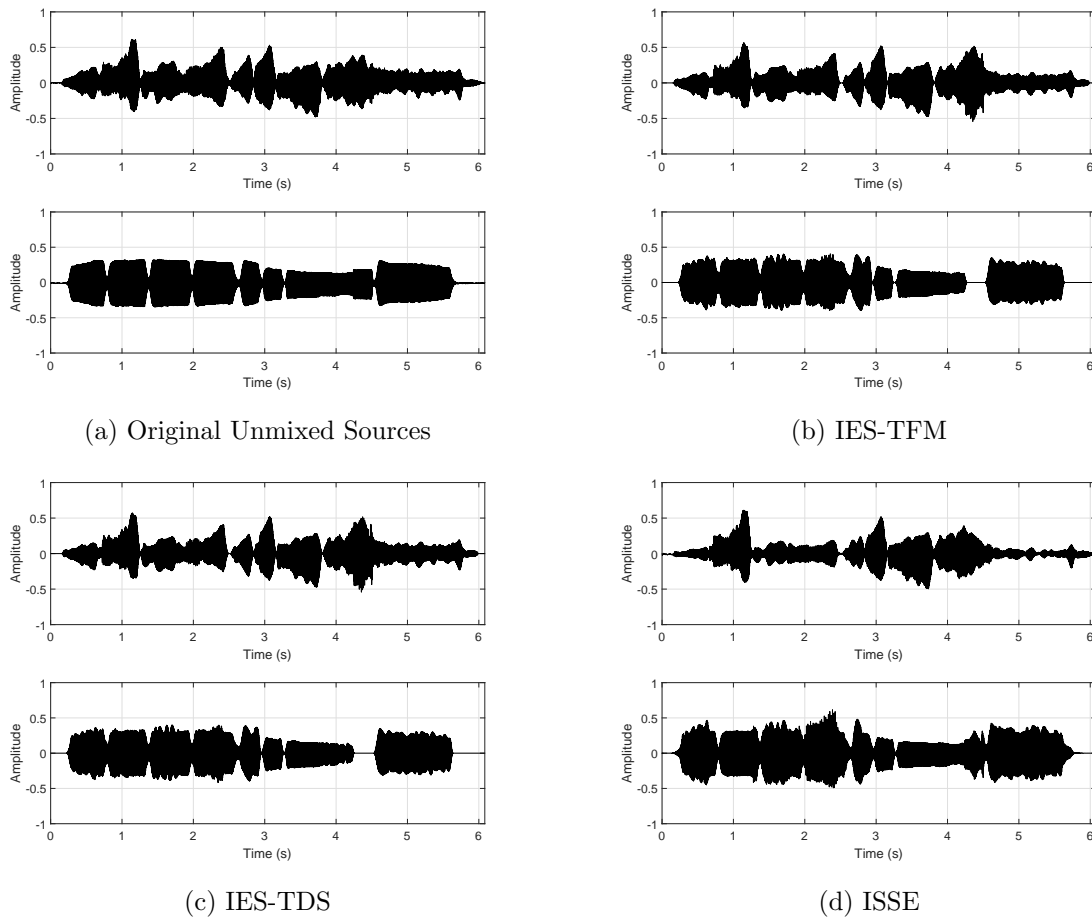


Figure 5.32: Original and separated sources for a representative test mixture from Group 1. (Top) Violin. (Bottom) Clarinet.

- High levels of distortion introduced by the strong phase interaction of very close frequency components tend to alter the shape of semi-overlapping partials, which complicates the identification and separation of the frequency component associated with the current predominant note event. In these cases, an algorithm based on optimisation might be the only way to extract an accurate set of parameters that will allow the reconstruction of the original components, provided that the optimisation approach has been properly initialised in a region close to the true solution.
- Fully-overlapping partials constitute a significant source of error in the proposed strategy, since their separation is currently being attempted by using an arbitrary partitioning of their spectral energy. This is another area in which an optimised separation approach might be useful to obtain the true number of overlapping partials and their parameters, allowing an exact partitioning of the energy and improving the separation quality.

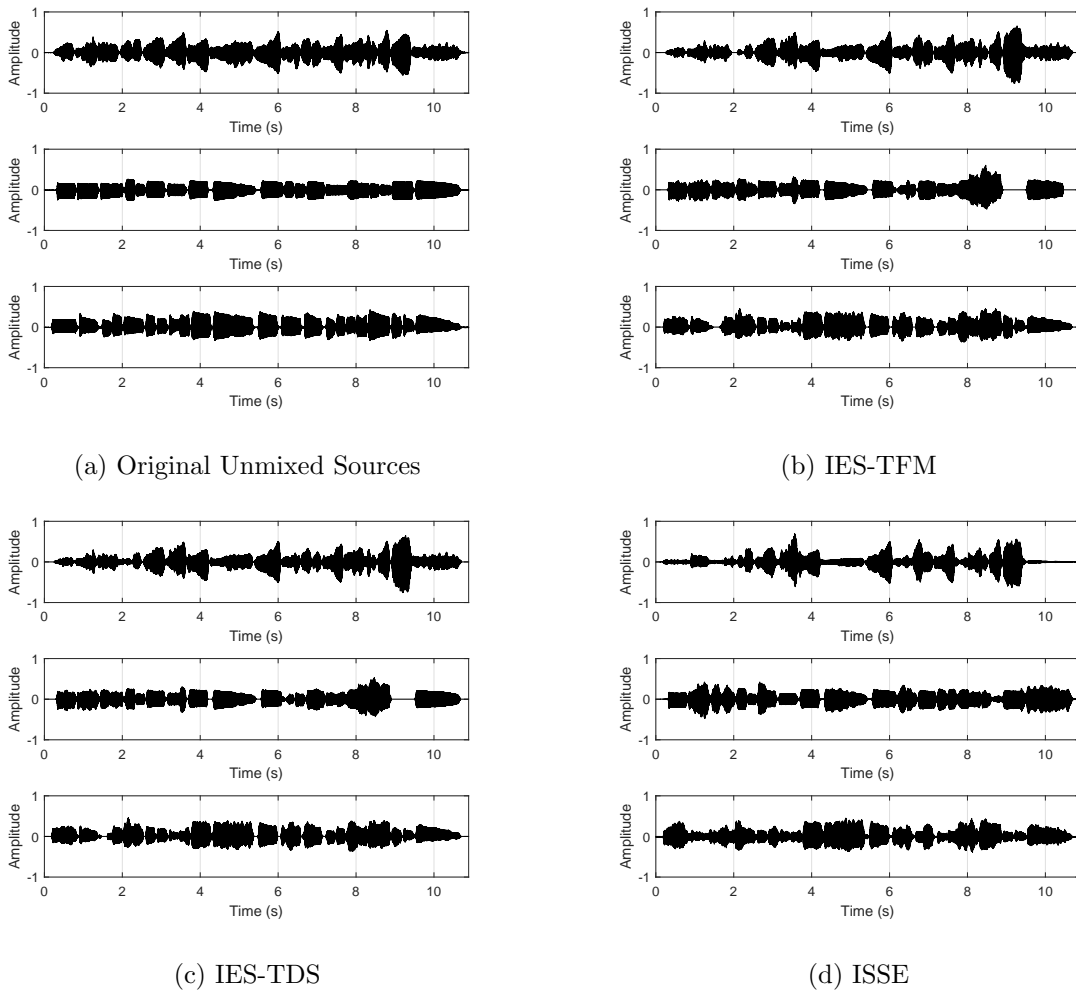


Figure 5.33: Original and separated sources for a representative test mixture from Group 2. (Top) Violin. (Middle) Clarinet. (Bottom) Tenor saxophone.

- Two or more sources playing the same note at the same time cannot be separated using the algorithm presented in this work. Usually, the system detects a single pitch trajectory associated with only one note event, which is the result of the colliding musical notes, but the extracted event cannot be effectively clustered with any of the separated sources since it contains non-separated spectral content. An appropriate separation of this type of event is highly difficult. However, an optimisation-based alternative might be able to recognise the presence of additional notes associated with the same pitch contour and could be used to find their relative intensities, allowing some degree of separation.
- Pitch trajectories comprising several consecutive notes coming from different sources are likely to occur, but they are not automatically separated by the algorithm and the corresponding events remain as a single unit. This prevents the end-user from correctly

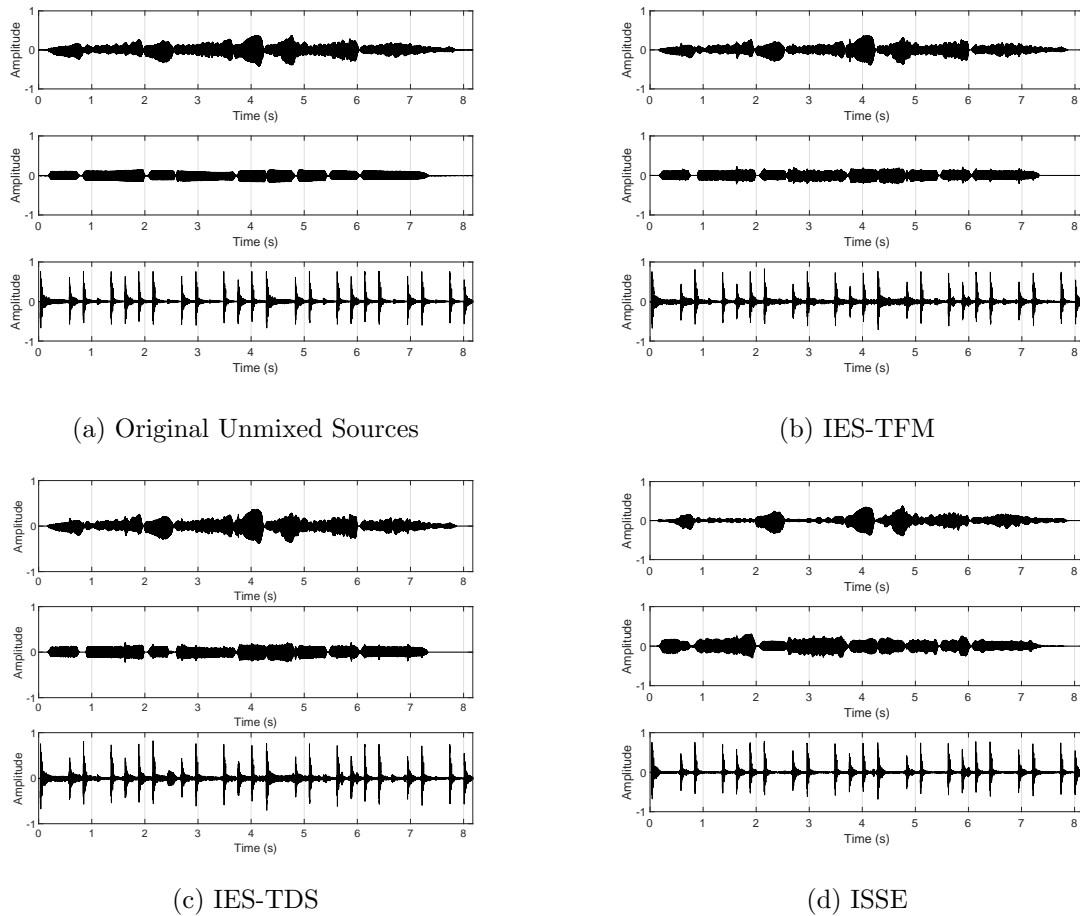


Figure 5.34: Original and separated sources for a representative test mixture from Group 3. (Top) Violin. (Middle) Clarinet. (Bottom) Percussion.

clustering the underlying musical notes with their corresponding sources, which increases the source-to-interference ratio and reduces the overall separation quality. This could be overcome by allowing the end-user to edit the separated note events as part of the clustering process.

5.9 Summary

This chapter has focused on separating the harmonic content of note events from within a polyphonic input mixture, using their pitch contours to identify the corresponding harmonic structures. The main issues have been in designing adaptive algorithms for tracking harmonic frequencies over time, and dealing with overlapping partials from multiple sources. Separation was performed on a frame-by-frame basis by using two different extraction methods, namely, time-frequency masking and time-domain subtraction. The separated note events were obtained

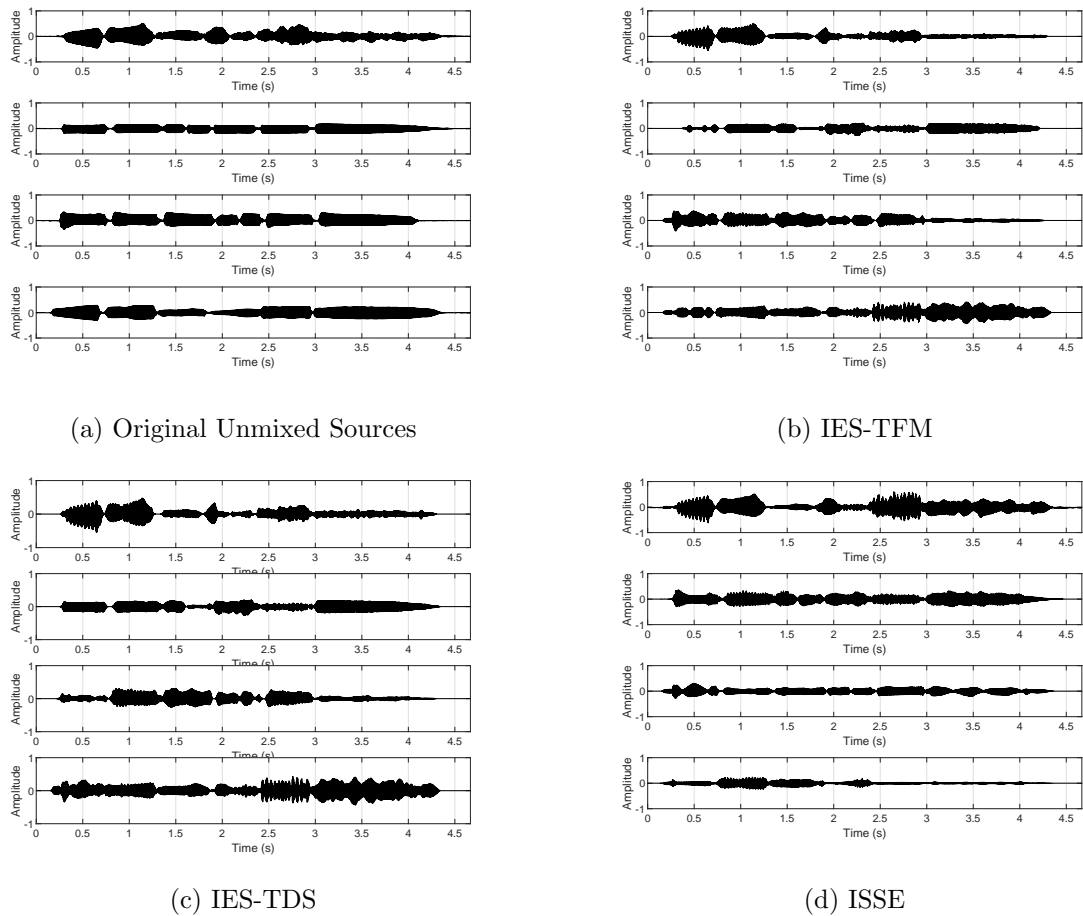


Figure 5.35: Original and separated sources for a representative test mixture from Group 4. (From Top to Bottom) Violin, Clarinet, Tenor Saxophone and Bassoon.

by inverse transforming of their extracted spectral content and using the overlap-add process to smooth out the time segments across frame boundaries. Separated sources were then constructed following a semi-supervised approach in which the end-user clustered all separated events into tracks, each of them consisting mostly of one individual source.

Given the pitch trajectory of each predominant note event from Chapter 4, its harmonics were tracked over time by finding a set of spectral peaks closest to the ideal harmonic frequencies in each frame. The peak-picking strategy was provided with some flexibility to tolerate small deviations in pitch estimates and to compensate for dynamic features of real music such as vibrato. Overlapping harmonics were handled in two different ways depending on the proximity of their underlying components. Semi-overlapping harmonics were separated by decomposing the shared peak into a number of components, from which the one closest to the ideal harmonic frequency was chosen as the one associated with the predominant note event, provided that a minimum spacing exists between their centre frequencies. Fully-overlapping harmonics, on the

other hand, were partitioned in a way that facilitated the detection of harmonically-related notes, where a set of preservation rates was used to control the amount of energy that was left in the residual and used to detect additional notes in later iterations. The aforementioned strategy was useful to increase the accuracy of the system in terms of multipitch estimation, while additional research is required to obtain an effective separation of fully-overlapping harmonics.

Two different methods were used to extract the spectral content of each note event from within the input mixture. Firstly, a non-binary time-frequency mask was constructed based on the shapes of the selected component and the observed spectral peak, using a dual-peak model in which the selected partial in the mixture is assumed to be the result of two components (dominant and secondary). The mask was applied to the input spectrogram to extract the spectral content of the note event, while a complementary mask was used to obtain the residual. Secondly, the separated spectral content of the note event was synthesised using sinusoidal modelling and reconstructed into a time-domain signal, which was then subtracted from the input mixture to obtain the residual. In both cases, phase information of the input mixture was used to inverse transform the spectral content associated with note events.

Evaluation of performance was conducted through a series of experiments where the proposed system was applied to a number of audio mixtures with different characteristics. First, six combinations of sustained violin notes were used to evaluate the performance of the algorithm on highly polyphonic mixtures. Second, the influence of the extraction order and other parameters of the system was assessed on a set of 12 mixtures with polyphony 3 consisting of musical notes played by different instruments. Finally, separation performance was also evaluated on 48 mixtures of real music with polyphonies 2 to 4, consisting of harmonic and percussive instruments. Results were compared with a previous method while Oracle estimates were also provided as a reference.

In general, the proposed system outperformed a previous MIDI-informed algorithm [28] in the separation of seven simultaneous violin notes, while the quality obtained by the system was comparable with the one produced by a residual-based system [23], specifically designed for this particular experiment. Although the order of extraction was found to have an impact on the separation performance, it was also found that this impact is not very large, while the extraction order based on salience represents an effective strategy to extract note events from within a mixture without the need for additional prior information. A pitch-dependent limit for the maximum number of harmonic partials extracted in every frame was also found to be beneficial, especially when low-pitched note events are present.

In terms of source separation of real music, the proposed algorithm showed comparable levels of separation quality to the ones obtained with a similar semi-supervised method, where the end-user provides annotations to guide the separation process, which is based on an implementation of the NMF algorithm. The higher separation quality obtained by the proposed system in polyphonies 2 and 3 was due to the lower levels of interference between the separated sources. In this respect, it was found that allowing end-user interaction during the clustering of note events was more effective than using it at the beginning of the process, where the complexity of the mixture can make it very difficult to recognise harmonic structures.

Chapter 6

Mono-to-Stereo Upmixing

6.1 Preamble

Spatial hearing is a crucial feature of the human auditory system that allows the segregation of multiple sound sources in complex acoustic environments. The localisation of a sound is encoded as binaural disparities in the form of Interaural Level Difference (ILD) and Interaural Time Difference (ITD), which are better known as binaural or spatial cues [134].

In mono recordings, such localisation information is not available, and hence converting them into stereo becomes a difficult and challenging task. The lack of spatial cues obviously restricts the listening experience, and hence many approaches have been presented which attempt to artificially create spatial cues from mono recordings. However, despite the fact that the resulting version may be perceived as having elements of a true stereo signal, it is not, in the sense that, in general, it is impossible to place individual sources into different parts of the stereo image.

The potential of audio source separation techniques, on the other hand, has significantly increased in recent years, allowing the extraction of individual sources from polyphonic music, preserving most of the original quality of the sound while introducing less distortion.

The aim of this chapter is to propose a method where stereo mixes are created based on the semi-supervised audio source separation strategy presented in the previous chapter, which allows the estimation of individual sources that can then be panned by the end-user into different parts of the stereo image. The quality of the new stereo mixes is then evaluated by means of

a listening test, in which a number of participants were asked to assess both the effectiveness and naturalness of ten upmixed recordings, generated from separated sources obtained with the proposed framework and with another separation approach.

6.2 Previous Approaches

Initial attempts to introduce a spatial impression in mono recordings, usually referred to as pseudostereophonic processes, focused on creating a signal pair (or as many signals as the number of output channels desired) that evokes some specific auditory spatial image in the listener, by means of taking copies of the original signal and then delaying or filtering these new channels in different ways [135].

However, these methods suffered from several limitations, in particular, the fact that the stereo effect was arbitrary and hence, it could not be controlled or associated with the different instruments. As a result, the placement of the individual instruments (or sources) at different points in the stereo field was not possible [10].

The use of parametric coding methods, widely applied to perceptual audio coding, were also explored in previous studies. The principal disadvantage, though, was that such systems often required additional information to be provided [136].

Machine learning techniques have been used to automatically divide soundtracks of movies into music and voice segments, which were spatialised differently depending on the type of segment. But this approach proved to be unsuitable for music, where instruments and voices are usually playing simultaneously [137].

Other methods based on source separation rapidly appeared, thanks to the increasing advances in detection accuracy and separation quality. In 2007, Lagrange et al. proposed an upmixing method based on source formation, defined as the automatic detection of time-frequency clusters, which were grouped into larger formations corresponding to sound sources. The user was then allowed to select the panning for each of the sources before creating the final stereo mix. Degradation of the audio quality was reported when the separated sources were hard panned to the left or to the right, due to the presence of separation artifacts [137].

Taking advantage of previously developed strategies to perform separation between harmonic and percussive sounds [138], FitzGerald presented a combined upmixing method that allowed the positioning of the separated sources in the stereo field, improving the naturalness of the resulting sound [10]. He also explored vocal extraction in [139] and expanded his upmixing

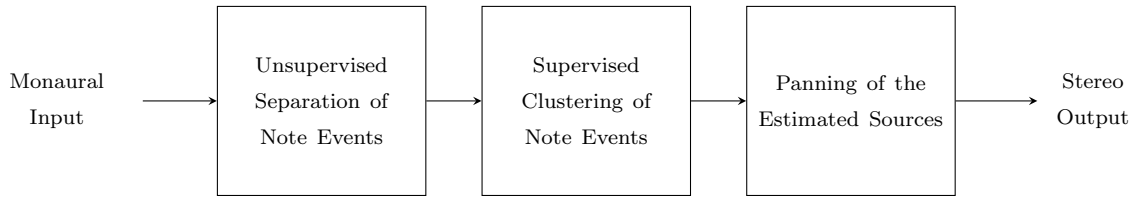


Figure 6.1: Simplified block diagram of the proposed mono-to-stereo conversion system.

process, which was later used to create stereo mixes from several commercial recordings of the Beach Boys, previously available in mono format only [140].

In more recent years, Uhle and Gampp presented a real-time process in which the mono signal was decomposed into foreground and background signals [136]. The background component was decorrelated by using a set of allpass filters, in order to generate stereophonic information. The final stereo version was produced by mixing the decorrelated background signal with the foreground section panned in the middle. Results from a listening test showed that the stereo versions were rated higher than the original mono signals. However, as with early pseudostereo methods, it is still impossible to place a single source at a chosen position within the stereo mix.

6.3 Proposed Method

According to FitzGerald [10], when source separation techniques are used in mono-to-stereo conversion, there are two considerations that have to be observed during the process: the audible artifacts and the stability of the pan position of the separated sources. This implies that high quality separation processes should produce better stereo mixes.

With this in mind, the upmixing process presented here exploits the semi-supervised source separation approach described in Chapter 5, to obtain individual sources that can be panned to different parts of the stereo image in order to create a wider spatial experience in the final version. The aforementioned separation strategy is based on the iterative detection and extraction of note events, which are considered to be harmonic sounds consisting of either one single musical note or several consecutive notes with similar pitches, usually coming from the same source, and characterised by a continuous pitch trajectory. When the iterative stage is complete, note events are clustered by the end-user to form individual sources. Non-harmonic signal content appears in a separated residual channel that can be manipulated further if required. The diagram presented in Figure 6.1 summarises the principal stages of the proposed solution.

Since the study presented in this chapter was conducted before developing the time-domain subtraction as an additional extraction method for note events, the results presented here are

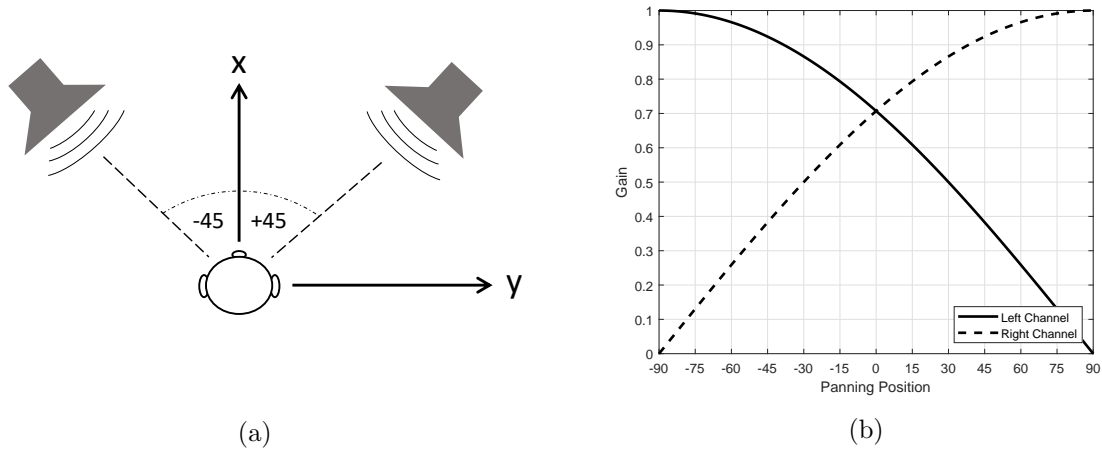


Figure 6.2: Panning of the sources. (a) Diagram of the stereo setup showing the positioning of two sources at different angles. (b) Left and right gains following the constant power law.

based on separated sources consisting of note events that were extracted from within the original mixture by means of time-frequency masking, as described in Section 5.6.1.

6.3.1 Source Panning

Given the estimated sources, the panning process now depends entirely on the user's choice. For instance, the sources can be loaded into a Digital Audio Workstation (DAW) and manipulated as a normal multi-track session, in which different panning and effects can be applied to each track individually. In this work, the constant power panning law is used to generate the stereo output, where the gains for the left and right channels, denoted as $L(\theta)$ and $R(\theta)$, respectively, are computed as follows.

$$L(\theta) = \cos\left(\frac{\pi(\theta + 90)}{360}\right) \quad (6.1)$$

$$R(\theta) = \sin\left(\frac{\pi(\theta + 90)}{360}\right) \quad (6.2)$$

where $\theta \in [-90 + 90]$ is the panning position, expressed as an angle in degrees, and selected in accordance with the two-speaker stereo setup shown in Figure 6.2(a). The corresponding gains $L(\theta)$ and $R(\theta)$ are presented in Figure 6.2(b).

The residual can also be used in the final stereo version to improve its naturalness and to smooth out possible separation errors. For this work, there is no attempt to further process the residual and hence the best option is to leave this track in the centre of the stereo image.

Table 6.1: Overview of the listening test items.

| Item | Description |
|----------|---|
| Mono | Original mixture in mono format used as an anchor. |
| Stereo 1 | Stereo mix using separated sources generated by the alternative ISSE source separation process. |
| Stereo 2 | Stereo mix using separated sources generated by the proposed method. |
| Stereo 3 | Stereo mix using separated sources generated by the proposed method plus the final residual panned in the centre of the stereo image. |

6.4 Evaluation of Performance

6.4.1 Database

In order to assess the proposed system, six excerpts from the Bach10 database [118] have been selected and used for evaluation. Tracks corresponding to different instruments in each excerpt are used to generate ten test mixtures with polyphony 2. The average duration of each mixture is around 6 seconds, and the complete set is 67.45 seconds long. Approximately 178 musical notes are present, with fundamental frequencies spanning from 86 Hz (F2) to 750 Hz (F#5). All mixtures are created in mono format and sampled at 44.1 kHz. These test recordings and the original unmixed sources are available online¹.

6.4.2 Listening Test

The sound quality and naturalness of the upmixed stereo recordings are evaluated by means of a listening test, which consists of ten questions. Approval for this listening test was obtained from the Ethics Committee, within the Department of Electronic Engineering, under the reference code *Castro060918*, and it was then applied from September 12th to September 17th, 2018.

In each of the questions, participants are asked to rate the quality and naturalness of the spatial sound of four different audio mixes, according to Table 6.1, using a scale from 0 (poor quality) to 100 (high quality). The first item corresponds to the original mono mixture, used here as a hidden anchor. The second item is a stereo mix created by means of an alternative semi-automatic source separation process, known as Interactive Sound Source Separation Editor (ISSE), proposed by Bryan and Mysore [133], where the underlying sources are separated using an NMF-based approach and annotations provided by the end-user. Items three and four are

¹<https://doi.org/10.5281/zenodo.3477406>

stereo mixes created from separated sources obtained with the proposed separation strategy. The only difference between items three and four is that the last also incorporates the final residual (obtained after extracting all detected note events) panned in the middle of the stereo field.

Stereo items in Table 6.1 are produced by panning the first estimated source at 70 degrees left from the centre of the stereo image, while the second one was positioned at 70 degrees to the right. When the final residual is included, it is always panned in the centre. This arrangement has been deliberately selected in order to produce a very wide stereo image, in which separation errors are likely to produce instability of the panning position of the sources, in order to stress the limitations of the approach.

No reference has been included in any of the questions given that the true stereo versions of the test recordings are not available. Participants are expected to judge the quality and naturalness of the upmixed stereo recordings based on their professional experience, which provides them with a general idea of how a stereo track should sound. Hence, only individuals with significant training in audio and music technologies were chosen to undertake the test.

The listening test was implemented using the Qualtrics Survey Software², based on a three-page framework. The first page provided participants with an introduction to the listening test, including the purpose of the study and general instructions. Each participant had to give consent before continuing to the second page, in which a practice trial was presented. Then, the third page contained the ten trials in the test. In each trial, participants were allowed to listen to all four items in any order before expressing their opinion by moving a slider below each of them. Figure 6.3 shows the general layout of each question in Qualtrics. Randomisation was used to change the order in which the trials and their items were presented to each participant.

Nine participants were then recruited (one audio producer, an associate researcher in audio processing, and seven postgraduate students in acoustics, spatial sound and music technology). The test was conducted under controlled listening conditions, using a reference quality audio interface (FireFace UC) and headphones (Beyerdynamic DT 770 Pro).

6.4.3 Results and Discussion

Results of the listening test are presented in Figure 6.4. In Figure 6.4(a), the median ratings for both variations of the proposed system (57.5 and 56) are the highest, followed by the ISSE (36), whilst the Mono mixtures are the lowest (1.5). To determine whether the median treatment

²<https://www.qualtrics.com/uk/>

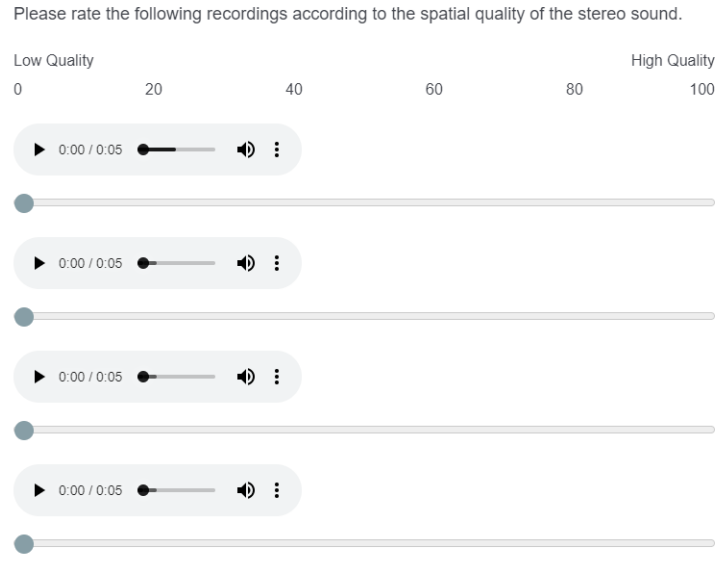


Figure 6.3: Sample trial of the listening test on Qualtrics.

effect differs for the upmixing process, a Friedman test on the data shows $p = 3.415 \times 10^{-31}$ and $\chi^2 = 144.84$. From a Chi-squared table, it can be obtained that the critical value for a significance level $\alpha = 0.05$ and 3 degrees of freedom is 7.81. Then, considering that $\chi^2 \gg 7.81$ and $p \ll 0.05$, it can be concluded that the null hypothesis can be rejected and hence, there is a difference among the four upmixing processes.

In order to determine whether the methods evaluated are significantly different, a multiple comparison procedure (pairwise test) is performed on the ratings provided by all participants in the listening test, using the same significance level $\alpha = 0.05$. Results of this analysis are summarised in Table 6.2 and indicate that significant differences exist between mono, ISSE and the proposed methods. Figure 6.4(b) shows the mean ratings for all four upmixing methods and their corresponding 95% confidence intervals. These reinforce the aforementioned findings. After this statistical analysis, it can be concluded that the proposed method is more effective than the original mono and the ISSE process.

Despite the fact that no significant difference has been observed between the pure version of the system and the system plus the residual, the small variation of their medians could be explained by the nature of the residual signal, which embraces most of the non-harmonic content in the mixture and sometimes corresponds to important acoustic cues, such as note

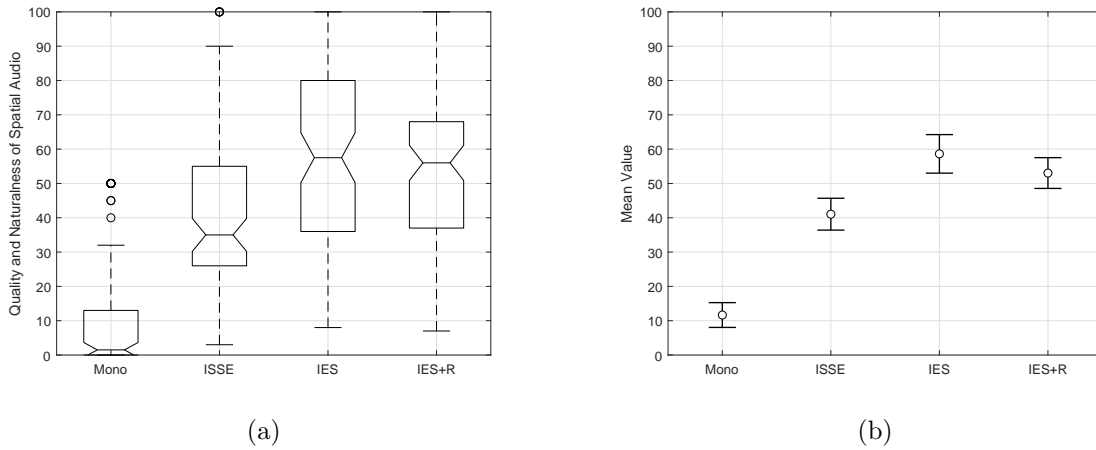


Figure 6.4: Results of the listening test. (a) Original data from all participants. (b) Mean values per item and 95% confidence intervals. (Mono) Original monaural mixture, (ISSE) Interactive Sound Source Separation Editor, (IES) Proposed system, and (IES+R) Proposed system with final residual panned in the middle.

Table 6.2: Pairwise comparison. Methods: (1) Mono, (2) ISSE, (3) IES, and (4) IES+R.

| Groups | Mean Difference | 95% Confidence Interval | | p-value |
|--------|-----------------|-------------------------|-------------|---------|
| | | Lower Bound | Upper Bound | |
| 1 & 2 | -29.39 | -37.99 | -20.78 | 0.00 |
| 1 & 3 | -46.97 | -55.57 | -38.36 | 0.00 |
| 1 & 4 | -41.38 | -49.98 | -32.77 | 0.00 |
| 2 & 3 | -17.58 | -26.18 | -8.97 | 0.00 |
| 2 & 4 | -11.99 | -20.59 | -3.38 | 0.00 |
| 3 & 4 | 5.59 | -3.02 | 14.19 | 0.34 |

attacks. Bringing back the residual into the stereo mix is an important factor in ensuring that the resulting mix sounds natural, but in terms of the spatial quality, the acoustic cues in the residual were no longer located in the same position of the estimated sources, resulting in some type of contradiction or confusion for some of the listeners. This effect was anticipated and emphasises the need for further post-processing of the residual.

Comparing the proposed system with the alternative one (ISSE), the difference in performance might be explained by considering the way in which user interaction is used. In our case, the end-user is called to cluster already separated note events, whilst the ISSE requires the user to provide annotations to guide the separation process before it starts, by means of painting on the spectrogram of the audio mixture. This scheme has two significant limitations. Firstly, it cannot cope with some special cases, for example, when the fundamental harmonic of one note overlaps with another note. Secondly, depending on the complexity of the input mixture, providing accurate annotations can be very difficult, especially when notes are short or close to

each other. It is believed that clustering meaningful note events into sources is simpler for the end-user than recognising structures inside a complicated audio mixture, but additional user tests are required to fully justify this statement.

6.5 Summary

In this chapter, a novel mono-to-stereo conversion process was presented, in which an iterative note-event based separation process is used to isolate the underlying components of the mono mixture, which are then clustered into sources by the end-user, before positioning them in different locations within the stereo field.

These note events can be seen as audio objects, so that different audio effects can be applied to them before the new stereo upmix is generated. Additional applications of the system include lead and accompaniment separation, as well as audio quality enhancement in old audio recordings and film soundtracks.

Evaluation of performance was carried out by means of a listening test, in which stereo mixes were created using separated sources obtained with the proposed method, and also with an alternative process based on an implementation of the NMF algorithm. A wide stereo image was considered so that separation errors would be easier to identify.

Two variations of the proposed strategy achieved better ratings than the unprocessed mono mixture and the alternative process. Among these two variations, stereo mixes generated from the pure separated sources received slightly higher rates than the ones with the final residual panned in the middle.

Chapter 7

Conclusions and Further Work

Audio source separation from single-channel recordings is a challenging field that continues to attract significant interest within the research community and the industrial sector. This thesis has addressed the problem by exploiting an iterative estimation-separation framework that allows the system to deliver the separated audio sources together with content-related information, such as their pitch trajectories. Chapters 1 to 3 have served as an introduction to audio signals and single-channel audio source separation techniques. Chapter 4 has described a novel approach to multipitch analysis based on the iterative estimation and extraction of note events, while Chapter 5 has presented a semi-supervised source separation framework based on the user-assisted clustering of note events. An application of the proposed framework to the conversion of mono recordings into stereo has been addressed in Chapter 6.

7.1 Final Remarks

The proposed solution combines multipitch detection with source separation in order to deliver a semi-supervised framework that consists of two main stages. Firstly, the input mixture is automatically decomposed into a number of note events, which are assumed to be harmonic sounds, characterised by a continuous pitch trajectory and associated with a number of musical notes, played by one particular source. Secondly, all detected note events are then clustered by the end-user to form individual sources. Any percussive content can also be recovered afterwards by subtracting the estimated harmonic sources from the original input mixture.

Note events are detected and extracted from within the input mixture following an iterative approach. In every iteration, the pitch trajectory of the predominant note event is selected from an initial set of fundamental frequency estimates, and then it is used to guide the separation of the spectral content associated with the predominant note event, on a frame-by-frame basis.

Two different methods have been proposed to extract the predominant note event from within the mixture in every iteration, based on time-frequency masking and time-domain subtraction. In both cases, a novel strategy is used to deal with overlapping partials, which allows the separation of closely located harmonics in those circumstances where only the predominant pitch is known. A different technique has been used to handle totally-overlapping harmonics, which focuses on facilitating the detection of other simultaneous harmonically-related notes, while the optimal separation of their components has not been attempted in this work. A proximity criterion has been presented as a way to distinguish totally-overlapping harmonics from other shared partials.

7.1.1 Note Event-based Multipitch Estimation

Decomposing an audio recording into note events has proven to be effective for the estimation of multiple fundamental frequencies in complex music with different levels of polyphony. Since the estimation of the underlying pitch trajectories is carried out in small sections, the accuracy of the estimates has increased while the number of outliers has been significantly reduced. Moreover, the continuous pitch trajectories characterising these note events should also facilitate the computation of additional content-related information, such as onset and offset times.

Since the proposed algorithm detects and then extracts the predominant note event in every iteration, heavily masked musical notes have also been detected once the louder ones are removed, representing an advantage over joint estimators. However, the correct detection of these additional events depends in many cases on whether overlapping partials can be appropriately identified and separated, which at the same time can be affected by the levels of phase interaction between other nearby frequency components and their relative volumes.

Within the proposed note event-based decomposition scheme, it has been observed that the automatic selection of the order in which note events are extracted, and the correct identification of simultaneous harmonically-related notes, are the more vulnerable aspects of the system in which the dynamic nature of music plays a significant role. Choosing the right order of extraction and detecting all harmonically-related notes is not always possible, but the framework presented

in this thesis has been conceived as an alternative that takes advantage of the information available in the audio mixture and requires no previous training on similar audio tracks.

Experiments conducted in Chapter 4 have shown high levels of accuracy in multipitch detection for audio mixtures consisting of musical notes with relatively high pitches, even for high levels of polyphony. The presence of low-pitched notes does not usually represent a significant problem if the polyphony of the mixture is not higher than two, otherwise the accuracy of the estimation can deteriorate due to the reduced space in between harmonic partials and the increased probability of observing overlapping harmonics consisting of more than two components.

In terms of processing times, obtaining the final set of pitch trajectories for the underlying sources takes significantly longer than the joint algorithm in [118], considering that each iteration requires the calculation of a completely new set of initial fundamental frequency estimates. This issue has been partially addressed by keeping the hop size at 50% during the estimation of the predominant pitch contour, which means that the initial pitch estimates are computed based on a spectrogram with fewer frames. Detecting multiple events in a single iteration might be another way to reduce processing times, as discussed in Section 7.2.

7.1.2 Semi-Supervised Audio Source Separation

The proposed separation framework is based on an automatic decomposition of the input recording into note events, followed by a clustering stage in which end-user interaction is used. This has proven to be effective for a variety of audio mixtures containing elaborate melodies in which harmonic or nearly-harmonic instruments are involved. From the end-user point of view, it is believed that grouping separated note events to form individual sources is easier and more effective than recognising different harmonic structures within the spectrograms of highly complicated audio mixtures.

To identify the harmonic partials of a given fundamental frequency, a peak-picking strategy has been proposed, which tolerates small deviations in frequency due to inharmonicity of the instrument or musical effects such as vibrato. However, the accurate detection of harmonic partials in musical notes played by some string instruments, such as piano, harp and guitar, has proven to be a difficult task, due to their rapidly decaying spectral envelopes and the influence of inharmonicity, as discussed in Section 5.4.1. Additional constraints have to be introduced within the peak-picking algorithm to improve the way in which the next harmonic frequency is iteratively predicted in every frame, during the extraction of the predominant note event, in

order to incorporate additional information into the process, such as approximate estimations of the inharmonicity coefficient or a description of the expected spectral envelope.

Results from experiments have displayed very good separation performance for audio mixtures with polyphonies 2 and 3, in which the level of interference is usually low and the artifacts introduced by the process do not affect the overall audio quality of the separated sources, especially when they are recombined to produce new multi-channel audio mixtures with different spatial properties.

Slightly better separation results have been obtained by using time-domain subtraction as a way to extract note events from the input mixture, providing important evidence of the benefits associated with this softer extraction process, which is particularly useful when there is not enough information to characterise all the underlying components of the mixture.

Significant emphasis has been put on the separation of harmonic or nearly-harmonic sources from within polyphonic input recordings. However, this does not represent a restriction on the type of audio sources that can be processed by the algorithm, and mixtures containing harmonic instruments and percussion have also been studied. Due to the nature of the proposed solution, percussive events cannot be directly extracted from within the input mixture; rather, they tend to appear in the final residual signal that is obtained after the extraction of all harmonic sources.

However, the presence of percussive sounds may complicate the detection of pitch trajectories, in particular, for low-pitched notes whose partials are heavily distorted by the low-frequency energy associated with percussive sources. Other extracted note events may also contain audible levels of interference caused by the incorrect extraction of additional energy at high frequencies associated with percussive events.

7.2 Further Work

Possible future directions of the work presented in this thesis are outlined in this section. What follows is a list of a few specific areas that might be fruitful to investigate further in order to provide improvements to the current implementation of the source separation system.

7.2.1 Extraction of Multiple Note Events per Iteration

Each note event that is extracted from within the input mixture increments the levels of noise in the residual, which is then used as an input to the next iteration. For this reason, the overall quality of the initial set of pitch estimates in every iteration is expected to drop as

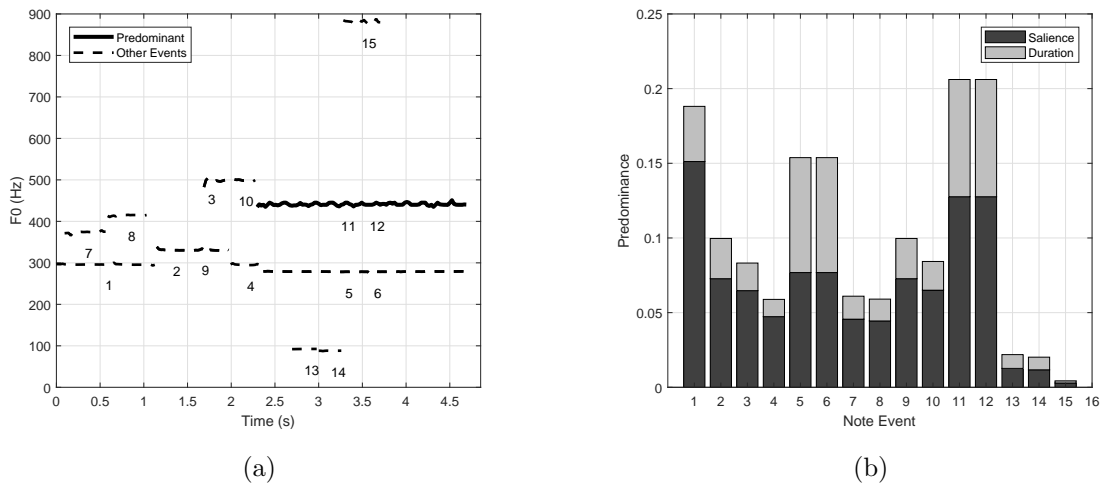


Figure 7.1: Audio mixture consisting of violin and clarinet. (a) Note events detected during the first iteration. (b) Predominance of each detected note event.

the number of iterations increases, reducing the chances of detecting and separating additional musical notes.

In some cases, the definition of the predominant note event could be modified to encompass several non-simultaneous note events with similar levels of predominance, so they can be individually extracted from within the current input mixture in a single iteration. By using a larger number of reliable pitch estimates in early iterations, the audio quality of the extracted note events should increase, while the computation time should be reduced by a large margin, provided that less iterations are now necessary to achieve a similar decomposition of the input mixture. The clustering stage should not be affected since each note event would continue to be handled individually.

Considering an input mixture consisting of two melodies being played by a violin and a clarinet, all detected note events in the first iteration of the system are presented in Figure 7.1(a), while their corresponding predominances are shown in Figure 7.1(b). The system has chosen the violin note A4 (events 11 and 12) as the predominant one during the first iteration. While events 1, 2, 5, 6, and 9 are also showing high levels of predominance, they are discarded by the algorithm and only the note A4 is extracted. The proposed modification would take advantage of these additional events to extract multiple notes in the same iteration, for instance, the clarinet notes D4 (event 1) and E4 (events 2 and 9), which are not simultaneous notes and have high levels of predominance.

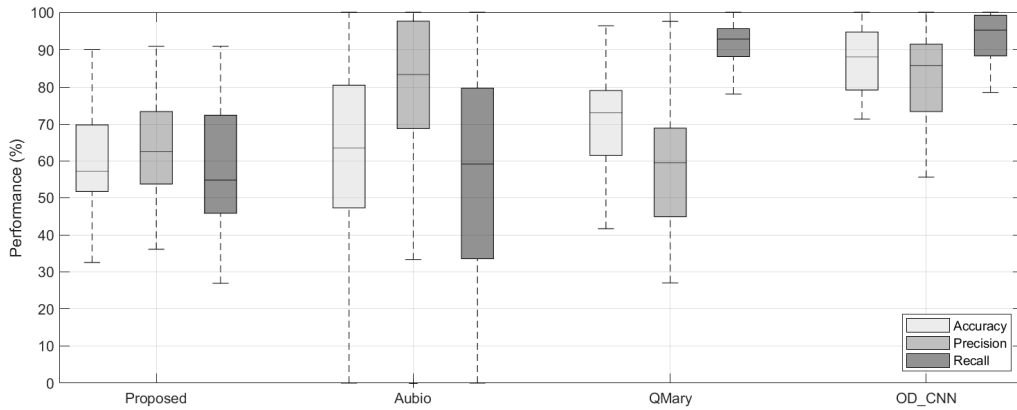


Figure 7.2: F-Measures obtained using a preliminary onset detector based on the proposed separation framework and three other onset detectors [141].

7.2.2 Onset Detection

A preliminary version of the proposed iterative separation system was applied to onset detection in [141], where an Onset Detection Function (ODF) was obtained from the final residual signal after the extraction of the harmonic content of musical notes. This algorithm was tested on a dataset consisting of 23 monophonic recordings, taken from the database used by Holzapfel et al. in [142], and results were compared with three other methods, namely, the Aubio onset detector (AUBIO), Queen Mary’s onset detector plug-in (QMARY), and the onset detector proposed in [143] based on Convolutional Neural Networks (CNN) (OD-CNN). These results are reproduced in Figure 7.2.

It is believed that the current version of the system should be able to perform better than the previous one in onset detection, not only for monophonic input signals, but also in polyphonic music. Since the input recording is now decomposed into note events, the computation of an ODF is not necessary, but each note event has to be processed further in order to identify note events associated with one of the following conditions.

- It contains a single musical note.
- It contains several musical notes and has to be subdivided.
- It contains a section of a larger musical note and has to be merged with other events.

Once the estimated note events have been rearranged according to the categories above, onsets and offsets could be directly obtained from their pitch trajectories. Moreover, some

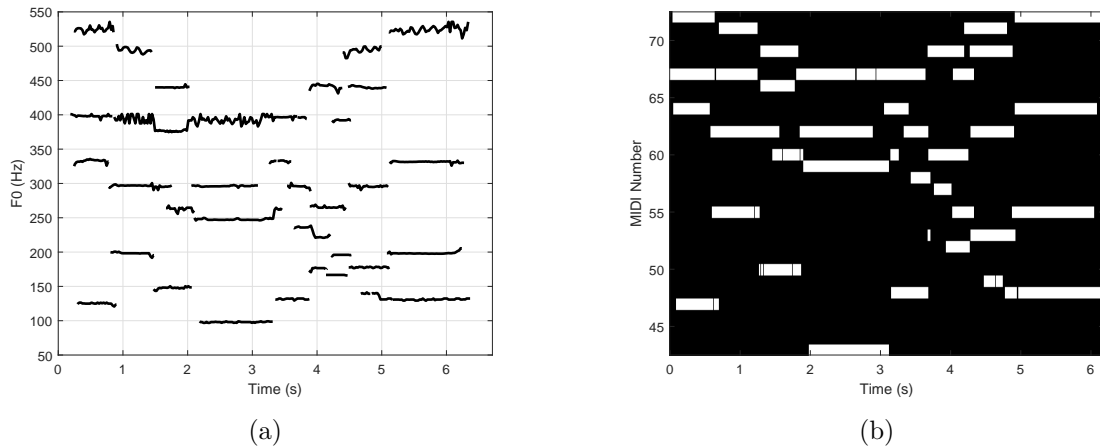


Figure 7.3: An example of WAV-to-MIDI conversion using an audio excerpt from the Bach10 database as an input recording. (a) Estimated pitch contours. (b) Equivalent MIDI piano roll.

adjustments could also be introduced if an important note attack is found in the final residual signal close to any of the estimated onsets, as well as being useful for tempo estimation.

7.2.3 WAV-to-MIDI Conversion

Another interesting area in which the proposed framework could be applied is WAV-to-MIDI conversion. Considering that the system already delivers pitch trajectories for the extracted note events, they can be easily averaged out across their duration and converted into a MIDI number. Then, the information of all note events could be embedded into a MIDI file and used in any DAW system, allowing the music to be played by other instruments or modified in any possible way. Figure 7.3 shows an example of such a conversion, using an excerpt from the Bach10 database, with polyphony 4, as an input signal. Pitch contours in Figure 7.3(a) are rearranged into individual notes and then converted into a MIDI file by means of the MATLAB[®] functions developed by Ken Schutte¹. The final piano roll is presented in Figure 7.3(b).

7.2.4 Automatic Clustering of Note Events

Considering the proposed source separation framework, presented in Chapter 5, comparing its levels of performance with other recent methods proved to be difficult, mainly because of its semi-supervised nature, in which end-user interaction is required to complete the clustering of note events into sources. However, the user-assisted grouping stage could be replaced by

¹<http://kenschutte.com/midi>

another strategy in which instrument identification methods are used, such as the ones presented in [144–149], in order to obtain a fully-automated system.

Instrument identification is a challenging task on its own, and many existing algorithms have been designed to select and extract a number of features from the input signal, in order to identify or categorise the underlying instruments. Some of them are specifically for monophonic music, while others are suitable for polyphonic audio mixtures. Integrating these methods with the proposed framework should benefit from two main aspects. First, as the proposed system is able to deliver a reliable estimation of the polyphony of the system, the number of possibilities within the search process is significantly reduced. Second, the feature extraction can be performed directly on the separated note events, which is equivalent to identifying one single instrument at a time.

The performance of the resulting fully-unsupervised source separation process could then be compared with many of the recently developed machine learning-based algorithms, which are now being trained on larger sets of data. The upgraded system could also be used as a supporting resource during the labelling of many of these example tracks within training datasets, reducing development times and the risk of human error.

7.2.5 Optimisation and Beyond

The separation of overlapping partials was discussed in Section 5.5, where two different methods were proposed to handle semi-overlapping and fully-overlapping harmonics. Shared partials within the first category are decomposed into a principal and secondary components, using information from the multipitch detector and the magnitude spectrogram, and then, the one closest to the ideal harmonic frequency is chosen as the separated partial. On the other hand, the energy contained in a fully-overlapping harmonic is not separated according to the contribution of each concurrent source, it is only arbitrarily partitioned in order to facilitate the detection of harmonically-related notes.

In this section, a new strategy for the separation of overlapping partials is presented, where magnitude and phase information are exploited within an optimisation process. Results obtained so far already show its potential as a tool for spectral analysis.

Within this framework, the STFT of the input signal is computed using a short frame size (typically 1024 or 2048 samples), with 87.5% overlap, a Hanning window function, and a zero-padding rate of $4/1$. In every frame of the input spectrogram, denoted as $\mathbf{X}(f, m)$, a set of spectral peaks with significant magnitudes is selected as relevant observed peaks, each of which

is assumed to be the result of a multi-component time-domain signal. Considering the v -th time frame of the input spectrogram, the main lobe of the i -th relevant observed peak is defined as follows.

$$\mathcal{B}_i = \mathbf{X}(f_{c,i} - \delta : f_{c,i} + \delta, m_v) \quad (7.1)$$

where $f_{c,i}$ is the approximated centre frequency of the i -th relevant peak and δ is a parameter of the system that can be adaptively defined based on the relative shape of the spectral peak. The time-domain signal $y_i(t)$, responsible for the i -th relevant peak, is assumed to be the combination of an infinite number of sinusoidal components as expressed in Equation 7.2.

$$y_i(t) = \sum_{k=1}^{\infty} s_k(t) = \sum_{k=1}^{\infty} A_k \cos(2\pi f_k t + \phi_k) \quad (7.2)$$

Assuming that most of the energy in the main lobe of the i -th relevant peak is the contribution of a finite number of sinusoidal components, an approximation to $y_i(t)$ is defined as follows.

$$\tilde{y}_i(t) = \sum_{k=1}^K s_k(t) = \sum_{k=1}^K A_k \cos(2\pi f_k t + \phi_k) \quad (7.3)$$

where the exact number of sinusoidal components and their parameters (amplitudes, centre frequencies and phase angles) are unknown. If we take the Fourier transform of the approximated signal $\tilde{y}_i(t)$, that is $\tilde{\mathbf{Y}}_i(f) = \mathcal{F}\{\tilde{y}_i(t)w(t)\}$, using the same frame size, window function $w(t)$ and zero-padding rate as the ones used to generate the original spectrogram, then an approximated main lobe can also be extracted as follows.

$$\tilde{\mathcal{B}}_i = \tilde{\mathbf{Y}}_i(f_{c,i} - \delta : f_{c,i} + \delta) \quad (7.4)$$

A cost function is then defined by adding the Mean-Squared Error (MSE) between the absolute values of the observed and approximated main lobes, their real parts and their imaginary parts, as defined by the following equation.

$$\begin{aligned} C_i(\tilde{\mathbf{A}}_i, \tilde{\mathbf{f}}_i, \tilde{\mathbf{\Phi}}_i) = & \text{E} \left[(|\tilde{\mathcal{B}}_i| - |\mathcal{B}_i|)^2 \right] + \text{E} \left[(\Re\{\tilde{\mathcal{B}}_i\} - \Re\{\mathcal{B}_i\})^2 \right] \\ & + \text{E} \left[(\Im\{\tilde{\mathcal{B}}_i\} - \Im\{\mathcal{B}_i\})^2 \right] \end{aligned} \quad (7.5)$$

where parameter vectors $\tilde{\mathbf{A}}_i$, $\tilde{\mathbf{f}}_i$, and $\tilde{\mathbf{\Phi}}_i$ are the estimated amplitudes, centre frequencies and phase angles of the K sinusoidal components used to approximate the original signal $y_i(t)$. The optimal number of sinusoidal components K , and their corresponding parameter vectors, can

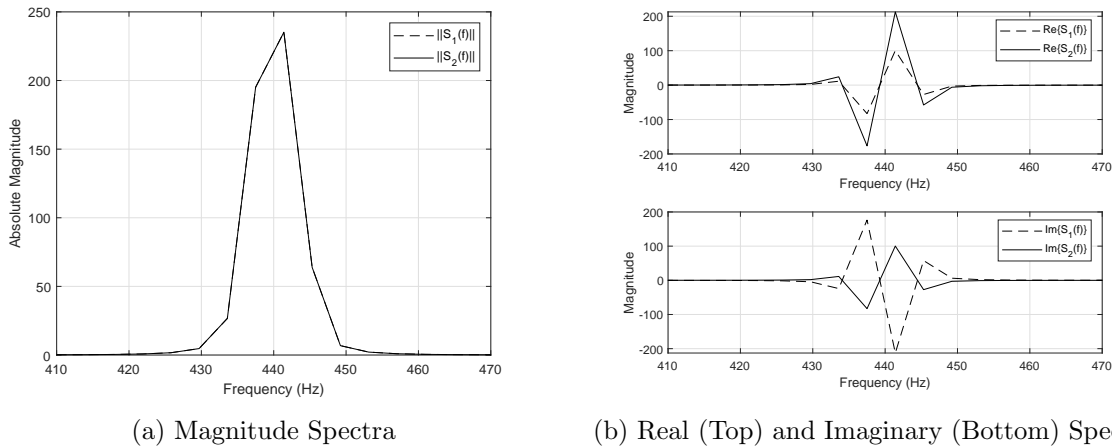


Figure 7.4: Magnitude, real and imaginary spectra for two pure sinusoids having the same amplitudes and frequencies, but different phase angles.

be obtained by minimising the cost function in Equation 7.5. In this work, an unconstrained non-linear programming solver is used for these purposes.

Considering the spectra in Figure 7.4, which correspond to the signals $s_1(t) = A_1 \cos(2\pi f_1 t + \phi_1)$ and $s_2(t) = A_2 \cos(2\pi f_2 t + \phi_2)$, where $A_1 = A_2 = 0.5$, $f_1 = f_2 = 440$ Hz, $\phi_1 = 0^\circ$, and $\phi_2 = 90^\circ$, it is clear that the phase difference cannot be detected from their magnitude spectra, shown in Figure 7.4(a), while their real and imaginary spectra present different shapes for each of the two signals, as shown in Figure 7.4(b). For this reason, the real and imaginary spectra have been included in Equation 7.5 as a way to bring phase information back into the estimation process and allow the algorithm to find phase angles for the underlying sinusoidal components, which is a task that cannot be accomplished by solely using the magnitude spectrum.

A different cost function is constructed and minimised for each relevant peak in every frame of the spectrogram. The resulting parameter vectors are arranged into a data structure that can be used to reconstruct each individual component, or eventually clusters of components, in order to separate a number of sources from within the input mixture.

The proposed algorithm has already shown a significant potential in the separation of overlapping partials, even when the number of underlying components is higher than two. The following examples are presented to illustrate the separation of overlapping partials using the proposed optimisation-based decomposition framework. The cost function is minimised by using a quasi-Newton solving algorithm (within the `fminunc` function in MATLAB[®]), in which the initial point for all parameter vectors is defined by first decomposing the observed peak using Parsons' method [81], and then obtaining raw parameters for each component. The maximum

number of function evaluations has been set to 3000, while the frame size used is 1024, at a sampling frequency of 44.1 kHz. In every case, a 4096-point Fourier transform is always applied.

First, the single-frame sinusoidal components in Figure 7.5 are considered. The true parameters of these signals are: $\mathbf{A} = [0.22 \ 0.36 \ 0.45]$, $\mathbf{f} = [440 \text{ Hz} \ 470 \text{ Hz} \ 500 \text{ Hz}]$, and $\Phi = [30^\circ \ 60^\circ \ 90^\circ]$. The time and frequency domain representations of the resulting mixture are presented in Figure 7.6, where the strong phase interaction between the overlapping components produces the amplitude variations shown in Figure 7.6(a), and the shared peaks presented in Figure 7.6(b).

When the optimisation-based decomposition algorithm is applied to the separation of the overlapping partial in Figure 7.6(b), three components were automatically obtained with the following parameters: $\tilde{\mathbf{A}} = [0.44 \ 0.23 \ 0.38]$, $\tilde{\mathbf{f}} = [500.2 \text{ Hz} \ 440.4 \text{ Hz} \ 469.3 \text{ Hz}]$, and $\tilde{\Phi} = [89.1^\circ \ 28.1^\circ \ 63.7^\circ]$. The approximated overlapping partial and the estimated components are presented in Figure 7.7.

If the analysis of this mixture were based solely on magnitude information, only the first and third components would have been located, with approximately the right centre frequencies, but with the wrong amplitudes and phase angles, which in the end would have led to an incorrect separation of the signals. The proposed method, on the other hand, delivers a set of highly accurate parameters, associated with the underlying components, and allows a precise reconstruction and separation of the signals involved.

A second example is presented in Figure 7.8, in which another mixture is generated by combining two sinusoids with the following parameters: $\mathbf{A} = [0.42 \ 0.62]$, $\mathbf{f} = [440 \text{ Hz} \ 445 \text{ Hz}]$, and $\Phi = [0^\circ \ 60^\circ]$. When the proposed strategy is applied to the mixture in Figure 7.9, the following estimated parameters are obtained: $\tilde{\mathbf{A}} = [0.63 \ 0.41]$, $\tilde{\mathbf{f}} = [444.99 \text{ Hz} \ 439.98 \text{ Hz}]$, and $\tilde{\Phi} = [59.9^\circ \ -0.1^\circ]$. Figure 7.10 shows the reconstructed components. Results in this example show that even two overlapping partials with only 5 Hz between their centre frequencies can be correctly separated by means of the proposed optimisation-based approach.

Finally, two examples are presented to show how the proposed optimisation-based approach can be used to improve the frequency resolution of a short-frame spectrogram, without changing the time resolution. In the first example, a synthetic input signal is constructed by combining several sinusoidal components with different centre frequencies and occurring at different times. Some of these components are as short as 0.02 s, while the frequency distance between some other events can be as small as just 20 Hz. An ideal time-frequency representation of this input signal is presented in Figure 7.11.

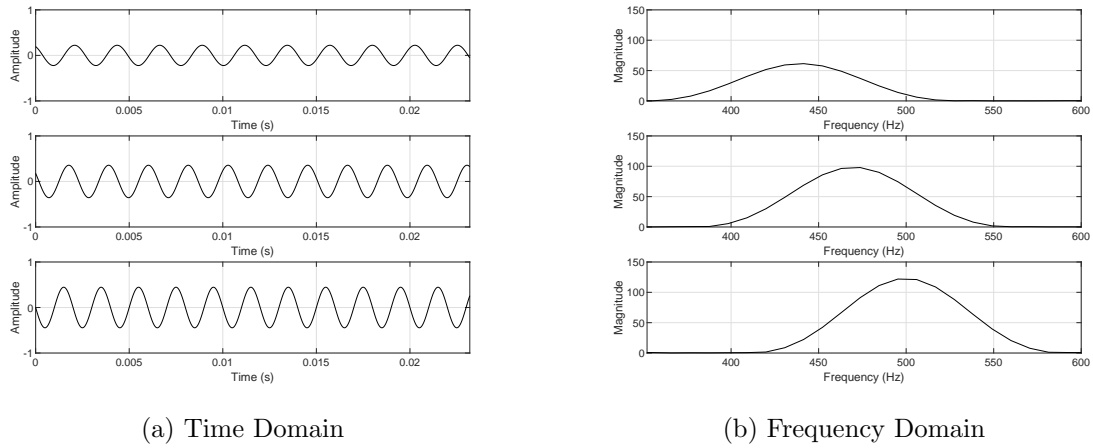


Figure 7.5: Three different sinusoidal components with centre frequencies separated 30 Hz from each other. (Top) $s_1(t) \rightarrow S_1(f)$, (Middle) $s_2(t) \rightarrow S_2(f)$, and (Bottom) $s_3(t) \rightarrow S_3(f)$.

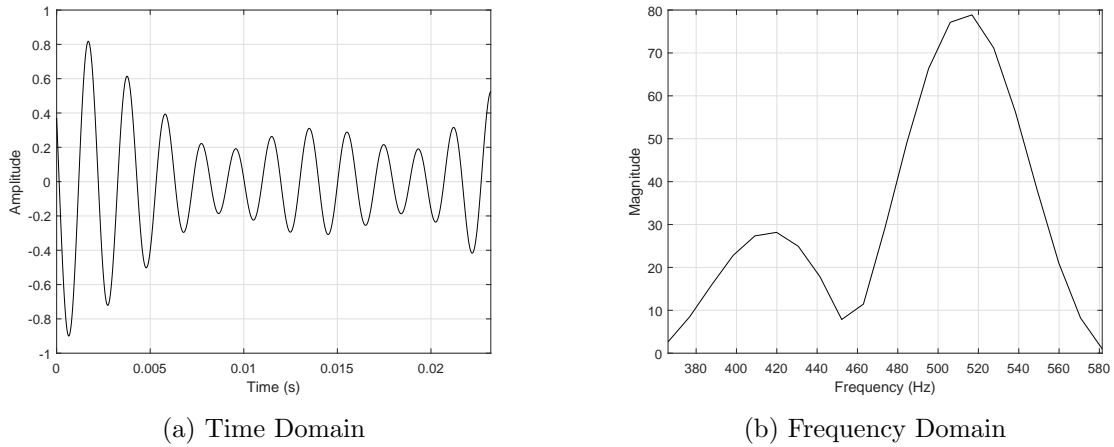


Figure 7.6: Input mixture generated by mixing the sinusoidal components in Figure 7.5.

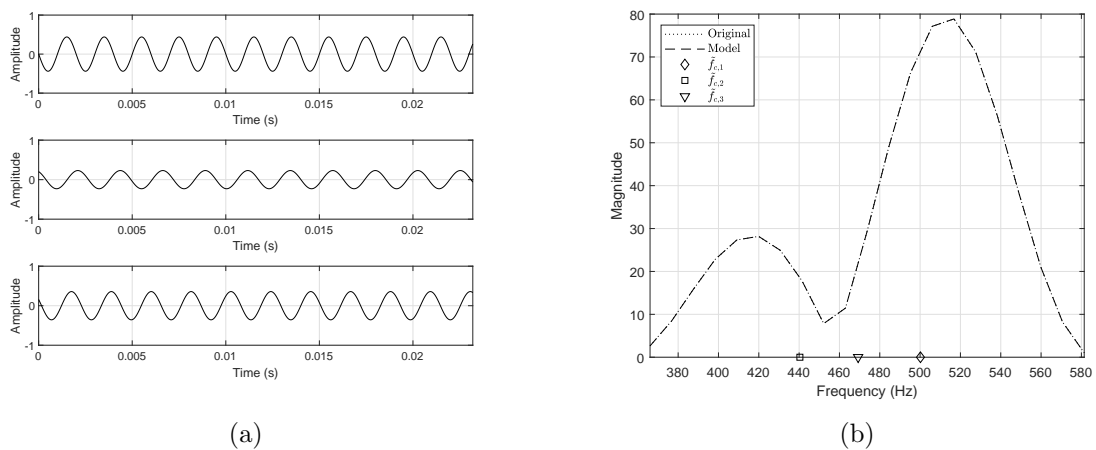


Figure 7.7: Results of the optimisation-based estimation process. (a) Identified components in the time domain. (b) Comparison between the observed and approximated overlapping partials.

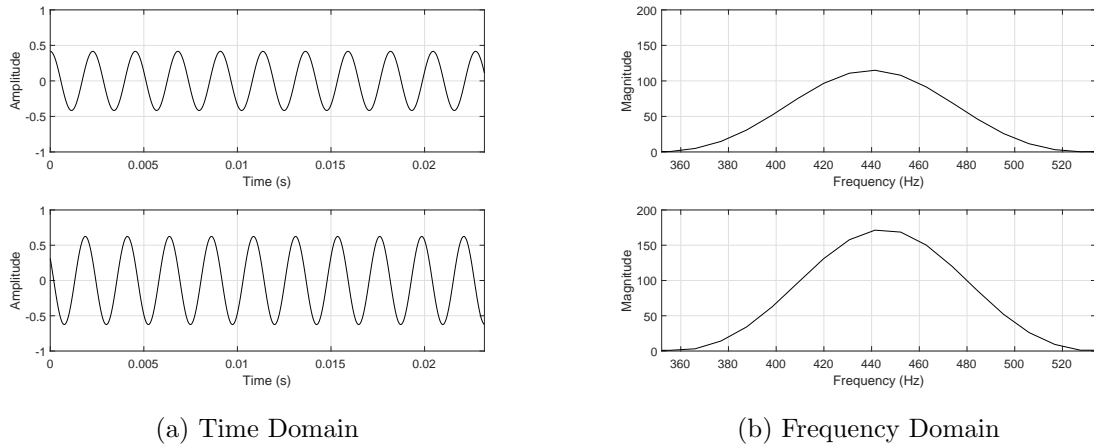


Figure 7.8: Two sinusoidal components with centre frequencies separated only 5 Hz from each other. (Top) $s_1(t) \rightarrow S_1(f)$, and (Bottom) $s_2(t) \rightarrow S_2(f)$.

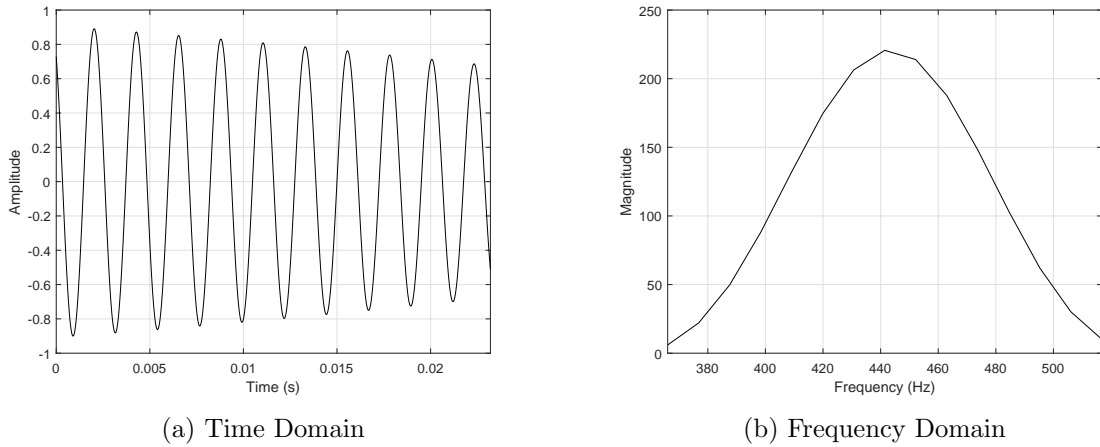


Figure 7.9: Input mixture generated by mixing the sinusoidal components in Figure 7.8.

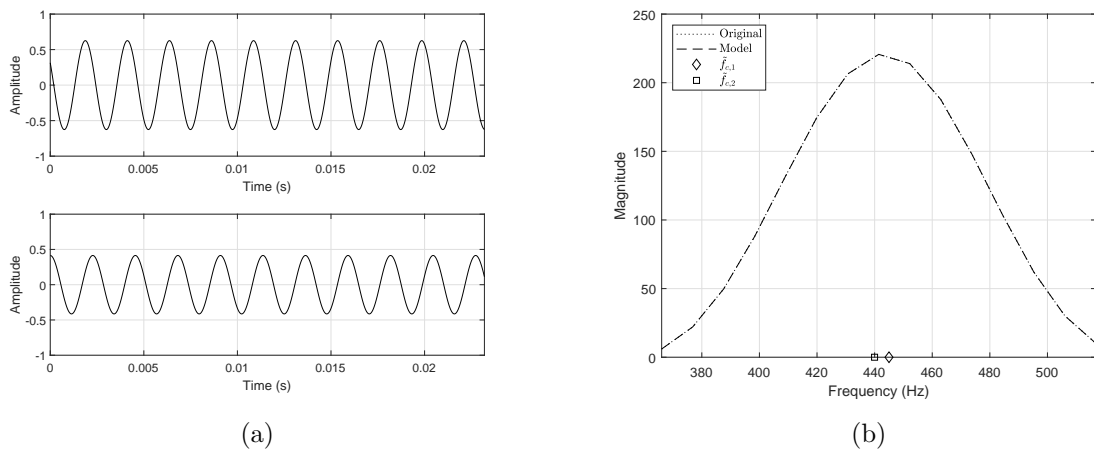


Figure 7.10: Results of the optimisation-based estimation process. (a) Identified components in the time domain. (b) Comparison between the observed and approximated overlapping partials.

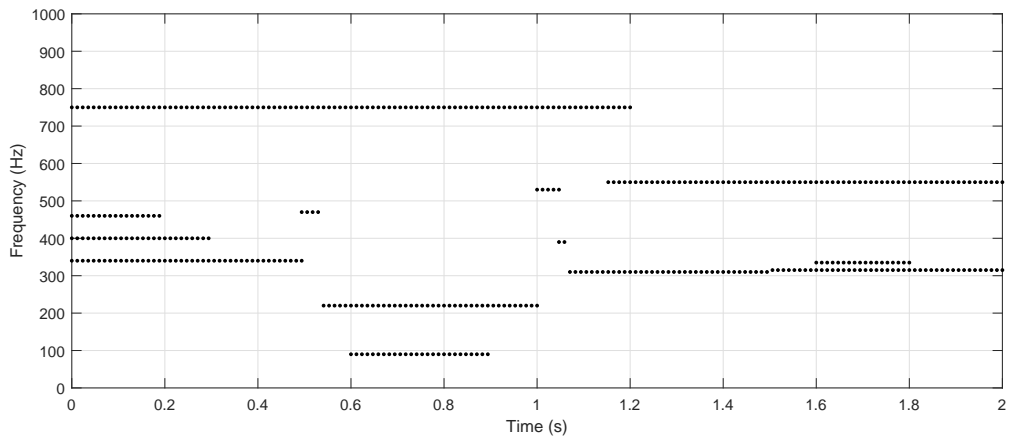


Figure 7.11: Ideal time-frequency representation of a synthetic input signal consisting of several sinusoidal components with different centre frequencies and occurring at different times.

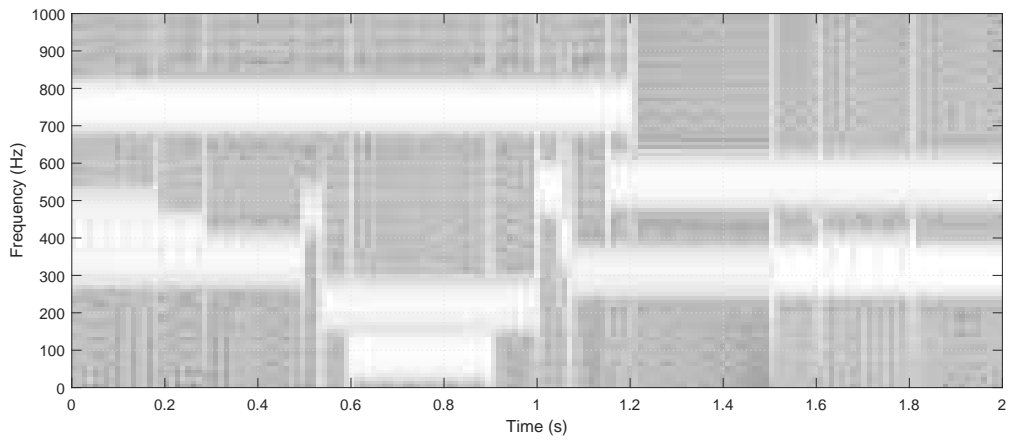


Figure 7.12: Spectrogram of the input signal characterised in Figure 7.11 using a frame size of 1024 samples ($f_s = 44.1$ kHz).

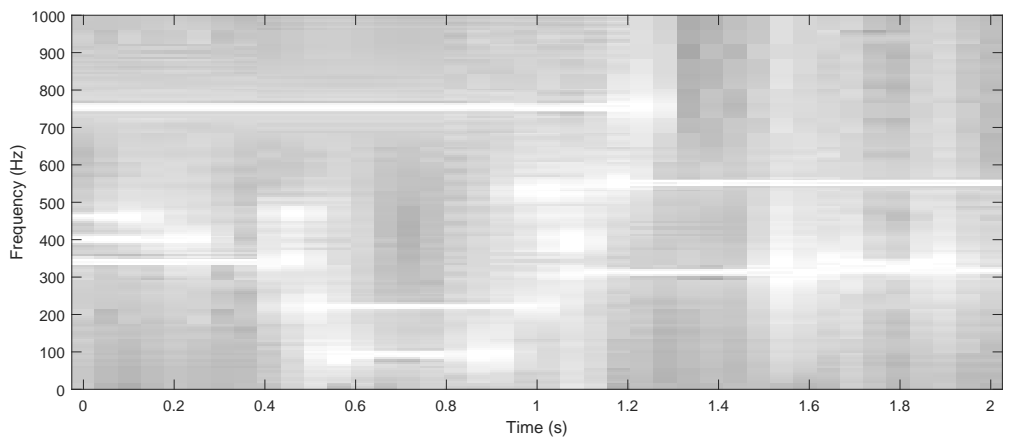


Figure 7.13: Spectrogram of the input signal characterised in Figure 7.11 using a frame size of 8192 samples ($f_s = 44.1$ kHz).

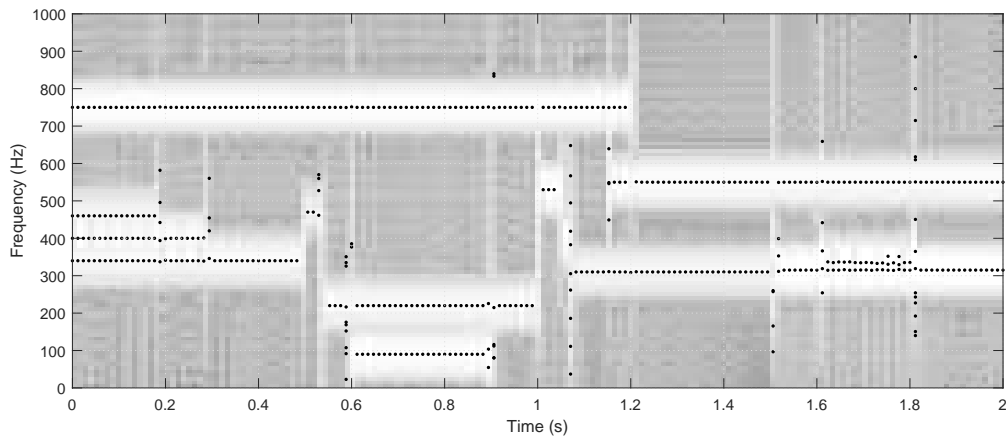


Figure 7.14: Optimisation-based process used to improve the frequency resolution of the spectrogram in Figure 7.12. Dots are used to mark the centre frequencies of the estimated components found in every frame.

Figure 7.12 shows a spectrogram of the input signal generated with a frame size of 1024, while another spectrogram is shown in Figure 7.13, where a frame size of 8192 is used. In both cases, the overlap between frames is 75% and $f_s = 44.1$ kHz. The short frame in Figure 7.12 delivers a high time resolution which allows a better localisation of each component in the time axis. However, the low frequency resolution makes it difficult to localise each component in the frequency axis, since the spectral information appears significantly blurred. Figure 7.13, on the other hand, shows a better frequency resolution due to the increased frame size, but its reduced time resolution makes it difficult to localise each component in the time axis.

When the proposed optimisation-based approach is applied to the short-frame input spectrogram in Figure 7.12, it identifies a number of relevant peaks in every frame, and minimises their corresponding cost functions, obtaining a set of parameters for their underlying components. In Figure 7.14, the centre frequencies of the estimated components are marked with dots right on top of the short-frame input spectrogram, but it has to be noticed that the algorithm also delivers their amplitudes and phase angles.

Several interesting details can be observed in Figure 7.14, firstly, the improved frequency resolution of the optimised representation allows the identification of the two sinusoidal components with very close centre frequencies in the interval from $t = 1.6$ s to $t = 1.8$ s. The distance between their centre frequencies is just 20 Hz (less than a frequency bin), and cannot be easily identified in the original spectrogram. Also, at $t = 1.5$ s, the frequency of the lower sinusoidal component changes from 310 Hz to 315 Hz, a change that can only be noticed on the new representation. This improved resolution in frequency does not change the localisation of

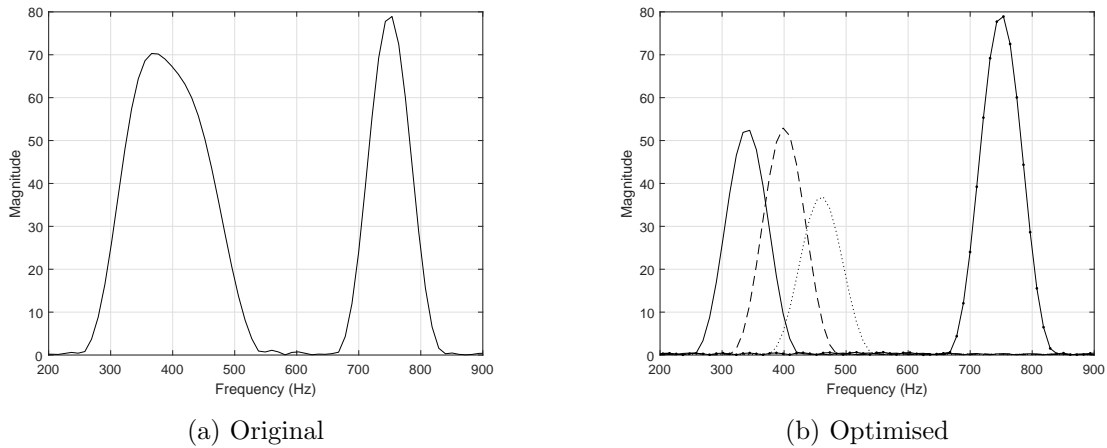


Figure 7.15: Magnitude spectra of the original mixture and the estimated components for a single frame of the spectrogram in Figure 7.12 at $t = 0.1$ s. The solid, dashed and dotted lines in (b) show that the original shared peak in (a), centred at 400 Hz, has been resolved as a combination of 3 overlapping components using the optimisation-based approach.

events in time, as evidenced at $t = 0.5$ s and at $t = 1.0$ s, where the two very short events are still detected by the algorithm. In some other frames, especially near transitions, the algorithm does not deliver the right number of components for some of the peaks and their parameters are incorrect, since the model cannot handle the non-stationary nature of these particular frames.

A cross-section of the spectrogram in Figure 7.12 is shown in Figure 7.15(a), considering the time frame at $t = 0.1$ s, and it is then compared with the magnitude spectra of the estimated components detected within the same frame. Figure 7.15(b) shows how the complex overlapping partial centred at 400 Hz is correctly decomposed into its three overlapping components, whose centre frequencies are 60 Hz from each other. This separation would not be possible by only using magnitude information.

A second example is presented in Figure 7.16. It shows the results of applying the optimisation-based representation to a mixture of real musical notes played by two instruments: viola and clarinet. Pitch contours of these notes were previously presented in Figure 5.19(a). Thanks to the improved resolution of the new representation, the centre frequencies of each harmonic partial are clearly localised, while overlapping harmonics are correctly resolved in most of the frames, as observed in Figure 7.16(b), from $t = 1.0$ s to $t = 1.7$ s. A cross-section of this spectrogram, at $t = 1.5$ s, is shown in Figure 7.17. From these graphs, it can be observed that two overlapping partials (at 880 Hz and 1800 Hz) are being separated into their corresponding components, which should allow an effective separation of the original sources.

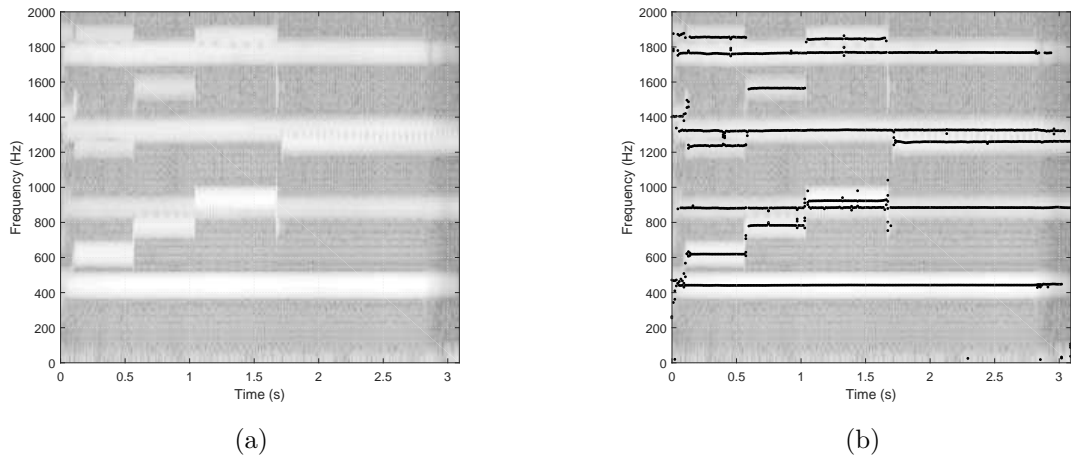


Figure 7.16: Optimisation-based process used to improve the frequency resolution of a short-frame spectrogram containing several real musical notes being played by a viola and a clarinet. (a) Spectrogram of the original mixture using a frame size of 1024. (b) Spectrogram of the original mixture showing the centre frequencies of the estimated components in every frame.

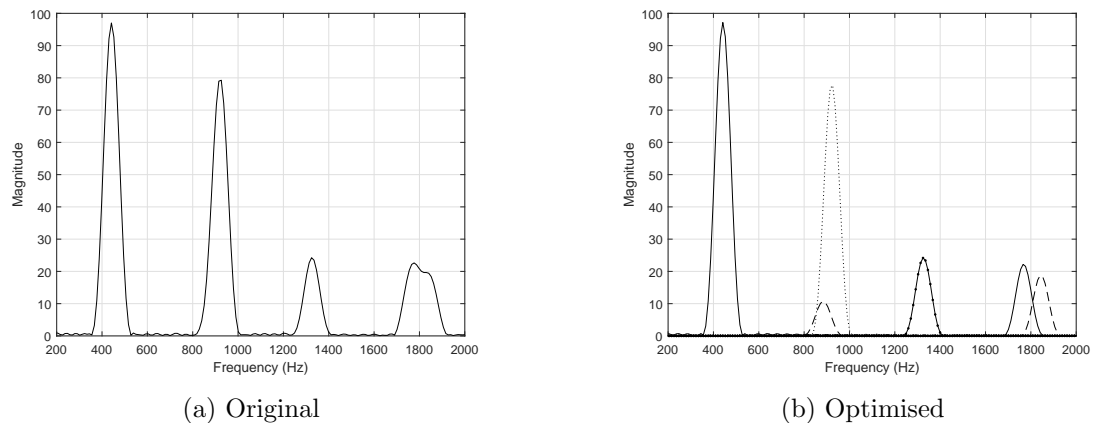


Figure 7.17: Magnitude spectra of the original mixture and the estimated components for a single frame of the spectrogram in Figure 7.16 at $t = 1.5$ s ($f_s = 44.1$ kHz).

The optimisation-based approach presented here is far from complete and additional research is required in order to fully understand its behaviour and to identify potential applications. The preliminary implementation that has been discussed here is highly dependent on the stationarity of the signals in every frame, which means that rapid transitions or high levels of noise might affect the accuracy of the estimated parameters. Selecting an appropriate value for δ during the analysis of each relevant peak is also important, given that an accurate detection of the main lobe of the spectral peak is crucial during the estimation of the number of interacting components involved and their parameters. Moreover, a different strategy is also necessary to provide the solver with an appropriate initial solution point for each parameter vector. The complexity of

the solution space associated with the cost function directly depends on the characteristics of the input signal, and hence, starting the minimisation process in the vicinity of the expected local minimum should prevent further errors in the estimated parameters.

Given the iterative note event-based multipitch detector and separation strategy presented in this thesis, the optimisation-based approach is also conceived as a correction stage in which the estimated pitch trajectories can be used to determine accurate initial points for the solver in every frame, while the optimised decomposition of each relevant peak could be used to confirm this information, or eventually to correct any estimation error.

7.3 Plan for the Future

Given the ideas presented in the previous section and the capabilities of the proposed semi-supervised audio source separation system, a set of priorities is presented and discussed here as a way to conclude the work of this thesis.

- It is believed that the development of the multiple-event extraction framework (Section 7.2.1) and the automatic clustering of note events (Section 7.2.4) represent the highest priorities in the further development of the proposed solution. The first one should improve the accuracy of the estimated pitch trajectories, reducing the number of false positives and false negatives, while the second one would allow the comparison of the obtained results with other unsupervised algorithms.
- Implementing an adaptive estimation of the inharmonicity coefficient inside the peak-picking algorithm, and designing a strategy to detect rapidly decaying spectral components, should allow the system to handle more effectively some special string instruments, such as piano and harp.
- The optimisation-based separation algorithm (Section 7.2.5) can then be used as a high-quality separation process, in which the estimated pitch trajectories of the iteratively detected note events are used to initialise the non-linear solver. This new stage should be able to detect and correct potential errors in the estimated pitch contours, and provide an effective separation of overlapping content.
- Finally, the high-quality separated note events and their corresponding pitch contours can be used to extract additional information, such as onsets and offsets, or to generate high-quality MIDI files for the original input mixtures and the separated sources.

References

- [1] E. Vincent, N. Bertin, R. Gribonval, and F. Bimbot, “From blind to guided audio source separation: how models and side information can improve the separation of sound,” *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 107–115, 2014.
- [2] D. Howard and J. Angus, *Acoustics and psychoacoustics*. Elsevier, fourth ed., 2009.
- [3] G. J. Brown and M. Cooke, “Computational auditory scene analysis,” *Computer Speech & Language*, vol. 8, pp. 297–336, Oct 1994.
- [4] B. Gao, *Single-channel blind source separation*. Ph.D, University of Newcastle, 2011.
- [5] A. Dhar, A. Senapati, and J. Sekhar Roy, “Direction of arrival estimation for smart antenna using a combined blind source separation and multiple signal classification algorithm,” *Indian Journal of Science and Technology*, vol. 9, no. 18, 2016.
- [6] Y. Zhou, J. Gao, W. Chen, and P. Frossard, “Seismic simultaneous source separation via patchwise sparse representation,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 9, pp. 1–14, 2016.
- [7] F. Negro, S. Muceli, A. M. Castronovo, A. Holobar, and D. Farina, “Multi-channel intramuscular and surface EMG decomposition by convolutive blind source separation,” *Journal of Neural Engineering*, vol. 49, pp. 1–45, 2016.
- [8] X. Duan, J. Wang, T. Peng, F. Li, S. Liu, and T. Liu, “Blind separation of permuted alias image with motion blurred using image enhancement in NSCT domain,” *Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 3, pp. 421–432, 2016.
- [9] E. Vincent, C. Févotte, R. Gribonval, X. Rodet, E. Le Carpentier, L. Benaroya, A. Röbel, and F. Bimbot, “A tentative typology of audio source separation tasks,” in *Proceedings of the International Symposium on Independent Component Analysis and Blind Signal Separation*, pp. 715–720, 2003.

- [10] D. FitzGerald, “Upmixing from mono - a source separation approach,” in *Proceedings of the 17th International Conference on Digital Signal Processing*, 2011.
- [11] J. Woodruff, B. Pardo, and R. Dannenberg, “Remixing stereo music with score-informed source separation,” in *Proceedings of the 7th International Conference on Music Information Retrieval*, pp. 314–319, 2006.
- [12] M. Cobos, J. J. López, A. González, and J. Escolano, “Stereo to wave-shield synthesis music up-mixing: an objective and subjective evaluation,” in *Proceedings of the 3rd International Symposium on Communications, Control, and Signal Processing*, pp. 1279–1284, 2008.
- [13] K. Reindl, Y. Zheng, and W. Kellermann, “Speech enhancement for binaural hearing aids based on blind source separation,” in *Proceedings of the 4th International Symposium on Communications, Control and Signal Processing*, pp. 3–5, 2010.
- [14] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, “Automatic music transcription: challenges and future directions,” *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 407–434, 2013.
- [15] S. Smita, S. Biswas, and S. S. Solanki, “Audio signal separation and classification: a review paper,” *International Journal of Innovative Research in Computer and Communication Engineering.*, vol. 2, no. 11, pp. 6960–6966, 2014.
- [16] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, “Content-based music information retrieval: current directions and future challenges,” *Proceedings of the IEEE*, vol. 96, no. 4, pp. 668–696, 2008.
- [17] F. J. Rodríguez Serrano, *Separación de fuentes sonoras en señales musicales*. Ph.D, University of Jaen, 2014.
- [18] G. Loy, *Musimathics*. The MIT Press, 2006.
- [19] Acoustical Society of America, “Acoustical terminology (ANSI S1.1-1994 R2004),” tech. rep., American National Standard Institute, 2004.
- [20] T. Letowski, “Timbre, tone color and sound quality: concepts and definitions,” *Archives of Acoustics*, vol. 17, no. 1, pp. 17–30, 1992.

-
- [21] H. Helmholtz, *On the sensations of tone: as a physiological basis for the theory of music*. Dover Publications, second ed., 1954.
- [22] J. Ponce de León Vázquez, *Análisis y síntesis de señales de audio a través de la transformada wavelet continua y compleja: el algoritmo CWAS*. Ph.D, University of Zaragoza, 2012.
- [23] G. Siamantas, *An iterative, residual-based approach to unsupervised musical source separation in single-channel mixtures*. Ph.D, University of York, 2009.
- [24] J. Jones, “Confirmation report,” tech. rep., 2000.
- [25] S. Hendry, *Inharmonicity of piano strings*. M.Sc, University of Edinburgh, 2008.
- [26] Z. Duan, Y. Zhang, C. Zhang, and Z. Shi, “Unsupervised single-channel music source separation by average harmonic structure modeling,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, no. 4, pp. 766–778, 2008.
- [27] J. O. Smith, *Spectral audio signal processing*. W3K Publishing, 2011.
- [28] M. R. Every, *Separation of musical sources and structure from single-channel polyphonic recordings*. Ph.D, University of York, 2006.
- [29] N. J. Bryan, *Interactive sound source separation*. Ph.D, Stanford University, 2014.
- [30] N. Ma, “Cochleagram representation of sound.” <http://staffwww.dcs.shef.ac.uk/people/N.Ma/resources/ratemap/>, [Online] Accessed on 29 Aug. 2016.
- [31] B. Gold, N. Morgan, and D. Ellis, *Speech and audio signal processing: processing and perception of speech and music*. John Wiley & Sons, second ed., 2011.
- [32] K. Fitz and L. Haken, “On the use of time-frequency reassignment in additive sound modeling,” *Journal of the Audio Engineering Society*, vol. 50, pp. 879–893, 2002.
- [33] P. Flandrin, A. Francois, and E. Chassande-Mottin, “Time-frequency reassignment: from principles to algorithms,” in *Applications in Time-Frequency Signal Processing*, pp. 179–204, CRC Press, 2002.
- [34] A. Ahrabian and D. Mandic, “Selective time-frequency reassignment based on synchrosqueezing,” *IEEE Signal Processing Letters*, vol. 9908, pp. 2039–2043, 2015.

- [35] A. Hyvärinen and E. Oja, “Independent component analysis: algorithms and applications,” *Journal of the International Neural Network Society*, vol. 13, no. 4-5, pp. 411–30, 2000.
- [36] D. Langlois, S. Chartier, and D. Gosselin, “An introduction to independent component analysis: InfoMax and FastICA algorithms,” *Tutorials in Quantitative Methods for Psychology*, vol. 6, no. 1, pp. 31–38, 2010.
- [37] J. F. Cardoso, “Blind signal separation: statistical principles,” *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009–2025, 1998.
- [38] J. Heeris, *Single channel blind source separation using independent subspace analysis*. B.Sc, University of Western Australia, 2007.
- [39] M. A. Casey and A. Westner, “Separation of mixed audio sources by independent subspace analysis,” *Proceedings of the International Computer Music Conference*, pp. 154–161, 2000.
- [40] J. Taghia and M. A. Doostari, “Subband-based single-channel source separation of instantaneous audio mixtures,” *World Applied Sciences Journal*, vol. 6, no. 6, pp. 784–792, 2009.
- [41] R. Kronland Martinet, J. Morlet, and A. Grossman, “Analysis of sound patterns through wavelet transforms,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 01, no. 02, pp. 273–302, 1987.
- [42] G. Tzanetakis, G. Essl, and P. Cook, “Audio analysis using the discrete wavelet transform,” in *Proceedings of the Conference in Acoustics and Music Theory Applications*, 2001.
- [43] A. Paradzinets, H. Harb, and L. Chen, “Use of continuous wavelet-like transform in automated music transcription,” in *Proceedings of the 14th European Signal Processing Conference*, 2006.
- [44] Y. Litvin and I. Cohen, “Single-channel source separation of audio signals using Bark scale wavelet packet decomposition,” *Journal of Signal Processing Systems*, vol. 65, no. 3, pp. 339–350, 2011.
- [45] T. Nakamura and H. Kameoka, “Fast signal reconstruction from magnitude spectrogram of continuous wavelet transform based on spectrogram consistency,” in *Proceedings of the 17th International Conference on Digital Audio Effects*, pp. 1–7, 2014.

-
- [46] S. K. Tjoa and K. J. R. Liu, “Factorization of overlapping harmonic sounds using approximate matching pursuit,” in *Proceedings of the 12th International Conference on Music Information Retrieval*, pp. 257–262, 2011.
- [47] D. Zantalis, *Guided matching pursuit and its application to sound source separation*. Ph.D, University of York, 2016.
- [48] G. Siamantas, M. R. Every, and J. E. Szymanski, “Separating sources from single-channel musical material: a review and future directions,” in *Proceedings of the Digital Music Research Network Workshop*, pp. 2–5, 2006.
- [49] A. Liutkus, J. L. Durrieu, L. Daudet, and G. Richard, “An overview of informed audio source separation,” in *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services*, pp. 3–6, 2013.
- [50] D. Wang, “Time-frequency masking for speech separation and its potential for hearing aid design,” *Trends in Amplification*, vol. 12, no. 4, pp. 332–353, 2008.
- [51] D. FitzGerald and R. Jaiswal, “On the use of masking filters in sound source separation,” in *Proceedings of the 15th International Conference on Digital Audio Effects*, pp. 1–7, 2012.
- [52] P. Li, Y. Guan, B. Xu, and W. Liu, “Monaural speech separation based on computational auditory scene analysis and objective quality assessment of speech,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, pp. 2014–2023, nov 2006.
- [53] L. A. Drake, J. C. Rutledge, J. Zhang, and A. Katsaggelos, “A computational auditory scene analysis-enhanced beamforming approach for sound source separation,” *EURASIP Journal on Advances in Signal Processing*, pp. 1–17, 2009.
- [54] S. Abdallah and M. D. Plumbley, “If the independent components of natural images are edges, what are the independent components of natural sounds?,” in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation*, pp. 534–539, 2001.
- [55] G. J. Jang, T. W. Lee, and Y. H. Oh, “Single-channel signal separation using time-domain basis functions,” *IEEE Signal Processing Letters*, vol. 10, no. 6, pp. 168–171, 2003.

- [56] M. S. Pedersen, D. Wang, J. Larsen, and U. Kjems, "Overcomplete blind source separation by combining ICA and binary time-frequency masking," in *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing*, pp. 15–20, 2005.
- [57] D. Barry, E. Coyle, D. FitzGerald, and R. Lawlor, "Single-channel source separation using short-time independent component analysis," in *Proceedings of the 119th AES Convention*, pp. 1–6, 2005.
- [58] M. Davies and C. J. James, "Source separation using single-channel ICA," *Signal Processing*, vol. 87, no. 8, pp. 1819–1832, 2007.
- [59] D. Mika and P. Kleczkowski, "ICA-based single channel audio separation: new bases and measures of distance," *Archives of Acoustics*, vol. 36, no. 2, pp. 311–331, 2011.
- [60] J. S. Calderón Piedras, Á. D. Orjuela Cañón, and D. A. Sanabria Quiroga, "Blind source separation from single-channel audio recording using ICA algorithms," in *Proceedings of the 19th IEEE Symposium on Image, Signal Processing and Artificial Vision*, pp. 1–5, 2014.
- [61] P. Bofill, "Identifying single source data for mixing matrix estimation in instantaneous blind source separation," *Lecture Notes in Computer Science*, pp. 759–767, 2008.
- [62] Y. Li, W. Nie, and F. Ye, "A complex mixing matrix estimation algorithm based on single source points," *Circuits, Systems, and Signal Processing*, vol. 34, no. 11, pp. 3709–3723, 2015.
- [63] Y. Li, W. Nie, F. Ye, and Y. Lin, "A mixing matrix estimation algorithm for underdetermined blind source separation," *Circuits, Systems, and Signal Processing*, vol. 35, no. 9, pp. 3367–3379, 2016.
- [64] R. A. Irizarry, *Statistics and music: fitting a local harmonic model to musical sound signals*. Ph.D, University of California at Berkeley, 1998.
- [65] T. Virtanen and A. Klapuri, "Separation of harmonic sounds using multipitch analysis and iterative parameter estimation," in *Proceedings of the IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, pp. 83–86, 2001.
- [66] D. Wang and Q. Huang, "Single-channel music source separation based on harmonic structure estimation," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 848–851, 2009.

-
- [67] Y. Wang, H. Wang, B. Zhu, and X. Wang, “Single-channel polyphonic signal separation based on a novel multi-F0 estimation method,” in *Proceedings of the 14th IEEE International Conference on Communication Technology*, pp. 1334–1338, 2012.
- [68] Y. G. Zhang and C. Zhang, “Separation of music signals by harmonic structure modeling,” *Advances in Neural Information Processing Systems*, vol. 18, p. 1617, 2006.
- [69] 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*, vol. 14, no. 5, pp. 1845–1856, 2006.
- [70] Y. Li, J. Woodruff, and D. Wang, “Monaural musical sound separation based on pitch and common amplitude modulation,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 7, pp. 1361–1371, 2009.
- [71] C. L. Hsu, D. Wang, J. S. R. Jang, and K. Hu, “A tandem algorithm for singing pitch extraction and voice separation from music accompaniment,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 5, pp. 1454–1463, 2012.
- [72] P. P. Ingale and S. L. Nalbalwar, “Singing voice separation using mono-channel mask,” *International Journal of Speech Technology*, vol. 21, no. 2, pp. 309–318, 2018.
- [73] D. FitzGerald, M. Cranitch, and E. Coyle, “Extended nonnegative tensor factorisation models for musical sound source separation,” *Computational Intelligence and Neuroscience*, no. 2, 2008.
- [74] R. Hennequin, R. Badeau, and B. David, “Time-dependent parametric and harmonic templates in non-negative matrix factorization,” in *Proceedings of the 13th International Conference on Digital Audio Effects*, pp. 1–8, 2010.
- [75] J. L. Durrieu and J. P. Thiran, “Musical audio source separation based on user-selected F0 track,” in *Proceedings of the 10th International Conference on Latent Variable Analysis and Signal Separation*, pp. 438–445, 2012.
- [76] E. Ochiai, T. Fujisawa, and M. Ikehara, “Vocal separation by constrained non-negative matrix factorization,” in *Processing of the APSIPA Annual Summit and Conference*, pp. 480–483, 2015.

- [77] Y.-J. Lin, Y.-L. Wang, A. Su, and L. Su, “Separation of musical notes with highly overlapping partials using phase and temporal constrained complex matrix factorization,” in *Proceedings of the 18th International Conference on Digital Audio Effects*, Trondheim, Norway, 2015.
- [78] H. Deif, *Single channel separation of vocals from harmonic and percussive instruments*. Ph.D, Brunel University London, 2017.
- [79] B. Fuentes, R. Badeau, and G. Richard, “Blind harmonic adaptive decomposition applied to supervised source separation,” in *Proceedings of the 20th European Signal Processing Conference*, pp. 2654–2658, 2012.
- [80] M. Zivanovic, “Harmonic bandwidth companding for separation of overlapping harmonics in pitched signals,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 5, pp. 898–908, 2015.
- [81] T. W. Parsons, “Separation of speech from interfering speech by means of harmonic selection,” *The Journal of the Acoustical Society of America*, vol. 60, no. 1976, p. 911, 1976.
- [82] J. Ponce de León Vázquez and J. R. Beltrán Blázquez, “Blind separation of overlapping partials in harmonic musical notes using amplitude and phase reconstruction,” *EURASIP Journal on Advances in Signal Processing*, no. 223, pp. 1–16, 2012.
- [83] Z. Rafii, A. Liutkus, F. R. Stoter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, “An overview of lead and accompaniment separation in music,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.
- [84] E. M. Grais, M. U. Sen, and H. Erdogan, “Deep neural networks for single-channel source separation,” in *Proceedings of the 39th IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3734–3738, IEEE, 2014.
- [85] Z. C. Fan, J. S. R. Jang, and C. L. Lu, “Singing voice separation and pitch extraction from monaural polyphonic audio music via DNN and adaptive pitch tracking,” in *Proceedings of the 2nd IEEE International Conference on Multimedia Big Data*, pp. 178–185, 2016.
- [86] S. Uhlich, F. Giron, and Y. Mitsufuji, “Deep neural network-based instrument extraction from music,” in *Proceedings of the 15th IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2135–2139, 2015.

-
- [87] E. M. Grais, G. Roma, A. Simpson, and M. D. Plumbley, “Two-stage single-channel audio source separation using deep neural networks,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 9, pp. 1469–1479, 2017.
- [88] P. Chandna, M. Miron, J. Janer, and E. Gómez, “Monoaural audio source separation using deep convolutional neural networks,” in *Proceedings of the 13th International Conference on Latent Variable Analysis and Signal Separation*, pp. 258–266, 2017.
- [89] M. Miron, J. Janer, and E. Gómez, “Monoaural score-informed source separation for classical music using convolutional neural networks,” in *Proceedings of the 18th International Conference on Music Information Retrieval*, pp. 55–62, 2017.
- [90] E. Cakir, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, “Convolutional recurrent neural networks for polyphonic sound event detection,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.
- [91] Z. Duan, J. Han, and B. Pardo, “Multi-pitch streaming of harmonic sound mixtures,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 22, no. 1, pp. 138–150, 2014.
- [92] M. R. Every and J. E. Szymanski, “A spectral-filtering approach to music signal separation,” in *Proceedings of the 7th International Conference on Digital Audio Effects*, pp. 197–200, 2004.
- [93] M. R. Every and J. E. Szymanski, “Separation of overlapping impulsive sounds by band-wise noise interpolation,” in *Proceedings of the 8th International Conference on Digital Audio Effects*, pp. 20–22, 2005.
- [94] M. R. Every, “Separating harmonic and inharmonic note content from real mono recordings,” in *Proceedings of the Digital Music Research Network Workshop*, pp. 1–5, 2005.
- [95] P. Smaragdis and G. J. Mysore, “Separation by humming: user-guided sound extraction from monophonic mixtures,” in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 69–72, 2009.
- [96] E. Vincent, R. Gribonval, and C. Févotte, “Performance measurement in blind audio source separation,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, pp. 1462–1469, Jul 2006.

- [97] E. Cano, D. FitzGerald, and K. Brandenburg, “Evaluation of quality of sound source separation algorithms: human perception vs quantitative metrics,” in *Proceedings of the 24th IEEE European Signal Processing Conference*, no. 1, pp. 1758–1762, 2016.
- [98] C. Févotte, R. Gribonval, and E. Vincent, “BSS EVAL toolbox user guide,” Tech. Rep. 1706, Institut de Recherche en Informatique et Systèmes Aléatoires, 2005.
- [99] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, “Subjective and objective quality assessment of audio source separation,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 7, pp. 2046–2057, 2011.
- [100] B. Gowrishankar and B. Nagappa, “An exhaustive review of automatic music transcription techniques: Survey of music transcription techniques,” in *Proceedings of the International Conference on Signal Processing, Communication, Power and Embedded System*, pp. 140–152, 2016.
- [101] L. Gao, L. Su, Y.-H. Yang, and T. Lee, “Polyphonic piano note transcription with non-negative matrix factorization of differential spectrogram,” in *Proceedings of the 42nd IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 291–295, 2017.
- [102] J. Salamon and E. Gómez, “Melody extraction from polyphonic music signals using pitch contour characteristics,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 6, pp. 1759–1770, 2012.
- [103] J. Salamon, E. Gómez, D. P. W. Ellis, and G. Richard, “Melody extraction from polyphonic music signals: approaches, applications, and challenges,” *IEEE Signal Processing Magazine*, vol. 31, no. 2, pp. 118–134, 2014.
- [104] D. Basaran, S. Essid, and G. Peeters, “Main melody extraction with source-filter NMF and CRNN,” in *Proceedings of the 19th International Conference on Music Information Retrieval*, pp. 82–89, 2018.
- [105] C. C. Wang and J. S. R. Jang, “Improving query-by-singing/humming by combining melody and lyric information,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 4, pp. 798–806, 2015.
- [106] A. Lv and G. Liu, “An effective design for fast query-by-humming system with melody segmentation and feature extraction,” in *Proceedings of the 1st IEEE International Conference on Computer Systems, Electronics and Control*, pp. 752–757, 2017.

-
- [107] M. Jaczynska, P. Bobinski, and A. Pietrzak, “Music recognition algorithms using queries by example,” *Joint Conference - Acoustics*, pp. 1–4, 2018.
- [108] P. T. Selvan and V. Vaishnavi, “Singing pitch extraction and voice separation from music accompaniment using trend estimation and tandem algorithm,” in *Proceedings of the IEEE International Conference on Emerging Trends in Computing, Communication and Nanotechnology*, pp. 232–235, 2013.
- [109] D. Y. Loni and S. Subbaraman, “Singing voice identification using harmonic spectral envelope,” in *Proceedings of the International Conference on Information Processing*, pp. 119–123, 2015.
- [110] T. Ratanpara and N. Patel, “Singer identification using perceptual features and cepstral coefficients of an audio signal from Indian video songs,” *Journal on Audio, Speech, and Music Processing*, no. 1, 2015.
- [111] M. Bay, A. F. Ehmann, and J. S. Downie, “Evaluation of multiple-F0 estimation and tracking systems,” in *Proceedings of the 10th International Conference on Music Information Retrieval*, pp. 315–320, 2009.
- [112] K. Rychlicki-Kicior and B. Stasiak, “Multipitch estimation using judge-based model,” *Bulletin of the Polish Academy of Sciences*, vol. 62, no. 4, pp. 751–757, 2014.
- [113] A. Klapuri, “Multipitch analysis of polyphonic music and speech signals using an auditory model,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, no. 2, pp. 255–266, 2008.
- [114] J. Ponce de León Vázquez, F. Beltrán, and J. R. Beltrán Blázquez, “A complex wavelet based fundamental frequency estimator in single-channel polyphonic signals,” in *Proceedings of the 16th International Conference on Digital Audio Effects*, pp. 1–8, 2013.
- [115] W. Zhang, Z. Chen, and F. Yin, “Main melody extraction from polyphonic music based on modified Euclidean algorithm,” *Applied Acoustics*, vol. 112, pp. 70–78, 2016.
- [116] L. Su and Y.-H. Yang, “Combining spectral and temporal representations for multipitch estimation of polyphonic music,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1600–1612, 2015.

- [117] K. Rychlicki-Kicior, B. Stasiak, and M. Yatsymirskyy, “Multipitch estimation using multiple transformation analysis,” in *Proceedings of the 1st IEEE International Conference on Data Stream Mining & Processing*, pp. 299–304, 2016.
- [118] Z. Duan, B. Pardo, and C. Zhang, “Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 8, pp. 2121–2133, 2010.
- [119] C. de Andrade Scatolini, G. Richard, and B. Fuentes, “Multipitch estimation using a PLCA-based model: impact of partial user annotation,” in *Proceedings of the 40th IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 186–190, 2015.
- [120] S. Karimian Azari, A. Jakobsson, J. R. Jensen, and M. Christensen, “Multi-pitch estimation and tracking using Bayesian inference in block sparsity,” in *Proceedings of the 23rd European Signal Processing Conference*, no. 2, pp. 16–20, 2015.
- [121] L. Gao and T. Lee, “Multi-pitch estimation based on sparse representation with pre-screened dictionary,” in *Proceedings of the 17th IEEE International Workshop on Multimedia Signal Processing*, pp. 1–6, 2015.
- [122] F. Elvander, J. Sward, and A. Jakobsson, “Online estimation of multiple harmonic signals,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 2, pp. 273–284, 2017.
- [123] R. M. Bittner, A. Wang, and J. P. Bello, “Pitch contour tracking in music using harmonic locked loops,” in *Proceedings of the 42th IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 191–195, 2017.
- [124] T. Cheng, S. Dixon, and M. Mauch, “Improving piano note tracking by HMM smoothing,” in *Proceedings of the 23rd European Signal Processing Conference*, pp. 2009–2013, 2015.
- [125] J. J. Valero Más, E. Benetos, and J. M. Iñesta, “A supervised classification approach for note tracking in polyphonic piano transcription,” *Journal of New Music Research*, vol. 47, no. 3, pp. 249–263, 2018.
- [126] H. Kirchhoff, S. Dixon, and A. Klapuri, “Missing template estimation for user-assisted music transcription,” in *Proceedings of the 38th IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 26–30, 2013.

-
- [127] V. Arora and L. Behera, “Multiple F0 estimation and source clustering of polyphonic music audio using PLCA and HMRFs,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 2, pp. 278–287, 2015.
- [128] S. Sigtia, E. Benetos, and S. Dixon, “An end-to-end neural network for polyphonic piano music transcription,” *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 5, pp. 927–939, 2016.
- [129] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, “RWC music database: music genre database and musical instrument sound database,” in *Proceedings of the 4th International Conference on Music Information Retrieval*, pp. 229–230, 2003.
- [130] S. Adavanne, A. Politis, J. Nikunen, and T. Virtanen, “Sound event localization and detection of overlapping sources using convolutional recurrent neural networks,” *IEEE Journal on Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 34–48, 2018.
- [131] E. Vincent, R. Gribonval, and M. D. Plumbley, “Oracle estimators for the benchmarking of source separation algorithms,” *Signal Processing*, vol. 87, no. 8, pp. 1933–1950, 2007.
- [132] E. Vincent and M. D. Plumbley, “BSS ORACLE toolbox version 2.1 user guide,” tech. rep., 2007.
- [133] N. J. Bryan and G. J. Mysore, “Interactive refinement of supervised and semi-supervised sound source separation estimates,” in *Proceedings of the 38th IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 883–887, 2013.
- [134] B. Laback, *The psychophysical bases of spatial hearing in acoustic and electric stimulation*. University of Vienna, 2013.
- [135] C. Faller, “Pseudostereophony revisited,” in *Proceedings of the 118th AES Convention*, 2005.
- [136] C. Uhle and P. Gamp, “Mono-to-Stereo Upmixing,” in *Proceedings of the 140th AES Convention*, 2016.
- [137] M. Lagrange, L. G. Martins, and G. Tzanetakis, “Semi-automatic mono to stereo upmixing using sound source formation,” in *Proceedings of the 122nd AES Convention*, 2007.
- [138] D. FitzGerald, “Harmonic/percussive separation using median filtering,” in *Proceedings of the 13th International Conference on Digital Audio Effects*, no. 1, pp. 10–13, 2010.

- [139] D. FitzGerald, “Vocal separation using nearest neighbours and median filtering,” in *Proceedings of the IET Irish Signals and Systems Conference*, pp. 1–5, 2012.
- [140] D. FitzGerald, “The good vibrations problem,” in *Proceedings of the 134th AES Convention*, 2013.
- [141] A. Delgado Castro, J. E. Szymanski, and G. Siamantas, “Onset detection via separation of harmonic content from musical notes,” in *Proceedings of the 10th York Doctoral Symposium on Computer Science and Electronic Engineering*, pp. 1–5, 2017.
- [142] A. Holzapfel, Y. Stylianou, A. C. Gedik, and B. Bozkurt, “Three dimensions of pitched instrument onset detection,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1517–1527, 2010.
- [143] B. Stasiak and J. Mońko, “Analysis of time-frequency representations for musical onset detection with convolutional neural network,” in *Proceedings of the Federated Conference on Computer Science and Information Systems*, vol. 8, pp. 147–152, 2016.
- [144] J. G. Arnal Barbedo and G. Tzanetakis, “Musical instrument classification using individual partials,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 1, pp. 111–122, 2011.
- [145] J. Wu, E. Vincent, S. A. Raczynski, T. Nishimoto, N. Ono, and S. Sagayama, “Polyphonic pitch estimation and instrument identification by joint modeling of sustained and attack sounds,” *IEEE Journal on Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1124–1132, 2011.
- [146] V. Arora and L. Behera, “Musical source clustering and identification in polyphonic audio,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 22, no. 6, pp. 1003–1012, 2014.
- [147] F. Yu and Y. Chen, “Musical instrument classification based on improved matching pursuit with instrument-specific atoms,” in *Proceedings of the 4th International Congress on Advanced Applied Informatics*, pp. 506–510, 2015.
- [148] S. Masood, S. Gupta, and S. Khan, “Novel approach for musical instrument identification using neural network,” in *Proceedings of the Annual IEEE India Conference*, pp. 1–5, 2015.

- [149] Y. Han, J. Kim, and K. Lee, “Deep convolutional neural networks for predominant instrument recognition in polyphonic music,” *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 25, no. 1, pp. 208–221, 2016.