Spectrogram Track Detection
An Active Contour Algorithm

THOMAS A. LAMPERT

Ph.D. Thesis

This thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

THE UNIVERSITY of York

Advanced Computer Architecture Group
Department of Computer Science
United Kingdom

March 2010

Abstract

In many areas of science, near-periodic phenomena represent important information within time-series data. This thesis takes the example of the detection of non-transitory frequency components in passive sonar data, a problem which finds many applications. This problem is typically transformed into the pattern recognition domain by representing the time-series data as a spectrogram, in which slowly varying periodic signals appear as curvilinear tracks.

The research is initiated with a survey of the literature, which is focused upon research into the detection of tracks within spectrograms. An investigation into low-level feature detection reveals that none of the evaluated methods perform adequately within the low signal-to-noise ratios of real-life spectrograms and, therefore, two novel feature detectors are proposed. An investigation into the various sources of information available to the detection process shows that the most simple of these, the individual pixel intensity values, used by most existing algorithms, is not sufficient for the problem. To overcome these limitations, a novel low-level feature detector is integrated into a novel active contour track detection algorithm, and this serves to greatly increase detection rates at low signal-to-noise ratios. Furthermore, the algorithm integrates *a priori* knowledge of the harmonic process, which describes the relative positions of tracks, to augment the available information in difficult conditions.

Empirical evaluation of the algorithm demonstrates that it is effective at detecting tracks at signal-to-noise ratios as low as: 0.5 dB with vertical; 3 dB with oblique; and 2 dB with sinusoidal variation of harmonic features. It is also concluded that the proposed potential energy increases the active contour's effectiveness in detecting all the track structures by a factor of eight (as determined by the line location accuracy measure), even at relatively high signal-to-noise ratios, and that incorporating *a priori* knowledge of the harmonic process increases the detection rate by a factor of two.

Contents

List of Tables

List of Figures

List of Algorithms

Acknowledgements

The time spent researching and documenting my Ph.D. has been both exciting and tumultuous. Many people have supported me, both academically and personally. There are many with whom I have had mere incidences, however, they have all, in some way, influenced that which is presented in front of you now. If I try to list all the people who have influenced my work, I will fail, and I therefore list those which are, for one reason or another, most prominent in my mind.

Needless to say, the academic content of this thesis has been primarily shaped by my supervisor, Dr. Simon O'Keefe, whose knowledge, guidance, encouragement, and support have allowed me to produce this document of my journey and to become an independent researcher. Dr. Nick Pears and Dr. Richard Harvey have both read, understood, and examined me upon its content and I greatly appreciate their dedication to the academic standard which has instilled a measure of self-confidence in my work. Whilst elucidating the problems tackled by this research I was very fortunate to have the practical and theoretical guidance of Jim Nicholson, who I would also like to thank for his finely tuned sense for grammatical correctness. Furthermore, Dr. Duncan Williams has supported my research and encouraged its dissemination and continuation. Filo Ottaway has always demonstrated a dedication to the students of this department, far beyond that which could be expected of her. I appreciate the encouragement, support, friendship, and dedication that she has given me. To all of the academic, administrative and industrial supporters who have encouraged me during the past four years, I am deeply grateful. I would also like to acknowledge the inspirational teachers and academics who have encouraged and contributed to my earlier education, in particular: Ms. Henderson, Mrs. Smith, Mrs. Mills, Mr. McPherson, and Prof. Everson.

My experience of Ph.D. studies leads me to believe that it is not only a journey of which the goal is to reach an understanding of research and science but that it is also a medium through which it is possible to gain a deeper understanding of oneself. As such, it is not only an exciting and enjoyable experience but it can also present worrying and disorienting challenges, and this is where the limitation of academic support is surpassed by that of family and friends. My parents, Andrew and Kathryn Lampert, have, throughout my life, provided me with the best possible support, encouragement and love, as has my sister, Harriet Lampert. I am grateful to them for all the moments that I have spent at home over the previous four years, where I have been able to relax and enjoy times away from the pressure of work. Whilst there, many days have been spent relaxing and contemplatively discussing thoughts next to rivers in the Cambridgeshire fens, fishing with my dear friend Dan Fordham. When I was in need of escape I could always rely on another close friend Olivier Guillemot to help me recover perspective. It was during one such adventure

in the countryside of the Rône Alps that I solved one of the key problems of this research. Whilst living here in York I have come to rely upon the friendship and support of some interesting and supportive people. Marcello Romero has been a good friend since my first months here and has supported me, both academically and personally, throughout my research. Eliza has punctuated my day with all manner of interesting discussion and has brightened up, what would otherwise be, a dull office. Leo Freitas has been a true friend, with whom I have had many discussions and memorable nights in various bars and pubs of York drinking the fine beer of the city. Every time that we play I am grateful to my friends who are the members of Saville Law; Andre, Leandro and Lorenzo, with whom I have a means of unconstrained expression; I have truly enjoyed what we have together. I would like to thank Burcu Can for our discussions, photography, and her delightful cooking. Frank Zeyda has encouraged my abilities in music and with whom I have enjoyed drinks, discussions and parties. Pierre Andrews is someone who has helped me in my work, has been a friend, and who has almost killed me in the Alps, we have spent some unique moments together on some spectacular adventures. I would particularly like to emphasise my fortune with the random events that have resulted in my friendship with Bere. There is no doubt that she has unselfishly offered me far more than anyone could ever wish for, I am happy to have spent every moment that we have had together, with such a kind person—*mi amiga querida*. Clarisse has been an extremely kind friend, I have enjoyed her excellent culinary skills, and she has supported me when I needed it most. Silvana, housed me when I was homeless and has been an excellent, attentive friend, I wish her luck with her future endeavours. Laure injected a little French madness into my life, *merci mon petit Français*. I thank Juan for distracting me from work with educational debates, of sorts. Isabelle for our bucolic adventures. Malihe for forcing me to dance. Berna has, well, been Berna, and it has been fantastic to know her. During my days in the lab, the most enjoyable parts have been spent over lunch, the food was not so good, but the company transformed these times into something to look forward to, for this I would also like to thank Napol, Tobias, Simon, José, and Marek. Furthermore, I would like to thank: Richard, Osmar, Simone, Lichi, Ahmad, Shailesh, Peng, Lin, and Ping, for making the department a more interesting place to be, each in your own particular way; Guy, George, Stewart, Alan, and Saira, for our times in Manchester; and Julia, Dan, Katharina, Valentina, Gioia, and Angelika, for our adventures in Spain. Finally, I cannot finish these acknowledgements without expressing my appreciation for Tatjana, her dedication to helping me complete this thesis, her love, and her unbridled support during my most difficult moments, have brought respite during the past year.

As I write these acknowledgements, I come to realise that the work presented here represents far more than a mere document of my research. To all of the uniquely interesting people that I have mentioned here, I most sincerely hope that this is not the end of the experiences that we share together.

To the memory of Mr. Peter Burrows—a dedicated and inspirational teacher, whose encouragement to study mathematics and computer science has most influenced my path through academia.

Declaration

Parts of the following research have been previously presented or published in:

- Lampert T. and O'Keefe, S., 2010. An Active Contour Model for Spectrogram Track Detection. Pattern Recognition Letters 31(10), 1201–1206.
- Lampert T. and O'Keefe, S., February 2010. A Survey of Spectrogram Track Detection Algorithms. Applied Acoustics 71(2), 87–100.
- Lampert T. and O'Keefe, S., 'Machine Learning of Harmonic Relationships which Maximise Source Detection and Discrimination', *NATO & DSTL Workshop on Machine Intelligence for Autonomous Operations*, Lerici, Italy, October 7–8, 2009.
- Lampert, T., Pears, N. and O'Keefe, S., 2009. A Multi-Scale Piecewise Linear Feature Detector for Spectrogram Tracks. In: *Proceedings of the IEEE 6th International Conference on Advanced Video and Signal Based Surveillance*. pp. 330–335, Genoa, Italy, September 2–4.
- Lampert, T., O'Keefe, S. and Pears, N., 2009. Line Detection Methods for Spectrogram Images. In: *Proceedings of 6th International Conference on Computer Recognition Systems.* Vol. 57 of Advances in Intelligent and Soft Computing, Springer, pp. 127–134.
- Lampert, T. and O'Keefe, S., 2009. A Comparison Framework for Spectrogram Track Detection Algorithms. In: *Proceedings of 6th International Conference on Computer Recognition Systems.* Vol. 57 of Advances in Intelligent and Soft Computing, Springer, pp. 119–126.
- Lampert, T. and O'Keefe, S., 2008. Active Contour Detection of Linear Patterns in Spectrogram Images. In: *Proceedings of the 19th International Conference on Pattern Recognition*. pp. 1–4, Tampa, Florida, USA, December 8–11.

This thesis has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree other than Doctor of Philosophy of the University of York. This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by explicit references.

I hereby give consent for my thesis, if accepted, to be made available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ..... (candidate)

Date .....

Chapter 1

Introduction

"*If you cause your ship to stop, and place the head of a long tube in the water*
and place the outer extremity to your ear,
you will hear ships at a great distance from you."

— Leonardo da Vinci, 1452–1519.

In many endeavours of science, pattern recognition in particular, there exists the problem of detecting near-periodic non-stationary phenomena within time series data. The continuous signal in which a phenomenon is embedded is measured, segmented in time, and frequency decomposition is performed on each section. The purpose of the analysis is to determine whether there exists a frequency component, or pattern of frequency components, within each of the segmented sections of the continuous signal. This bounds the assumption that the frequency component is stationary within each segmented section. A typical representation for such data is a spectrogram (also known as a LOFARgram, periodogram, sonogram, or spectral waterfall), in which time and frequency are variables along orthogonal axes, and intensity is representative of the power observed at a particular time and frequency. This forms a visual representation of the frequency-time variation of the original time-series data using the Short-Term Fourier Transform (STFT) [7, 6]. If a slowly varying frequency component exists within the time-series, it will appear over several consecutive time segments, and the resulting spectrogram will contain a track; a discrete set of points that exist in consecutive time frames of the spectrogram, each point related to the frequency component(s) of the time-series data. Consequently, detecting the tracks within a spectrogram determines the presence and state of a periodic or near-periodic phenomena in the original time-series data.

The problem of detecting tracks in spectrograms has been investigated since the spectrogram's introduction in the mid 1940s by Koenig et al. [101]. Research into the use of automatic detection methods increased with the advent of reliable computational algorithms during the 1980s, 1990s and early 21st century. The research area has attracted contributions from a variety of backgrounds, ranging from statistical modelling [137], image processing [3, 57] and expert systems [117]. The problem can be compounded, not only by a low Signal-to-Noise Ratio (SNR) in a spectrogram,

which is the result of weak periodic phenomena embedded within noisy time-series data, but also by the variability of a track's structure with time. This can vary greatly depending upon the nature of the observed phenomenon, but typically the structure arising from signals of interest, can vary from vertical straight tracks (no variation with time) and oblique straight tracks (uniform frequency variation), to undulating and irregular tracks. A good detection strategy should be able to cope with all of these.

In the broad sense this "problem arises in any area of science where periodic phenomena are evident and in particular signal processing" [148]. In practical terms, the problem forms a critical stage in the detection and classification of sources in passive sonar systems, the analysis of speech data and the analysis of vibration data—the outputs of which could be the detection of a hostile torpedo or of an aeroplane engine which is malfunctioning. Applications within these areas are wide and include identifying and tracking marine mammals via their calls [130, 125], identifying ships, torpedoes or submarines via the noise radiated by their mechanical movements such as propeller blades and machinery [196, 38], distinguishing underwater events such as ice cracking [68] and earth quakes [86] from different types of source, meteor detection, speech formant tracking [163], and so on. The research presented in this thesis is applicable to any area of science in which it is necessary to detect frequency components within time-series data.

There exist two distinct approaches to this problem: the time domain and the frequency domain. A discussion of the differences between the two has been presented by Wold [185] and reviews of methods which are applied in the time domain have been presented by Kootsookos [105] and Quinn and Hannan [149]. In summary, the transformation of a time domain signal into the frequency domain often allows more efficient analysis to be performed [32]. The transformation also has the effect of quantising a series' broadband noise into the spectrum of frequency bins, and therefore, the SNR of a narrowband feature in the time series is enhanced in the frequency domain [72]. Nevertheless, when constructing a 'conventional' spectrogram image the phase information is lost and, therefore, frequency domain methods should be applied to areas in which the time of measurement commencement is not important. The transfer of the signal from the time domain into the frequency domain allows for the application of algorithms from a wide variety of research disciplines, as highlighted in the literature review of this thesis (see Chapter 2), whereas generally time domain analysis is restricted to the fields of signal processing and statistical analysis.

The passive sonar process sufficiently encapsulates the attributes of this problem and the remainder of this introduction, and thesis, will concentrate on the passive sonar problem and its related literature. Having said that, it is not necessary to have any prior knowledge of the passive sonar process or the propagation of sound within the underwater environment—the problem will be tackled from a pattern recognition viewpoint and any information from outside this sphere that is necessary in understanding the problem is presented in the latter half of this introduction. Furthermore, existing algorithms that have been applied to the problem of spectrogram track detection will be reviewed in Chapter 2.

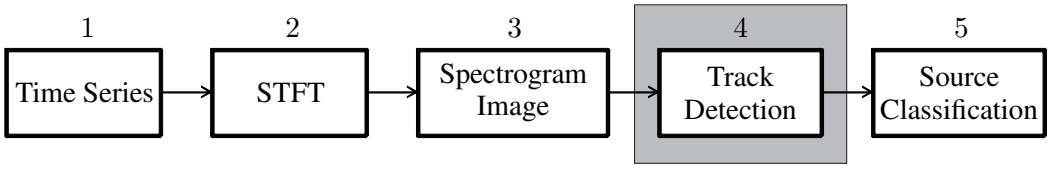


Figure 1.1: Flow diagram of the passive sonar process.

1.1 The Passive Sonar Problem

Passive sonar is a form of sonar in which no energy is emitted from the detection apparatus [178]. Instead, the acoustic pressure surrounding a hydrophone (the transducer) is converted into an electrical signal and analysed to reveal the presence of a source within the environment. Passive sonar is typically used by navies for the identification of submarines, torpedoes and ships and within science and ecology for the monitoring of marine mammals and fish. Currently, trained operators analyse the passive sonar data in spectrogram images to detect and classify any acoustic sources in the surrounding environment [120]. This is a complex task, with many spectrograms being analysed from an increasing number of look-directions, in which the detection of each track is critical to subsequent information processing. Recent advances in mechanical technology, leading to noise reduction, has fuelled the need for more robust, reliable and sensitive algorithms to detect ever quieter engines in real time and in short time frames. Also, recent awareness and care for endangered marine wildlife [125, 172] has resulted in increased data collection, which requires automated algorithms to detect calls and determine local specie population and numbers. Consequently, it is of interest to develop computational algorithms to achieve track detection automatically.

The acoustic data observed via passive sonar systems is conventionally transformed from the time domain into the frequency domain using the short-term Fourier transform [179]. This allows for the construction of a spectrogram image which provides a visual representation of the distribution of acoustic energy across frequencies and over time [174]. The vertical axis of a spectrogram typically represents time, the horizontal axis represents the discrete frequency steps, and the amount of power observed by the hydrophone is represented as the intensity at each time-frequency point. It follows from this that if a source which emits narrowband energy is present during some consecutive time frames a track, or line, will be present within the spectrogram.

The process by which passive sonar exploits narrowband sound radiated in an underwater environment is outlined in Fig. 1.1. Passive sonar systems do not emit any sound and therefore only sound radiated from the target can be detected by the receiver (box 1). The short-term Fourier transform of the observed signal is calculated (box 2) to determine the power present at each frequency band in a particular time sample. These Fourier transforms are then collected together and a spectrogram image is formed (box 3) which represents the energy at each time-frequency point (these points will be discussed further, and illustrated, in the next section).

Sound sources such as ships and other machines radiate some of their energy as narrowband sound that is dependent upon engine speed [174]. The sources of this radiated sound can be grouped under the classes of internal machinery noise and external propeller noise and produce

tracks in a spectrogram that vary in frequency according to the state which the machine is in. For example, when a source is running at a constant speed and there is an absence of the Doppler effect [49], the frequencies emitted are stationary and the narrowband energy that is radiated results in time-invariant tracks. Moreover, a source in which the machinery speed increases, i.e., the source is accelerating, results in tracks that increase in frequency over time. Other sources of radiated narrowband sound that are not dependent on engine speed, the hydrodynamic flow noise and the remainder of the machinery noise, result in constant frequencies regardless of the machine's state. As each type of source emits a particular frequency pattern, it may provide sufficient information for its identification using a spectrogram (Fig. 1.1, box 5). Urick presents a full discussion on the radiation of acoustic energy from submerged machinery in "Principles of Underwater Sound" [174]. Due to the Doppler effect and the nature of the source's machinery the track is often time-variant and therefore, general line detection algorithms, as will be shown in this thesis, are not suitable. It still holds, however, that a particular, relative, frequency pattern will be emitted by each source.

The principle source of complexity in the analysis of passive sonar is that all noise from each concomitant event in the underwater environment is observed. This results in the presence of large amounts of non-uniform background broadband noise in the spectrogram. This noise distorts the tracks, causing them to be broken, particularly at low frequency ranges, and also introduces points of high energy at spurious frequencies. Discriminating these from the signals of interest is particularly hard in low signal-to-noise ratio conditions. Another cause for broken tracks in the spectrogram is the Lloyd mirror, or image-interference, effect [174]. This occurs when the sea is calm; an interference pattern is created by constructive and destructive interference between the direct and surface-reflected sound.

1.2 Data

Following the discussion of the problem, a detailed description of the type of signals that are under consideration will be presented. Consequently, this provides a basis by which synthetic data can be generated for evaluating algorithms designed to detect such signals.

1.2.1 Signal Generation

A continuous signal $x(t)$, observed by a sensor, is the superposition of a longitudinal sound wave emitted by a source $s(t)$, after propagation through, in this case, the ocean environment $s'(t)$ [174], and background noise $n(t)$ [72], such that

$$x(t) = s'(t) + n(t). \tag{1.1}$$

The detection of the periodic or near-periodic narrowband frequency components of $s'(t)$ through spectrogram analysis is the concern of this thesis. Periodicity is defined such that

$$s(t) = s(t + jP), \qquad \forall j \in \mathbb{N}, \tag{1.2}$$

where P is the period of the signal, and near-periodicity such that

$$|s(t) - s(t+P)| < \varepsilon \tag{1.3}$$

where ε is a marginal error resulting from a variation in periodicity. The effects of propagation will be discussed in more detail in Section 1.2.2. Throughout this thesis the noise $n(t)$ is assumed to be Gaussian [72, 11].

The signal $x(t)$ is sampled at a period of T_s seconds (a sampling rate of $f_s \triangleq 1/T_s$ Hz) using the Dirac comb [47] defined by

$$\Delta_{T_s}(t) \triangleq \sum_{m=-\infty}^{\infty} \delta(t + mT_s)$$

where δ is the Dirac delta, to form a discrete signal $x_s(t)$, such that

$$x_s(t) = x(t)\Delta_{T_s}(t). \tag{1.4}$$

The period T_s (or sampling rate f_s) is chosen according to the Nyquist sampling theorem such that the highest meaningful frequency in the application is representable.

This thesis concentrates on the detection of narrowband mechanical sources such as torpedoes, ships and submarines within the ocean. Being mechanical devices, powered by an engine and propelled by a propeller blades, the sound waves emitted are periodic [174]. As such $s(t)$, which is the superposition of a set of harmonically related sinusoids, comprises a *fundamental* frequency, ω_0^t, being the lowest frequency sinusoidal in the sum, and h harmonics of this [11], such that

$$s(t) = \mu + \sum_{k=1}^{h} A_k \sin(k\omega_0^t t + \phi) \tag{1.5}$$

where ω_0^t is the fundamental frequency at time t and, ϕ, its phase, h is the number of harmonics observed, μ is the mean value, and A_k is the amplitude of the kth harmonic. These harmonics are directly related to the rotational speed of the drive shaft.

Several other components of a mechanical device cause the emission of frequency components which are related to this fundamental frequency but which are not harmonics, i.e. they are not integer multiples of the fundamental frequency, and these are referred to as inter-harmonics [115]. Reduction gear ratios connecting the propeller blades, the propeller blades themselves and the power plant emit additional low frequency inter-harmonic components [174]. Auxiliary units such as pumps, generators, servos, and relays also emit noise in the ultrasonic region [139]. These, the fundamental, harmonic and inter-harmonic, frequency components comprise the signature of a particular mechanical device [174]. The signature, due to the differences in the mechanical construction and components, is unique for each type of device and will be referred to as the pattern set, $\boldsymbol{P}_s$, such that

$$\boldsymbol{P}_s = \{m_1, \ldots, m_h\}$$

where $m_1 = 1$ and the term $h \geq 1$ is the number of relative frequency components (the first component of the set corresponds to the fundamental frequency) of the signal $s(t)$.

The signal $s(t)$ can now be defined to be the superposition of sinusoids having harmonically related frequency components defined in $\boldsymbol{P}_s$, such that

$$s(t) = \mu + \sum_{m_k \in \boldsymbol{P}_s} A_k \sin(m_k \omega_0^t t + \phi) \tag{1.6}$$

where $m_k \in \boldsymbol{P}_s$ is the kth relative frequency component of $\boldsymbol{P}_s$ and A_k is its amplitude.

1.2.2 Signal Propagation

Physical phenomena may influence the signal so that the observed signal has different properties from that which is emitted by the source. The passive sonar equation [173]

$$SL - TL = NL - DI + DT \tag{1.7}$$

describes the effects of the oceanic environment upon the intensity of the signal and the conditions upon which it is detectable against background noise. It has three fundamental parts, which are all expressed in decibels (dB): the observed signal intensity, the noise level NL, and the system's detection threshold DT. The observed signal intensity is the difference between the radiated signal level SL, in decibels, and the transmission loss TL, due to the signal's propagation through the ocean. This occurs due to a combination of the following physical effects: spreading, ray path bending, absorption, reflection, and scattering. Therefore, the intensity level of the signal arriving at the sensor is described by the left side of Eq. (1.7), that is $SL - TL$. In addition to receiving the source signal the passive SONAR sensor also receives ambient noise NL. To some extent this can be counterbalanced by the gain of the receiver array DI [174], resulting in an overall noise level of $NL - DI$. When the equality in Eq. (1.7) holds the target is on the system's detection threshold i.e. "a binary choice detector will dither between 'target present' and 'target absent' indications" [171].

The difference between the intensity of the observed source signal $s'(t)$ and that emitted by the source $s(t)$, Eq. (1.1), can be expressed as a scaling of the emitted signal [189], such that

$$s'(t) = \alpha s(t) \tag{1.8}$$

where α is the scaling factor, that is $\alpha \propto SL - TL$, and represents propagation loss.

In addition to this, when a source is performing a circling manoeuvre offset from the receiver, is approaching the sensor, or is receding from the sensor, the Doppler effect [49] causes the emitted sound wave to compress or expand and therefore the perceived frequency $\hat{\omega}_0^t$, may differ from that at the source ω_0^t [66], such that

$$\hat{\omega}_0^t = (\frac{c}{c \pm v_s})\omega_0^t \tag{1.9}$$

where c is the speed of sound through the medium, and v_s is the source to receiver velocity radial component (in the case that both source and receiver are in motion). This equation is dependent

upon the speed of sound in seawater and in 1981 a simplified, nine-term equation for calculating this speed, c (ms^{-1}), was developed by Mackenzie [119], such that

$$\begin{aligned} c &= 1448.96 + 4.591T - 5.304\times10^{-2}T^2 + 2.374\times10^{-4}T^3 + \\ &\quad 1.340(S-35) + 1.630\times10^{-2}D + 1.675\times10^{-7}D^2 - \\ &\quad 1.025\times10^{-2}T(S-35) - 7.139\times10^{-13}TD^3 \end{aligned} \tag{1.10}$$

where T is the temperature in degrees Celsius, S is the salinity in parts per thousand, and D is the depth in meters. Its ranges of validity are: temperature -2 to 30°C, salinity 30 to 40‰, and depth 0 to 8,000 m. Nevertheless, if these conditions are unknown, or an approximate value is sufficient, c can be assumed to be 1,500 ms^{-1} [139]. Other, more complicated, equations exist and are accurate over a wider range of conditions [53, 62], including the international standard (UNESCO) algorithm [39, 186].

Taking the effect of amplitude scaling, by a factor of α, and the changes in perceived frequency $\hat{\omega}_0^t$ described by the Doppler effect into account, Eq. 1.6, which previously described the observed signal $s'(t)$, can be re-written such that

$$s'(t) = \mu + \alpha \sum_{m_k \in \boldsymbol{P}_s} A_k \sin(m_k \hat{\omega}_0^t t + \phi). \tag{1.11}$$

Using these properties, synthetic acoustic signals can be generated which mimic the behaviour of a mechanical device operating in various states.

1.2.3 Spectrogram Formation

A spectrogram $\boldsymbol{S}$ is formed by splitting a discrete time-domain signal $x_s(t)$ into sections τ seconds in length [101], such that

$$x_s^m(t) \triangleq x_s(t + mR), \qquad t = 0, 1, \ldots, T-1$$

where x_s^m is the mth frame of the signal, $T = \lfloor \tau f_s \rfloor$ is the frame length (f_s is the sample rate used when sampling the continuous signal in Eq. 1.4) and $T \geq 1$, and R is the time advance from one frame to the next (in number of samples). Throughout this thesis τ is taken to be one second and R is taken to be $R = T/2$, so that there is a half second overlap between each frame.

The power spectrum of a frame can be calculated using the Short-Term Fourier Transform (STFT) [160], such that

$$F_m(\omega) = \sum_{t=0}^{T-1} x_s^m(t) w(t) e^{-2\pi i \omega t}, \qquad 0 < \omega < \frac{2}{T} \tag{1.12}$$

where $\omega \in \mathbb{R}^+$ represents ordinary frequency (Hz) and $w(t)$ is a window such as the Hamming

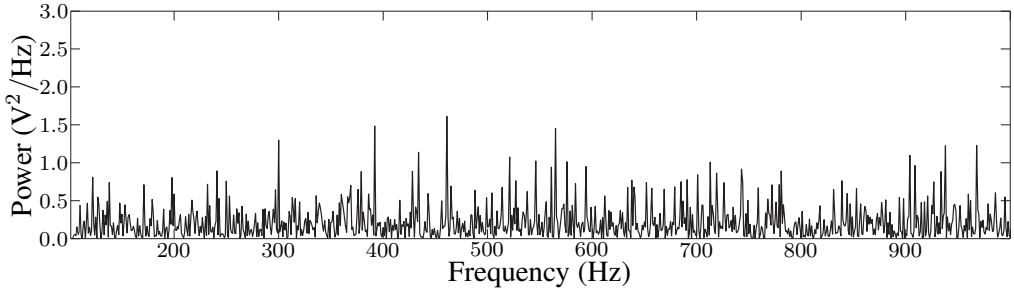


Figure 1.2: Magnitude Squared of the Fourier transform of an acoustic signal at one time frame. The x-axis represents frequency (Hz) and the y-axis power (V^2/Hz). The signal has frequency components of 120, 240, 360, 480 and 600 Hz plus noise derived from a Gaussian distribution (with mean SNR of 3 dB).

window function [76], such that

$$w(t) = 0.53836 - 0.46164 \cos\left(\frac{2\pi t}{T-1}\right). \tag{1.13}$$

The use of windows such as the Hamming window reduces the effects of 'spectral-leakage' [76], which occurs when processing finite-duration signals, by weighting the signal at the frame boundaries close to zero.

The STFT results in the magnitude and phase over frequency of the signal. By taking its squared magnitude and multiplying by a normalisation factor, the periodogram estimate of the power spectrum is derived which satisfies Parseval's theorem [146], according to

$$P_m(\omega) = \frac{1}{\sum_{t=0}^{T-1} |w(t)|^2} \left|F_m(\omega)\right|^2. \tag{1.14}$$

An example of the power spectrum of one time frame of a signal is presented in Fig. 1.2. It can be observed that, at low SNRs, the components of the frequency-set indicated are indistinguishable from the noise. As such, the detection of low SNR frequency components is difficult in single time frame STFTs. Nevertheless, over time, noise is uncorrelated and therefore has a relatively large variance, however, a signal that contains a frequency component is correlated and therefore has less variance; under these assumptions the detection of the frequency components should be easier within a number of successive power spectra.

Treating the power spectrum of a frame, $[P_m(\omega_0)\ P_m(\omega_1)\ \ldots\ P_m(\omega_{N-1})]$, as a row vector, successive vectors can be stacked up and interpreted as a grey scale image $\boldsymbol{S}$, a spectrogram, which has M rows and N columns, such that

$$\boldsymbol{S} = [s_{ij}]_{M\times N} = \begin{bmatrix} P_0(\omega_0) & P_0(\omega_1) & \ldots & P_0(\omega_{N-1}) \\ P_1(\omega_0) & P_1(\omega_1) & \ldots & P_1(\omega_{N-1}) \\ P_2(\omega_0) & P_2(\omega_1) & \ldots & P_2(\omega_{N-1}) \\ \vdots & \vdots & \ddots & \vdots \\ P_{M-1}(\omega_0) & P_{M-1}(\omega_1) & \ldots & P_{M-1}(\omega_{N-1}) \end{bmatrix} \tag{1.15}$$

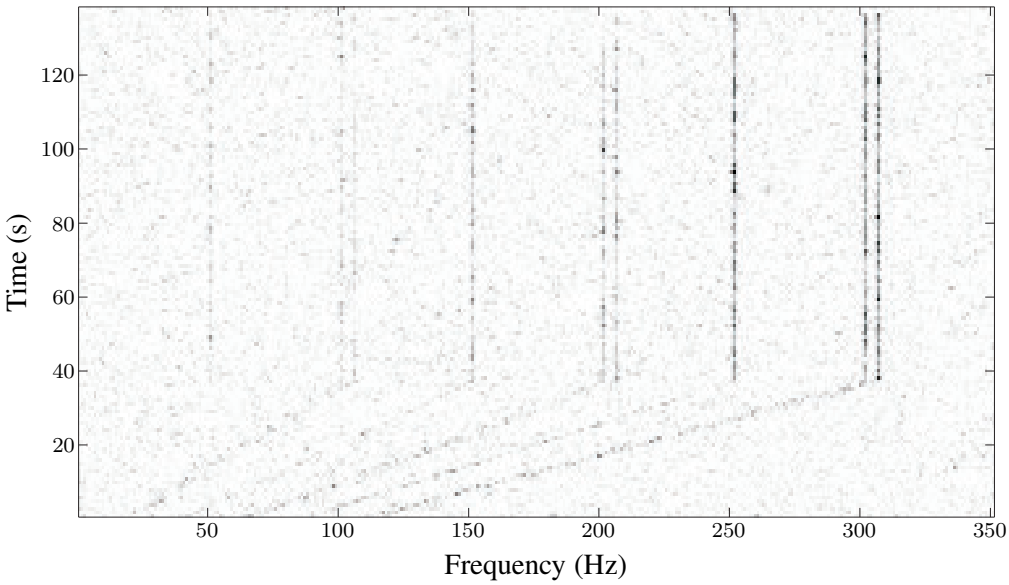
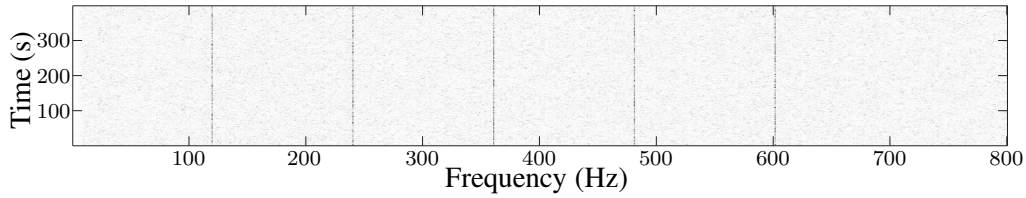


Figure 1.3: A spectrogram image where intensity represents signal power (voltage-squared per unit bandwidth, that is V^2/Hz). In this example the tracks have an SNR of (from left to right): three 3 dB, three 6 dB, and three 9 dB.

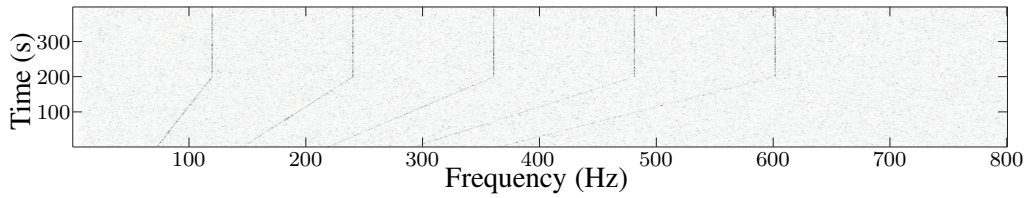
where $i = 0, 1, \ldots, M-1$ is the time frame, $j = 0, 1, \ldots, N-1$ is the frequency bin, $N \in \mathbb{N}$ is the number of frequency bins calculated using the STFT, and $M \in \mathbb{N}$ is the number of previous frames to be retained. Therefore, the grey scale intensity in a spectrogram represents the amount of energy present in each frequency component at a particular time frame. An example of a spectrogram image, the composition of ($M = 40$) power spectra can be seen in Fig. 1.3. As each new power spectrum becomes available it is prepended onto the first row of the spectrogram and the oldest spectrum is removed, forming a "rolling window", also known as a "waterfall display".

A frequency component of $x(t)$, which is constant or varying slowly over time, and is therefore present in more than one consecutive row of $\boldsymbol{S}$, is referred to as a *track*. A track appears in a spectrogram as a (perceptually) connected non-linear structure that can vary in its frequency position in each time frame according to the state of the underlying mechanism. Several states have been mentioned with regards to the domain signals: constant, increasing, sinusoidal and random. For example, a mechanical source that is constantly approaching then receding from the receiver will emit a frequency component that undulates around a central frequency due to the Doppler effect. Within a spectrogram this is represented as a track that is sinusoidal in appearance. Three examples of synthetic spectrogram images which represent a number of track appearances are presented in Fig. 1.4.

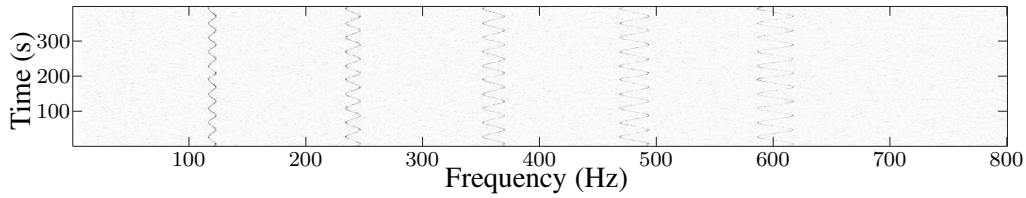
As discussed previously, each of the components of $\boldsymbol{P}_s$ will form a track in the spectrogram at a position relative to the fundamental frequency. For example an acoustic signal may contain fundamental frequencies and their harmonics and inter-harmonics at relative positions to them, in spectroscopy analysis molecules with particular spectral characteristics could form the pattern or electromagnetic signatures that correspond to an object through relative frequencies against background radiation.



(a) A source which is at a constant speed.



(b) A source which accelerates and then reaches a constant speed.



(c) A source that repeatedly approaches and recedes from the receiver.

Figure 1.4: Three examples of synthetic spectrogram images which exhibit a variety of track appearances at an mean SNR of 16 dB. Intensity is proportional to power in voltage-squared per unit bandwidth, that is V^2/Hz.

Within this thesis the mean, frequency domain, signal-to-noise ratio of a spectrogram is calculated such that [72]

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\bar{P}_t}{\bar{P}_b}\right) \tag{1.16}$$

$$\bar{P}_t = \frac{1}{|P_t|}\sum_{(i,j)\in P_t} s_{ij}, \qquad \bar{P}_b = \frac{1}{|P_b|}\sum_{(i,j)\in P_b} s_{ij} \tag{1.17}$$

where $P_t = \{(i,j)|s_{ij} \text{ belongs to a track}\}$ is the set of points related to the frequency components of $s'(t)$ such that $P_t \neq \emptyset$ and $P_b = \{(i,j)|(i,j) \notin P_t\}$ is the set of points which represent noise such that $P_b \neq \emptyset$.

There are two specific approaches to measuring the SNR in this problem and it is necessary to make the distinction: in the time domain (also known as the broadband SNR) or in the frequency domain. As this thesis is concerned with the detection of tracks within a spectrogram image the time domain SNR is not a true representation of the problem complexity, and therefore, all SNRs presented in this thesis are taken within the frequency domain according to Eq. 1.16. As an example of the difference between the two measurements; a time domain SNR of -27.01 dB equates to a frequency domain SNR of 2.99 dB when a sample rate of 2 kHz is used and assuming a 1 Hz bin size STFT.

1.3 Thesis Contributions

Thesis proposition: to demonstrate that a multiple active contour framework is effective at detecting patterns of tracks in spectrograms.

The work is initiated with a full review of the algorithms that have been applied to the problem; this forms the first key contribution of this thesis. The review reveals that two areas have drawn the majority of interest, statistical models, such as the hidden Markov model [150], and image processing/pattern recognition. It is also concluded that, although there has been a great expansion of the areas of pattern recognition and image processing in recent years, there has been relatively little research on applying these advances to the passive sonar domain. Additionally, many of the machine learning techniques that are commonly known in the area of pattern recognition, and that may offer improvements over techniques already applied to the problem of spectrogram track detection have not been evaluated. The active contour algorithm is found to encompass many of the features that have been proposed for use in the detection of spectrogram tracks and to overcome some of the limitations of existing algorithms.

This motivates the next stage of research, and consequently the thesis' second contribution: an investigation into, and evaluation of, low-level pattern recognition and image processing techniques applied to the spectrogram track detection problem. This investigation involves the definition and evaluation of an exhaustive greedy detection method based on multi-scale template correlation to demonstrate an 'optimal' detector's performance. This is the thesis' third contribution as it establishes a benchmark result, which is obtainable using all the information available to detect low-level features. This feature detector is empirically compared with other 'optimal' detectors that utilise less information, and also to feature detectors which utilise dimensionality reduction to simplify the detection process. One of which employs an equivalent data model to the 'optimal' detector and this comparison demonstrates that dimensionality reduction degrades detection performance. All of these low-level feature detectors are evaluated by calculating their Receiver Operating Characteristic (ROC) curves on a set of spectrograms, which contain a variety of SNRs and track appearances. It is shown that none of the standard feature detection methods reach the performance of the exhaustive detector. Nevertheless, near 'optimal' performance can be gained by using machine learning techniques to extract filters from training data and fitting a statistical model to classify unseen examples—simplifying the detector's search space.

The findings and conclusions of this research motivate the development of a high-level track detection framework using an active contour model. This incorporates an interchangeable low-level feature detector into a single and multiple track detection algorithm—the thesis' fourth contribution. The framework provides a flexible detection mechanism that allows for the detection of tracks that have unknown appearances. Furthermore, this framework enables the enhancement of detection probabilities by integrating information taken from either harmonically related positions in the spectrogram or from positions defined by the signature of a specific source. This is a further contribution of this thesis. The framework is evaluated upon a set of synthetic spectrogram images, the properties of which have been outlined in Section 1.2. Testing upon synthetic spectrograms also allows the automatic calculation of ground truth data, which would be hard to obtain

for real-world data, allowing for accurate evaluations to be conducted. The measure used to evaluate the track detection framework is the line location accuracy score [145], which has previously been used by Di Martino and Tabbone [57] for evaluating algorithms applied to this problem. It is shown through a number of empirical comparisons that the solutions presented in this thesis are necessary for the application of the active contour algorithm to this problem. Moreover, the proposed active contour algorithm encompasses aspects of existing approaches, whilst overcoming some of their limitations, such as: high computational complexity, sensitivity to noise, and assumptions of track structure, to name but a few. Ultimately, the algorithm is demonstrated to be an effective method for the detection of tracks that display a variety structures.

1.4 Thesis Structure

The remainder of this thesis is organised as follows. In Chapter 2 a taxonomy, evaluation and review of the spectrogram track detection algorithms found in the literature are presented. The evaluation criteria are defined and example applications are presented along with the criteria which should be met to allow for the successful application of an algorithm. Due to the complexity of quantitatively evaluating each algorithm upon a common data set, the methods are qualitatively evaluated based upon results and algorithm descriptions presented in the respective papers. Chapter 3 presents an investigation into existing and novel low-level feature detection algorithms from the areas of pattern recognition and image analysis. Also, an investigation into the detection of features in harmonically related positions is presented with the aim of enhancing feature detection in low SNR conditions. Chapter 4 proposes a high-level track detection framework for single and multiple tracks which integrates the findings of the previous chapters into the active contour model. The chapter also contains an analysis of the computational complexity of the model. In Chapter 5 the proposed track detection framework is evaluated and a discussion of its effectiveness is presented. Finally, in Chapter 6 the conclusions resulting from the research presented in this thesis are drawn and future research directions are put forward.

Chapter 2

The Field as it Stands

This chapter presents a review of the spectrogram track detection algorithms present in the literature. Constructing such a review reveals the approaches that have been taken to solve this problem whilst ascertaining their limitations, strengths and weaknesses—laying the foundations for future innovations within the field. The research surveyed here is taken from a variety of computer science disciplines and is concerned with the specific problem of track detection within spectrogram images applied to passive sonar. Whilst there is a huge amount of literature on acoustic analysis and pattern recognition the intersection of these fields is relatively small—this chapter provides a review of this intersection. The algorithms are grouped within a taxonomy and evaluated according to the following factors, some or all of which are essential for a successful application: their ability to cope with noise variation over time; high variability in track shape; closely separated tracks; multiple tracks; the birth/death of tracks; low signal-to-noise ratios; their ability to perform track association; that they have no *a priori* assumption of track shape; and, for real time implementations, that they are computationally inexpensive. This evaluation is based on what is presented in the literature.

The chapter starts by defining the evaluation criteria. A taxonomy of the reviewed algorithms is presented and these algorithms are surveyed and reviewed. This leads to a discussion of their principal shortfalls with respect to the criteria defined, and to the identification of issues to be addressed in future research. Finally, the chapter's summary is drawn.

2.1 Definition of Evaluation Criteria

The criteria by which the algorithms will be evaluated, some or all of which are essential for a successful application, are defined below (in no particular order):

C1 Low SNR — Is reliable detection achieved in a frequency domain SNR below 3 dB, defined as Eq. (1.16)?

C2 Temporal Noise Variability — Does the method allow for a time-variant noise model?

C3 Birth/Death of Tracks — Does the algorithm cope with the initiation and/or termination of tracks at some point within the spectrogram?

Application	Typical Track Characteristics	Criteria Required
Whale vocalisation	Short duration, high variability, predictable appearance, initiation and termination observed.	C2 Temporal Noise Variability, C3 Birth/Death Tracks, C4 Multiple Tracks, C7 High Track Variability.
Passive Sonar	Long duration, low SNR, initiation and termination observed.	C1 Low SNR, C2 Temporal Noise Variability,
-Submarine	Low variability.	C3 Birth/Death Tracks, C4 Multiple Tracks, C5 Closely Spaced Tracks, C6 Crossing Tracks, C7 High Track Variability,
-Torpedo	High variability.	C8 No *A Priori* Shape Assumption.
Directly instrumented vibration analysis	Long duration, high SNR.	C4 Multiple Tracks, C5 Closely Spaced Tracks, C6 Crossing Tracks, C7 High Track Variability, C8 No *a priori* Shape Assumption.

Table 2.1: Track characteristics and criteria specific to typical applications of spectrogram track detection algorithms.

C4 Multiple Tracks — Can the algorithm detect two or more separate tracks that exist concurrently (in the same time frame)?

C5 Closely Spaced Tracks — Can the algorithm distinguish two or more tracks that are separated by one frequency bin?

C6 Crossing Tracks — Will the algorithm detect and distinguish between multiple tracks that occupy the same point in a spectrogram for one or more consecutive time frames?

C7 High Track Variability — Does the algorithm detect time-invariant tracks that have high variability?

C8 No *A Priori* Shape Assumption — Is the method free from the assumption of a strict track shape model and therefore can generalise to unknown cases?

C9 Track Association — Does the method output a series of points that it deems as belonging to the same track?

C10 Computationally Inexpensive — Does the algorithm have an on-line computational burden with less than polynomial complexity (not including any training requirements)?

The importance of each criterion depends upon the algorithm's application, as each application is concerned with the detection of signals with different characteristics. The dominant signal characteristics of some example applications, along with the criteria that should be met to demonstrate an algorithm's suitability, are identified in Table 2.1. In addition to these, the need to fulfil the C9 (Track Association) criterion is dependent upon the type of subsequent processing that will be performed and when on-line detection is needed the C10 (Computationally Inexpensive) criterion should be met.

2.2 Algorithm Taxonomy

Algorithms presented in the literature are identified and categorised in Table 2.2 (in chronological order within subheadings). It should be noted that the majority of research has been conducted in the areas of statistical modelling, image processing and neural networks, with additional contributions from relaxation techniques. Hidden Markov models have attracted, by far, the largest proportion of research interest. Considering the relative size, breadth of techniques and the recent speed of progress in the areas of image processing and pattern recognition they have received very little attention in the literature.

It should be noted for completeness that additional methods exist, particularly those that are presented in the literature as Master's theses [197, 40], which it was not possible to survey (although they have been included in the taxonomy presented here). Nevertheless, it is believed that similar techniques from different authors have been reviewed and therefore that the key algorithms are still presented in this review.

2.3 Literature Survey

This section presents a review of the methods found in the literature under the categories presented in Table 2.2. The techniques presented here are specifically those found in the literature that have been applied to the problem of spectrogram track detection in passive sonar systems. As such this is not intended to form a full catalogue of general purpose detection or tracking methods as this falls outside the problem domain specified by this thesis.

It was noted in Section 1.2.3 that there are two distinct approaches to measuring the SNR in spectrogram images. In order to convert between the two, full information regarding the short-term Fourier transform process is needed and this is not obtainable for all of the papers reviewed in this survey. Therefore, where time domain signal-to-noise ratios are presented the distinction is noted.

2.3.1 Maximum Likelihood Estimators

Maximum likelihood estimators (MLE) are based upon statistical assumptions regarding the data in question. A statistical test is defined that decides whether a frequency bin contains noise or a track (signal). Maximum likelihood methods make detections on single spectrogram points and lend themselves to the detection of temporally invariant tracks as no assumptions are made regarding the temporal evolution of a track. Nevertheless, the simplicity of the detection methods limit their application to high SNR cases. This limitation is overcome with MLE methods based on convolution, which make assumptions regarding the temporal evolution of a track to augment low SNR detection. The large search space needed to perform real world detections, however, makes them unfeasible.

Rife and Boorstyn [152] state that after the short-term Fourier transform output has been obtained, the frequency bin that has the maximum value is the maximum likelihood estimate of the

Approach	Representative Works
Maximum Likelihood	
MLE	Maximum value [152]
	Correlation [8]
	Multi harmonic [20]
Image Processing Techniques	
Likelihood Ratio Test	Morphological operators [3]
Hough Transform	Graph theoretic tracking & heuristic search Hough transform[a] [30]
Multi-stage Decision	Multi-stage decision cost function optimisation [55]
Steerable Filter	Gap bridging, region locating & multi-stage decision process [56, 57]
Two-Pass Split-Window	Broadband subtraction via estimation [38]
Edge Detector	Gaussian filtered spectrogram [69]
Neural Networks	
Supervised Learning	Autoassociative memory & multi-layer perceptron [99]
	Multi-layer perceptron [114]
	Multi-layer perceptron constrained using Ockham's networks [98]
	MNET1 [4]
	MNET2 [4]
	RNET [4]
Unsupervised Learning	Kohonen self-organising map [54]
Statistical Models	
Dynamic Programming	Logarithmic likelihood function [162]
Hidden Markov Model	Viterbi & max amplitude [169]
	Viterbi, "mixed" track & threshold [190]
	Viterbi & "mixed" track [191]
	Viterbi & double threshold [165]
	Viterbi & probabilistic data association [88]
	Parallel, multi model detection [175]
	Forward-backward linking, SNR estimate & track gradient [137]
	Forward-backward linking & SNR estimate [138]
	Viterbi & SNR estimate [138]
	Forward-backward linking & spectrum interpolation [74]
Tracking Algorithms	
Particle Filter	Formant detection [163]
Relaxation Methods	
Relaxation	Relaxation[a] [197]
Simulated Annealing	Simulated annealing[a] [40]
	Simulated annealing [112]
Expert Systems	
Double detection	Double threshold & priority ranking [117]

[a] Master's theses which are not surveyed in Section 2.3.

Table 2.2: Categorisation of spectrogram track detection techniques in chronological order within subheadings.

frequency position in the observation, $\hat{\omega}_i$, that is,

$$\hat{\omega}_j = \arg\max_i |s_{ji}|, \qquad j = 0, 1, \ldots, M-1. \tag{2.1}$$

This is repeated for each observation. Thus, a single frequency is detected within each and every time frame j, and the estimated track is a series of these frequency positions. Ferguson [66] has applied this method to the analysis of aircraft acoustics received by an underwater hydrophone.

According to Barrett and McMahon [20], the single frequency case described above, Eq. (2.1), can be extended to the detection of a single frequency that exhibits harmonics, such that

$$\hat{\omega}_j = \arg\max_i \sum_{l=1}^{m} |s_{j,li}|^2, \qquad j = 0, 1, \ldots, M-1. \tag{2.2}$$

These early MLE techniques disregard information describing the distribution of the intensity values attributed to each class, opting to use the maximum instead. This would lead to the method mistaking spurious high power noise for instances of a track. Nevertheless, an important introduction in the multi-harmonic case is the concept of detecting a fundamental frequency by integrating information from its harmonics. This integration of information should greatly increase the detectability of tracks at low SNRs.

Altes [8] presents a likelihood ratio test based upon the correlation of a spectrogram with an expected, noise free, reference spectrogram $\boldsymbol{Z}_k = [z_{ji}(\rho_k)]$, such that

$$p(\boldsymbol{S}|\boldsymbol{Z}_k) \approx \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} \left[\frac{-z_{ji}(\rho_k)}{\sigma^2} + \frac{s_{ji} z_{ji}(\rho_k)}{\sigma^4} \right] \tag{2.3}$$

where σ is the standard deviation of the time domain noise, which is assumed to be known *a priori*. This process is repeated for K reference signal hypotheses (each with a hypothesised signal parameter of ρ_k) and the maximum response is taken to be the detected signal, such that

$$\hat{k} = \arg\max_{1 \leq k \leq K} [\ln p(\boldsymbol{S}|\boldsymbol{Z}_k)].$$

The use of the correlation function allows for the detection of very weak SNR tracks. Nevertheless, for the method's use in remote sensing applications, where the state and behaviour of the phenomenon under observation are unknown, a very large reference set is needed. For example, performing a full search for instances of the sinusoidal track model outlined in Section 3.3.1, which has five free parameters (the additional parameters are the frequency position and phase of the sinusoidal track), would result in a search complexity of $\mathrm{O}(n^5)$ and this complexity grows exponentially with each additional parameter.

2.3.2 Image Processing

Image analysis techniques [71] applied to this area treat the spectrogram as an image containing features to be extracted, applying statistical and image processing algorithms to achieve this.

Image analysis is a vast research area, and provides a wide range of techniques that could be beneficial to this problem. These are often inspired by human visual perception models, which suggests they might be applicable to this problem, as it is accomplished by human operators. The complexity of more advanced methods, however, often makes real-time implementation difficult.

2.3.2.1 Two-Pass Split-Window

Chen et al. [38] propose the use of the two-pass split-window (TPSW) to estimate the background broadband noise within a spectrogram. Once an estimate of this has been calculated, subtracting it from the image should result in a cleaned spectrogram containing narrowband tracks. The TPSW algorithm consists of two steps: first a local mean is calculated over a neighbourhood surrounding each bin in the STFT, such that

$$\hat{s}_{ji} = \frac{1}{2W+1} \sum_{l=i-W}^{i+W} s_{jl}, \qquad i = W, \ldots, N-1-W \tag{2.4}$$

where $j = 0, 1, \ldots, M-1$ and $2W+1$ is the number of bins used to calculate the local mean. The result, $\hat{s}_{ji}$, is clipped and a second, local, mean is calculated upon these (as defined by Eq. (2.4)).

Although this is a filtering technique, a threshold criterion can be defined upon the TPSW output and a detection made using this. As with any filtering technique, there is a balance to be made between the amount of smoothing and the detectability at low SNRs. In this case, this is controlled with the window size W. As the TPSW is calculated independently for each time step in the spectrogram it has no assumption of track structure. This allows the detection of time-invariant tracks that may be highly irregular in appearance.

2.3.2.2 Edge Detection

Gillespie [69], proposes an edge detection method that initially smoothes the spectrogram using a Gaussian filter $\boldsymbol{G}$, such that

$$\boldsymbol{S}' = \boldsymbol{S} * \boldsymbol{G} \tag{2.5}$$

$$\boldsymbol{G} = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \tag{2.6}$$

The benefit of smoothing is that it prevents edges from breaking up into many parts; the detrimental effect is a reduction of the spectrogram's resolution if the smoothing kernel is too large.

Each point (i, j) in the smoothed spectrogram $\boldsymbol{S}'$ is thresholded by comparison to the background measurement b_{ji}. This background measurement is continuously updated to allow for time-invariant noise conditions and computed independently for each frequency bin, such that

$$b_{ji} = b_{j,i-1} + \left(\frac{s'_{ji} - b_{j,i-1}}{\alpha} \right) \tag{2.7}$$

and the spectrogram is thresholded according to

$$\frac{s'_{ji}}{b_{j,i-1}} > H \tag{2.8}$$

where H is the threshold value. Furthermore, detections in subsequent time frames are linked if they are within adjacent or overlapping frequency positions.

This method is applied by Gillespie to whale call detections and of the 2,077 calls detected by humans the method successfully detected 1,897 (90%). Nevertheless, as with all methods that rely on smoothing of the spectrogram, the detection of low SNR tracks can be compromised as they may be eliminated in the transformed image.

2.3.2.3 Likelihood Ratio Test

Abel et al. [3] propose a statistical likelihood test to be used for track detection. The probability distribution of a signal (assumed to be Gaussian) is determined along with the distribution of noise probabilities. A likelihood test is defined such that

$$\frac{r_{ji}}{r_{ji}+1} \cdot \frac{s_{ji}}{b_{ji}} \underset{H_B}{\overset{H_N}{\gtrless}} T_\lambda \tag{2.9}$$

where r_{ji} is the SNR at point (i, j) and b_{ji} is the broadband power at point (i, j), and H_N and H_B are the hypotheses of a pixel containing narrowband and broadband signal, respectively. The result of applying this test is fragmented tracks and isolated false positive detections. These inconsistencies are repaired using the morphological operators dilation and erosion, which expand and contract a track respectively. In set theory, erosion is defined as $A \ominus B = \{z \in E | B_z \subseteq A\}$ where E is a Euclidean space or an integer grid, $A = \{(i, j) | s_{ji}$ belongs to a track$\}$ in E, B is a structuring element and B_z is the translation of B by vector z. Informally, erosion means to translate the structuring element B to all points in A and take only the points where the structuring element overlaps completely with points in A. Dilation is defined as $A \oplus B = \{z \in E | (\hat{B})_z \cap A \neq \emptyset\}$ where $\hat{B}$ is the symmetric of B. Informally, this means to translate the structuring element to every point in A and take all the points that are covered by the structuring element. Combined and ordered in this way produces 'closing', $A \cdot B = (A \oplus B) \ominus B$, [71] which has the effect of smoothing, eliminating thin protrusions and filling narrow gaps in the tracks. After this process, the region grow algorithm is employed to group pixels into a single track. This algorithm recursively groups connected pixels based upon a similarity measure, which, in this case, is that the pixels are part of a track.

The likelihood ratio test is described as being optimal as, for a given probability of a false alarm, the probability of detection is maximised. The background noise is not assumed to be stationary and therefore broadband equalisation is used to estimate r_{ji} on a frame-by-frame basis by taking the trimmed mean over a sliding frequency window [2]. Over-smoothing, however, may reduce its applicability to the detection of low SNR tracks. This method also requires the use of a threshold that must be determined *a priori*, further limiting its generalisation. In the noisy test

image presented in the paper [3] the method appears to cancel a large amount of the background noise whilst preserving the track. No quantitative results are presented however. Additionally, the use of the erosion operator limits this method to approximately stationary tracks because of its assumptions about track shape. Sections of tracks which do not fit the operator B exactly, i.e. tracks that rapidly increase/decrease in frequency, will be eliminated from the resulting detection.

2.3.2.4 Multi-Stage Decision Process

Di Martino et al. [55] present an alternative approach based on feature grouping theory. In the paper it is stated that several studies in feature grouping theory [102, 80] suggest that it is possible to find perceptual features of patterns that allow efficient figure-ground discrimination. In the case of spectrogram tracks, Di Martino et al. define these features, which distinguish a set of points belonging to a track from those belonging to noise, to be: frequential curvature regularity; temporal continuity; high average intensity; and high point density. A new cost function, which incorporates these perceptual features, is defined over a track ζ, such that

$$\Phi(\zeta) = \frac{\alpha.G(\zeta) + \beta.C(\zeta)}{A(\zeta)}, \tag{2.10}$$

where $A(\zeta)$ is the track's amplitude, such that $A(\zeta) = \sum_{(i,j)\in\zeta} s_{ji}$, and the terms $G(\zeta)$ and $C(\zeta)$ are its continuity and curvature respectively. The cost function will decrease if a spectral track is detected and increase otherwise and, therefore, the problem is transformed into optimising the cost function along all paths of length N, starting from a given image point. Each time an optimal path is found to traverse a point in the image, the point's counter is incremented.

It is claimed by Di Martino et al. [55] that the computation of the optimal path according to the cost function $\Phi(\zeta)$ is linear in N and the algorithm is amenable to parallel processing. The qualitative result presented by Di Martino et al., obtained using one spectrogram, reports that the method reduces the noise and that the spectral track "becomes more perceptible" [55]. It is stated that the method has been tested on a set of spectrograms with differing SNRs, the results of which show that this method increases track detection and decreases false positive detections (although these results are not presented).

A point to be made regarding the continuity measurement used in Eq. (2.10); the measurement is defined to be proportional to the number of track points that have zero amplitude and this is rarely the case in spectrograms that contain background noise—such as those from sea environments. In this case it may be more intuitive to define the measurement to be proportional to the number of track points that are below a defined threshold. Moreover, the use of the track's amplitude in the denominator gives the output a large dependency upon this factor. Thus restricting the detection to relatively high SNR tracks; if the weights are chosen to detect high curvature, high continuity tracks that have high amplitude, tracks that have low curvature, high continuity and low amplitude are likely to be missed. Also, if there are spurious points of high amplitude noise present in the spectrogram, which would have high curvature and low continuity, there is a high probability that these would cause a false positive detection.

2.3.2.5 Steerable Filter

Di Martino and Tabbone [57] propose an approach using steerable filters. Three steps are defined: the detection process, region locating and track tracing. The detection process starts by smoothing the spectrogram using a Gaussian filter and an energy function is defined, such that $E(\theta) = \ddot{G}(\theta)^2 + H(\theta)^2$ (where $\ddot{G}$ is the second derivative of the Gaussian and H is its Hilbert transform in the direction θ), to detect edges using steerable filters. Continuing the detection process, the local dominant direction θ_d, such that $\theta_d = \arg\max_{\theta}[E(\theta)]$, for each point in the spectrogram is found, the second derivative of the image is calculated to enhance tracks and the local maxima in the direction perpendicular to θ_d is found. False contours that result from this process are suppressed using an hysteresis threshold [33] and gap bridging is utilised to provide continuity. The regions surrounding the detected edges are located by computing the zero-crossings of the second derivative on either side of the detections and, to remove the effects of noise on the zero-crossings, the mean distance along the curve to its zero crossings. This determines the region $R_i = \{(i,j)|l_i \leq j \leq r_i\}$, where l_i and r_i are the region's left and right boundaries and i is the row index, that encompasses them. A multi-stage decision process (as described in Section 2.3.2.4) is used to trace the track in the original spectrogram within the regions detected during the processing to extract the spectrogram tracks. This maximises the cost function $\Phi(\mathcal{C})$ defined as

$$\Phi(\mathcal{C}) = \sum_{i=0}^{N-1} A(P_i) - \alpha \sum_{i=2}^{N-1} |l(P_{i-1}, P_i) - l(P_i, P_{i+1})| \tag{2.11}$$

where $P_i \in R_i$, $A(P_i)$ is the amplitude of P_i, and $l(P_i, P_j)$ is the slope of segment $[P_i, P_j]$. This extracts contours present within the regions R_i. The initial stages of this process (region location) are used to refine the search space within which the multi-stage decision process optimises, thus reducing the computational burden.

It is noted that locating the regions in the proposed way does not guarantee that two tracks have not been merged during smoothing and therefore that only a single track is present within the track tracing search region. Also, the proposed method is not truly unsupervised as a threshold parameter value needs to be manually determined within the track detection stage. The method was tested using spectrograms of varying SNRs[a] (1.50–7.45) and varying spatial frequencies [56]. It achieves above 87% detection performance over all SNRs and spatial frequencies and can perform the detection within a 128×128 pixel spectrogram in 36.74 seconds. It is not possible to perform a direct comparison between the SNRs used in this experiment and others as a different SNR measurement is used[a].

The use of the cost function $\Phi(\mathcal{C})$, Eq. (2.11), provides a balance between the detection of temporally invariant tracks and high SNR tracks. The local nature with which the curvature is calculated prevents the method from linking spurious high amplitude noise responses that are some distance away from the current track, whilst allowing globally fluctuating tracks to be detected.

[a]It is assumed that the paper's authors use the same SNR calculation as is presented in their other paper [57] and therefore that these figures are calculated as SNR $= 10\log_{10}([\bar{P}_t - \bar{P}_b]/\sigma_b)$ where σ_b is the standard deviation of the noise.

Nevertheless, in situations in which high amplitude noise points exist within the identified region R_i, there is a high probability that they will cause the detection to deviate from the true location.

2.3.3 Neural Networks

Neural networks are a widely applied classification architecture and a wide variety of neural networks exist, many of which are described in "Neural Networks: A Comprehensive Foundation" by S. Haykin [77] and "Neural Networks for Pattern Recognition" by C. Bishop [28]. Multi-layer non-linear neural networks can be effective as pattern classifiers [180] and have a proven ability to extract salient features of high-dimensional input spaces, allowing the identification of patterns in complex problems [77]. These properties make them a strong candidate for applications such as this. The atomic unit of a neural network is a neuron and a neuron is a simple mathematical model of the neurons that exist in biological nervous systems [121], such that [28]

$$z = g\left(\sum_{i=0}^{n} w_i x_i\right)$$

where z is the neuron's output, g its activation function, w_i is the weight applied to the ith input x_i and n is the number of input synapses. They have been studied by Rosenblatt [153] who referred to them as perceptrons and Widrow and Hoff [183] who called them adalines.

It is the aim of a well designed neural network to learn a statistical model of the process that generates some data. This is achieved by iteratively adjusting the weights of neural connections with the aim of minimising an error function defined upon some training examples [28]. In a supervised learning setting these examples have class labels attributed them and the error is a function of mistaken classifications. Unsupervised learning does not make use of class labels and instead the neural network may determine statistical similarities of the data. A key drawback in the use of an improperly designed neural network, and any model that learns by example, is the possible reduction in the model's ability to generalise to unknown cases. In applications such as this, frequency tracks can vary greatly and it is quite probable that a training set will not fully represent the range of variations that the model may need to identify.

2.3.3.1 Supervised Learning

Di Martino and Tabbone remark that such methods "need a supervised learning set that reduces their utility in real cases" [57]. Kendall et al. investigate this by testing several methods for improving the generalisation of neural networks [98]. In terms of the application this improves the networks' ability to detect track structures that were not included in the training data. Several techniques to improve a network's generalisation ability are tested: heuristically changing the number of hidden nodes, weight decay, soft weight sharing and Ockham's networks.

A hidden node is a neuron within a neural network that is neither an input or output unit. These are described as being hidden because their activations are not directly seen by the outside world. The hidden layer (the layer of the neural network that is made up of hidden units) learns to represent the input data in a way that captures salient information. The number of nodes, or even

the number of hidden layers, determine the network's ability to represent complex, non-linear, patterns [28]. Having too many hidden nodes, however, increases the network's complexity, and can have the side affect of allowing the network to quickly over-fit training data (a problem also referred to as the bias-variance trade-off)—reducing its ability to generalise [28]. Unfortunately, there is no definitive method to determine the number of hidden nodes that are needed to solve a classification problem [60] and so trial and error is often employed.

Weights are applied to the values passed between nodes of the network and control how much effect the value has on the receiving node's activation. Utilising weight decay [28] helps to avoid over-fitting training data by forcing the weights to remain small and can lead to significant improvements in generalisation performance [84]. This is realised through a simple regularisation function utilised during training, which shrinks the weight's value after they have been updated. This function is defined as

$$C = \sum_i \sum_j (o_j - d_{ij})^2 + \lambda \sum_i w_i^2 \tag{2.12}$$

where d_{ij} is the desired value of output o_j in the network's output layer, w_i is the network's i^{th} weight and λ can be thought of as a normalising parameter.

Weight sharing [111] is a technique in which a single weight is shared among several connections in a network, reducing the number of adjustable parameters. This requires good knowledge of the problem background so that it is possible to specify which connections will share weights [111]. Soft Weight Sharing [134] utilises Gaussian mixture models during training to determine the weight's values and which weights should be linked dynamically. This removes the dependence on the user to fix the weighting links *a priori*.

Ockham's razor states that more simple models should be preferred to more complex models and that this preference should be balanced with the extent to which the model fits the data [28]. This philosophy is utilised in Ockham's networks to improve the generalisation performance of neural networks in the absence of large amounts of training data [97]. The minimum description length principle [21] is utilised to attribute a coding length to a network and the classification errors it produces. A cost function is defined such that

$$C = I(x|\Theta) + I(\Theta) \tag{2.13}$$

where $I(x|\Theta)$ is the description length of the data misfit x, given the chosen model Θ (the input/output values of all the training pairs not correctly classified) and $I(\Theta)$ is the description length of the model itself (the neural network's weights). The cost function is optimised by a genetic algorithm [128] and the network that equates to the minimum is optimal—according to Ockam's razor—as it has the least combined classification errors and complexity.

Weight decay and Ockham's networks were found to be the most advantageous methods evaluated by Kendall et al.. Weight decay, constrained by the cost function outlined in Eq. (2.12), was found to significantly reduce the classification variance on a generalisation set when using a network with one hidden node. For a complex network (eight hidden nodes), correct values of

λ not only reduce variance but also provide improvements in the generalisation performance by reducing the network complexity. Ockham's networks, however, were found to be the most successful method for improving generalisation. Nevertheless, it was shown that, in this problem, the method has limited effect and reduces the generalisation error rate by no more than 3% upon a test set containing 121 instances of 9×9 pixel spectrogram windows (which were independently labelled from the training set). Furthermore, the method is very computationally expensive, requiring 24 hr of computation time for one run. Because of this, no averaging over many trials was performed. It is stated, however, that "given that the genetic algorithm is finding a near global minimum for C it is likely that the variance will be small" [98]. Aside from limited improvements in generalisation error, the Ockham's network method did result in the lowest complexity network based on the minimum description length principle.

Khotanzad et al. [99] implement a track detection mechanism with the following steps. Initially the spectrogram is thresholded to obtain a binary image. An auto-associative memory (ASM) [85] is employed to eliminate the noise and to reconstruct the received signal. The ASM is trained using a learning algorithm based on Hebb's rule [79] upon a number of clean reference signals that contain a target or no target, of which the closest to the noisy input signal is recalled during evaluation. The output of the ASM is then passed to a multi-layer perceptron (MLP) [28] trained using the back-propagation algorithm [155] to classify the clean data from the ASM as containing a target or not.

It is stated in the paper that in an initial study a classification accuracy of 97% was achieved for spectrograms that contain a track, and 100% for noise only spectrograms. These results, however, were obtained using a very small test set that was derived by adding Gaussian noise to the training spectrograms and that consisted of 24 spectrograms containing a track and 12 noise only spectrograms. Moreover, the shape of the tracks present in test set were regular and do not vary greatly in appearance. Under these conditions, it is possible that the networks are over-fitting the data, explaining the very high classification rates, and that the technique would not generalise well.

Leeming [114] also investigates the applicability of the MLP, however, in this study its ability to determine the number of tracks present in an example is under scrutiny. The MLP network was trained in two ways; the first, to classify a window as containing 0, 1, 2 or greater than 2 tracks, and, the second, whether the MLP can recall a clean picture with no noise from the input data (a similar problem to that investigated by Khotanzad et al. using the ASM). The evaluation is performed upon a collection of spectrogram windows containing strong time-invariant tracks 10–20 dB above noise, weak time-invariant tracks 4–10 dB above noise and time varying tracks 7–10 dB above noise (having a random frequency variation of ± 1 frequency bin per time frame).

The results conclude that MLPs with one hidden layer do not have the capacity to model data that contains two or more tracks, however, data containing noise or one track could be recognised by a such an MLP. An additional finding is that it is possible to remove noise from windows using a network topology consisting of one hidden layer and increasing the number of nodes improves clarity, especially in the case that the window contains time varying tracks. It is found in the paper that when applying the networks to the task of counting tracks, increasing the size of the second hidden layer produces no increase in effectiveness and it is concluded that this suggests that the

second layer is counting tracks and the first is removing noise (although it is noted that these networks require far fewer nodes in the first hidden layer than those tested for removing noise and therefore this distinction is not clear).

The experimental results presented in the paper demonstrate that this method detects 75% of tracks that are time-invariant within an SNR range of 4–10 dB and 79% of time varying tracks with SNRs ranging between 7–10 dB (when trained to detect the respective track types). To test the generalisation performance, a network trained to detect time-invariant tracks is tested using the time varying test set and *vice versa*. In this case the performance drops to 69% (trained on time-invariant tracks and tested on time varying tracks) and 43% (trained on time varying tracks and tested on time-invariant tracks). These results highlight the difficulty of applying neural networks to this problem, however, it is not clear whether this is a fundamental limitation of the neural network architectures or the effects of over-fitting.

An alternative architectural approach is taken by Adams and Evans who implement MNET—a multi-layer feed-forward NN architecture for track detection inspired by the Hidden Markov Model (HMM) [4] (see Section 2.3.4.2 for a full discussion of HMM techniques). In the context of HMMs, the probability of a particular observation sequence, given a HMM, is calculated using the forward-backward algorithm [150]. A method analogous to this algorithm is used to calculate the output of each node at each time step in the MNET architecture. The estimated sequence of track locations are then obtained by finding the node (each node represents a frequency state) with the largest output at each observation time. In the paper, two networks are derived from this architecture: MNET1, which is trained using a supervised learning algorithm; and MNET2, in which parameters are derived analytically from knowledge of the problem structure (a method used by Streit and Barrett [169] and Xie and Evans [190] to determine HMM parameters). Additionally, the architecture is extended to form RNET, in which the nodes representing the HMM states are replaced by an MLP network, and this is trained using a supervised learning algorithm. The addition of the hidden layers, and the use of the sigmoid activation function within them, creates a non-linear mapping between network's input and output [28].

The tracking problem presented in the paper was simplified by quantising the STFT frequency range into eight possible states and, therefore, the HMM, MNET1, MNET2 and RNET architectures had eight states corresponding to each of these sub-ranges. These architectures were compared against a MLP NN and a HMM using the Viterbi algorithm to track the frequency. The Viterbi algorithm [150] is used to determine the most likely sequence of hidden states defined by the HMM (called the Viterbi path) that results in the observed sequence. It is concluded that the HMM outperforms the other methods in tests where SNRs are between 4 and -5.6 dB and RNET achieves the closest performance to the HMM; followed by MNET1, NN then MNET2. Nevertheless, the operational computational complexity of RNET and both the MNET architectures, $O(NM)$, is lower than that of the HMM, $O(M^2N)$. An advantage of MNET's architecture over the NN is that its number of nodes is tied to the problem formulation and is therefore predefined, whereas the size of a NN needs to be determined by trial and error. Also, compared with the NN, MNET has a smaller network size. This is also true when compared to RNET (which is also smaller than the neural network), however, the addition of RNET's hidden layer creates a

non-linear mapping from input to output, allowing it to model more complex data and achieve a higher detection rate. A limitation of the experimentation is the coarse frequency resolution into which the spectrograms are subdivided; this limits the method's ability to detect tracks that have small frequency variations, however, it results in networks that have fewer states and, therefore, simplifies the detection problem.

2.3.3.2 Unsupervised Learning

Methods using unsupervised learning may show more reliable application to real world cases as they are not trained to detect a specific track structure but learn the statistical similarities between the data [28].

Di Martino et al. [54] propose the use of a two layer adapted Kohonen self-organising map [104] that is constrained according to the same perceptually relevant track features as those outlined in their previous paper [55] (and outlined in section 2.3.2.4). The map, with an input layer of 147 nodes (three nodes for each input pixel that represent time, amplitude and frequency) and an output layer of 49 nodes (N), is applied directly to the spectrogram in an attempt to extract tracks. In using three input nodes for each pixel, each being attributed to a different aspect of the pixel, the defined perceptual track features can be evaluated upon the converged map. Once this convergence occurs within a spectrogram, a cost function, $\Phi(W)$ that incorporates the defined features, is evaluated on the weights of each type of input node (time, amplitude and frequency). This cost function tests the convergence for the presence of a track, such that

$$\Phi(W) = \frac{\sum_{i=1}^{N} W_i^A}{N \sum_{i=2}^{N-1} (W_{i-1}^F - 2W_i^F + W_{i+1}^F)^2} \tag{2.14}$$

where W^F and W^A represent the weights attributed to the connection of the frequency and amplitude input nodes to the output layer respectively.

The method was applied by the authors to a spectrogram with an SNR of $2\,\mathrm{dB}$[b] and the network's detection resolution was taken to be a 7×7 pixel window in a 70×70 pixel spectrogram. The resulting spectrogram has the majority of the noise removed and contains a large response where the track is present in the ground truth data. The track in the original spectrogram is not continuous as noise obscures parts of it, however, the resolution of the self-organising map causes many of these gaps to be bridged. A property that could also result in localisation problems and extend terminated tracks. With regard to the cost function $\Phi(W)$, its formulation allows for the detection of high amplitude, low curvature tracks as its numerator takes a high value and the divisor a low value, equating to a high response. When a high amplitude high curvature track is encountered, however, the function will take on a low value, giving a high probability of false negative detections. This would also be the case for low amplitude low curvature tracks, which is a limitation when low SNR track detection is needed.

[b] calculated as SNR $= 10 \log_{10}([\bar{P}_t - \bar{P}_b]/\sigma_b)$

2.3.4 Statistical Models

Statistical models determine the optimal path through a number of detections, which include false and true positives, by calculating the path with the maximum likelihood depending upon the data and a model of the data's generative process.

2.3.4.1 Dynamic Programming

Scharf and Elliot [162] model a frequency track as a random walk, $z_k = z_{k-1} + \epsilon_k$, and derive a dynamic programming [25, 108] approach for track extraction. Dynamic programming is a method of solving complex tasks by solving the simpler, smaller problems that they comprise. In this case, the state of the frequency track is determined by calculating its state at each row of the spectrogram recursively. The method is described as being applicable to frequency or phase tracking, stating that "the distinction between the two is more imagined than real". A logarithmic likelihood function, l, is defined such that

$$l \sim \frac{1}{2\sigma^2} \sum_{n=0}^{N-1} Re\{e^{(-i\phi_{nk})} P_n(\hat{\omega}_n)\} + \sum_{n=0}^{N-1} \ln p(\hat{\omega}_n | \hat{\omega}_{n-1}) \tag{2.15}$$

where $\hat{\omega}_n$ is the estimated discrete frequency state, $p(\hat{\omega}_n|\hat{\omega}_{n-1})$ is the transition probability, which is chosen to model a notion of physical reality, σ is the standard deviation of the time domain noise and $e^{(-i\phi_{nk})}$ is the phase shift of the STFT, where ϕ_{nk} is the total accumulated phase after nk steps (k is the number of samples in which the phase is assumed to increase at a fixed linear rate). Here σ is fixed and therefore the standard deviation of the noise is assumed to be stationary and known *a priori*. The most likely track is one that maximises l and dynamic programming is used to determine this by calculating the best path through the observed peaks (a more complete discussion of a related non-linear tracking algorithm is presented by Scharf et al. [161]).

The algorithm was tested on two spectrograms with a carrier-to-noise ratio (SNR of a modulated signal) of $-3\,\text{dB}$ (time domain) using 60 time steps of data to calculate the optimal path. They note that even when STFT peaks are unreliable the method tracks the true frequency. Nevertheless, it can be observed in the qualitative data presented that, at several points, the tracking diverges from the true frequency.

2.3.4.2 Hidden Markov Model

Shin and Kil [165] argue that to effectively track a signal any *a priori* knowledge of the signal's behaviour should be used and that Hidden Markov Models allow for this. Hidden Markov Models (HMM) [150] are well known for their application to this type of problem as they allow for the modelling of an unobservable stochastic process that is observed through an additional stochastic process, producing a sequence of observations (in this case the STFT output).

A general limitation of the HMM is the automatic discretisation of an estimated continuous variable [105], in this case the signal's frequency. This does not, however, affect its application to this problem as the continuous frequency is discretised during the STFT and the HMM estimates

the state within these frequency bins. Another limitation associated with HMMs is the automatic determination of the model's parameters given some training data. An approximation to the solution can be achieved using iterative methods such as the Baum-Welch algorithm [22], the Extended Baum-Welch algorithm [94], which are generalised Expectation-Maximisation algorithms, or gradient techniques [151]. Employing such methods can reduce the generalisation ability of the resulting HMM to track variations that are similar to those present in the training data—a typical supervised learning problem called over-fitting. Anderson et al. [11] further discuss issues associated with HMM models.

2.3.4.2.1 Single Track Streit and Barrett [169] demonstrate the use of a HMM spectrogram frequency tracker. In this formulation only the most powerful frequency bin is used in each observation, limiting the method to the detection of single tracks. The inclusion of a zero state allows the tracker to model disappearing and re-appearing tracks, the occurrence of which is detected using a threshold value. Frequency cells composed of a subset, or gate, centred on the previously detected frequency cell (therefore representing the allowed wandering frequency positions) are identified with the states of the hidden Markov chain. Analytic expressions for the basic parameters of the HMM are obtained in terms of physically meaningful quantities. It is shown that the computational complexity of the Viterbi algorithm is $[(n+1)+c_1]^2T$, where c_1 is the complexity (in units equivalent to addition) of computing the measurement probability density function (PDF) (in the case where it is computed for each symbol in the measurement vectors), and the computational complexity of the forward-backward linking algorithm is $[(n+1)+c_2]^2T$, where c_2 is the PDF calculation complexity in units equivalent to multiplication.

The performance of the HMM tracker was qualitatively evaluated for two sets of simulated data and demonstrates good detection results in time domain SNRs of $-20\,\text{dB}$ and $-23\,\text{dB}$ with the disappearance and initiation of tracks. The HMM tracker was compared with the dynamic programming method presented by Scharf and Elliot [162] and it was found that their method is equivalent to an HMM using real valued continuous measurement vectors. Scharf and Elliot, however, do not include a zero state to account for the absence of a signal. It is noted that the dynamic programming algorithm presented for maximising the likelihood function l, Eq. (2.15), is equivalent to the Viterbi algorithm.

Paris and Jauffret [138] and Shin and Kil [165] both investigate the use of HMMs applied to this problem. Both compare forms of the Viterbi line detector (a global optimisation scheme) while Paris and Jauffret also test the forward-backward (F-B) local optimisation algorithm.

Shin and Kil use the smoothed amplitude of the short-term integrator as a feature for the algorithm. Subsequently, a double threshold Viterbi line detector is employed; two thresholds are used to identify which STFT bins are to be linked, reducing the algorithm's computational load. A likelihood function based upon each cell's amplitude and linking distance is used which, as this is based upon amplitude information, allows the algorithm to cope with time varying signal and noise characteristics. Below an SNR of $-4\,\text{dB}$ (time domain) the performance of the Viterbi algorithm is shown to be weak as false detections become apparent. To compensate for this the authors propose to extract features from projection spaces other than the spectrogram image and

employ feature fusion, optimisation and classification techniques (discussion of this is beyond the scope of this chapter). Qualitative results (of the Viterbi detector alone) were presented from one spectrogram image showing that tracks with slow spatial variation are recovered accurately.

Paris and Jauffret propose to integrate SNR estimates into the HMM algorithm to improve tracking performance when the spectrogram SNR is not known *a priori*. Two methods for estimating the SNR of a spectrogram are proposed: a parametric maximum likelihood estimation (MLE), which gives the scaled likelihood, defined as

$$b^s(s_{ji}) \simeq exp\left[\frac{Ns_{ji}}{\sum_{l=0}^{N-1} s_{jl}}\right]; \tag{2.16}$$

and a non-parametric probabilistic integration of the spectral power (PISP) approach by taking the normalised spectrogram, such that

$$\bar{s}_{ji} = \frac{s_{ji}}{\sum_{l=0}^{N-1} s_{jl}}. \tag{2.17}$$

Implementing an SNR estimate in this way slightly reduces the computation time associated with the MLE method. Calculating the likelihood of the current observation in terms of its mean allows for detection even if the noise level varies with time.

It was shown that both the Viterbi and the F-B algorithms perform equally well in the experiments, and that estimating the SNR results in no loss of performance (it is also noted that both SNR estimates perform equally well). It is stated, however, that the Viterbi algorithm performs many more comparisons (but fewer multiplications) than the forward-backward algorithm (reflecting that found by Streit and Barrett [169]) and that PISP is less computationally intensive than MLE. One shortfall of these methods is that they do not take into account the appearance or disappearance of a frequency track or the existence of multiple tracks.

Jauffret and Bouchet [88] outline a probabilistic data association (PDA) method coupled with the Viterbi line extractor. The spectrogram is thresholded resulting in a set of false alarms and a set of true detections. The likelihood of a spectrogram track is calculated to be proportional to

$$L(\boldsymbol{S}_{j*}|y_j) = 1 - P_d + \frac{P_d}{\lambda}\sum_{i=0}^{N-1}\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{(s_{ji}-y_j)^2}{2\sigma^2}}, \qquad j = 0, 1, \ldots, M-1 \tag{2.18}$$

where $\boldsymbol{S}_{j*}$ denotes row j of the spectrogram $\boldsymbol{S}$, σ is the standard deviation of the Gaussian distribution modelling the correct detections, y_j represents the state of the system at time j, P_d is the probability of detection and λ is the probability of false alarm (per frequency cell). Several assumptions regarding the nature of the data are made in this calculation, which are outlined in the paper. The Viterbi line detector is then used to extract the most likely track from the spectrogram.

This method was shown to reliably detect slowly varying tracks when the SNR is above 4 dB, in both simulated and real world examples. Van Cappel and Alinat comment that "probabilistic data association with severely limited branching factors suffers from various difficulties due to the low SNR and to the variability of track frequencies and amplitudes" [175]. The proposed method also does not account for the birth and death of tracks.

Gunes and Erdöl [74] argue that if concentrated noise exists in specific frequency ranges, deriving the observation estimates with respect to the full spectrum, as has been presented thus far, will typically lead to unbalanced observation likelihoods. They outline a HMM for the detection of vortex frequency tracks in low SNR conditions that overcomes this limitation by defining an observation likelihood measure based upon the interpolation between local maxima of the spectrum. The spectral estimate's local maxima are determined within each time frame and form the centres of windows within which interpolation across subsequent time frames is performed. This results in a set of smoothed local maxima, which are used to mask the original spectral estimate during the observation likelihood calculation—thus the calculation is determined with respect to a subset of the spectrum.

Gunes and Erdöl implemented the forward-backward linking algorithm to perform track association. The method was shown to reliably detect tracks within two spectrogram images, one of which exhibits time variant noise irregularly distributed throughout the frequency spectrum and the other Gaussian noise.

2.3.4.2.2 Multiple Tracks Paris and Jauffret demonstrate a HMM scheme that is able to detect multiple simultaneous tracks [137]—an event that occurs when multiple acoustic sources exist in a number of consecutive time frames or when a single source emits multiple frequencies that could be harmonically related, as described in Section 1.2.1. An additional constraint is imposed upon the F-B algorithm, that is, that two tracks cannot inhabit the same place in state space by adding the track's rate of frequency change, $\dot{f}$, to the representation of the state y_i, such that

$$y_i = \frac{1}{\Delta f} \begin{bmatrix} f_i \\ \dot{f}_i \end{bmatrix} \tag{2.19}$$

where f is the state's frequency position and Δf is the frequency resolution used in the STFT.

These modifications also allow two tracks to cross the same point in a spectrogram. The appearance and disappearance of the tracks, which was left unaddressed previously [138], is determined by a sequential test using the mechanism of the F-B algorithm. The tracks are extracted from the spectrogram and their start and end points are calculated using past and future detections.

This technique is not a true general multi-track detector as an upper bound on the number of tracks to be found is a parameter of the algorithm. Tests using this algorithm show that it performs well both with known and unknown SNR, with a slight rise in the mean square error in the latter case. In a test on a synthetic spectrogram with multiple frequency tracks that were highly corrupted the algorithm recovers them all accurately. When the algorithm is applied to a real spectrogram it again accurately detects the frequency tracks. Nevertheless, overestimating the number of tracks increases computational workload, which would not be desirable in a real time application.

Xie and Evans [190] propose a multi-track approach using the Viterbi algorithm, which operates on the thresholded output of the STFT. They define a "mixed" track and use the Viterbi algorithm to produce the maximum *a posteriori* "mixed" track estimates. The estimation of the threshold requires good knowledge of the SNR of the signal under scrutiny. They later present further results [191] that remove the need of thresholding and show superior performance over

the previous method (although this is at the expense of increased storage space). To separate the "mixed" tracks into individual tracks it is proposed to use amplitude and transition probability information. If two tracks do not cross then transition information alone is enough; if they do cross then they are assumed to have different constant amplitudes and this, together with state transition information, is used for separation—it is unclear what effect two tracks having the same amplitude will have.

Simulation results are presented that show good tracking performance when the track's frequency varies by 5 Hz over approximately 11 hours of data. In these over-restricted conditions, which are unrealistic in this problem, the tracker is able to detect a track at an SNR of -23 dB (time domain).

Van Cappel and Alinat propose an alternative HMM approach to multiple track detection; multiple HMMs are utilised to implement several frequency track variation models in parallel [175]. It is noted that the solution to track detection must be designed "firstly in taking into account as long as possible observed data blocks (batch processing), secondly in delaying the decisions (knowledge of future) and thirdly in using several frequency line variation models in parallel" [175]. A HMM is described to extract tracks from thresholded STFT outputs where the threshold is related to the noise level. A generalised likelihood ratio test is performed using two models in parallel as two standard deviation estimates are used; one accounting for stable tracks and the other for unstable. Three track models are taken into account: the first, a stable track with zero order; the second, an unstable track with zero order; and, lastly, a stable track with an order of one. The change from one model to another is triggered by a Bayesian test using the track variation of the recent observations.

Qualitative results are presented for a spectrogram containing tracks that exhibit a large amount of variability. It can be seen that each model has the ability to detect tracks with different characteristics separately and, when combined, the mechanism incorporates the detection attributes of all the models contained.

2.3.5 Tracking Algorithms

Tracking algorithms such as the Kalman filter [93] form a series of estimates, or predictions, of a system state (in this case the track position). Based upon an existing estimate, the state of the system in the next time frame is predicted; once a measurement becomes available (in this case the STFT output) the estimate is updated according to the observation and the process is repeated. An issue associated with this type of detection method, especially when applied to areas that need quick, accurate detections, is the latency of detection, i.e. the number of observations that are required to update the *a priori* estimate to accurately locate and track a feature.

2.3.5.1 Particle Filter

The particle filter is a sequential Monte Carlo method [58], in which the posterior probability density function (PDF) is represented as a set of particles and associated normalised weights in state space, which generalises the Kalman filter [13]. At each time step particles are drawn from

the previously calculated set with probabilities proportional to their weights. The weights of these particles are then updated according to the current observation and used to calculate the Bayesian estimate of the state for the current time step. This is repeated at each time step and has the effect of tracking a state estimate of a non-linear non-Gaussian process, in this case the frequency of a spectrogram track, through time.

Shi and Chang investigate the use of particle filters to extract the formants (peak frequencies of speech signals and therefore tracks) from a spectrogram [163]. Pre-processing converts the spectrogram from log energy to the grey-scale range (0–255). Particle filtering is employed to estimate the state (the frequency) of the kth formant at time t, $\hat{F}_t^{(k)}$, based upon the state estimate in the previous time step, $\hat{F}_{t-1}^{(k)}$, which represents all the previous observations, such that

$$\hat{F}_t^{(k)} = E[F_t^{(k)} | R_t^{(k)}, \hat{F}_{t-1}^{(k)}] \tag{2.20}$$

where $R_t^{(k)}$ is the formant spectrum region (the observation).

The prediction stage updates the current state to predict the frequency location for the next observation, and, as the next observation becomes available, this prediction is updated. The prior $p(F^{(k)})$ and conditional prior $p(F_t^{(k)}|F_{t-1}^{(k)})$ PDFs are assumed to be Gaussian or products of Gaussians,

$$p(F^{(k)}) \sim \mathcal{N}(F^{(k)}; \mu_{F^{(k)}}, \sigma_{F^{(k)}}) \tag{2.21}$$

$$p(F_t^{(k)}|F_{t-1}^{(k)}) \sim \mathcal{N}(F_t^{(k)}; F_{t-1}^{(k)}, \sigma_{F_{t|t-1}^{(k)}}) \tag{2.22}$$

where $\mu_{F^{(k)}}$ and $\sigma_{F^{(k)}}$ are the PDF's mean and standard deviation and are learnt from manually labelled formant tracks. The particle filter algorithm can thus detect the track on a frame-by-frame basis.

In this form, the particle filter is applicable to detecting a single track in a spectrogram. Nevertheless, the paper outlines a method to split the spectrogram into k non-overlapping regions $R^{(k)}$ and to perform tracking in each region, therefore allowing for multiple tracks to be followed.

The results of the experiments presented by Shi and Chang [163] show a mean frequency error of 71, 115 and 113 Hz for the first three formants (it should be noted that the tracks in this application cover a larger range of frequencies compared with the very narrow band tracks discussed in other papers). This is a relatively large error, especially for applications that require accurate frequency estimation to perform subsequent source classification.

2.3.6 Relaxation Methods

Relaxation algorithms search for a global sub-optimal solution to a problem by progressively relaxing constraints, analogous to annealing in metallurgy [100], which involves the heating and controlled cooling of a material to increase the order of its atoms and reduce defects.

2.3.6.1 Simulated Annealing

Lee [112] applies Simulated Annealing (SA) [100, 177] to globally optimise a cost function defined upon the SNR over time. The assumption is made that the initial frequency location is known

and that the track is constrained to a frequency variance of 0, 1 or -1 frequency bins in each time step. This assumption limits the method's application to cases where it is known *a priori* that the spectrogram contains a track. If this is not the case and the method is applied, a false track throughout the spectrogram will be detected. The cost function is defined as

$$C(j) = \sum_{k=1}^{K} (\alpha\mu_k - s_{ja_k}) \tag{2.23}$$

where $(a_k)_{k=\overline{1,K}}$ is monotonically increasing sequence such that $a_k = i$ if s_{ji} belongs to a track and $a_k \neq a_t$, $k \neq t$. The term s_{ja_k} represents the power of the track at point (a_k, j), α is a threshold that controls the detection sensitivity and μ_k is the estimate of noise from the previous track, or the spectrogram border, to the current track, such that

$$\mu_k(j) = \begin{cases} \frac{1}{a_k} \sum_{i=0}^{a_k-1} s_{ji}, & \text{if } k = 1 \\ \frac{1}{a_k - a_{k-1}+1} \sum_{i=a_{k-1}+1}^{a_k-1} s_{ji}, & \text{if } k > 1. \end{cases} \tag{2.24}$$

The global cost function is defined as $C_T = \sum_{j=0}^{M-1} C(j)$, the minimum of which, determined by the SA algorithm, guides the solution towards tracks in the spectrogram.

An initial track configuration is generated at random, which is then incrementally improved using the SA algorithm. This method was tested using a test set containing -18 to 3 dB SNR (time domain) spectrograms that have a single track at 64 Hz. In these experiments the initial frequency location of the track is known and the detection initiated from this frequency bin. The CPU time required to detect a single frequency track within a 128×128 pixel spectrogram varied from 380 to 572 seconds. Qualitative results are presented, which demonstrate reliable detection of time-invariant tracks in most SNRs, with the detection in some cases varying from the true location. Additional experimental results are presented that test the need for accurate *a priori* knowledge of the track's frequency location. The initial state was set to 75 Hz and the experiments repeated with the method successfully recovering the track. This experiment, however, was conducted upon a single spectrogram with a very high SNR of 3 dB (time domain).

2.3.7 Expert Systems

Lu et al. [117] employ the use of an expert system and priority ranking to improve the performance of weak track detection and tracking by allowing for a certain degree of learning. The following stages are followed: the broadband component of the STFT output is removed from the signal, a double threshold is taken where the spectrogram is thresholded with a low threshold value and then a second is applied "to make further judgement according to the characteristics of the shape of the frequency line and timing continuity". The detected frequencies are then stored in an expert database and their initial priority ranks are set to zero. The threshold of each entry in the expert database is adjusted and the narrow region encompassing the initial detection is tested according to the characteristics of a typical track. The priority ranking is reduced or increased depending on the outcome of these tests. A track is eliminated when its priority falls below zero, thus false detections are eliminated.

Qualitative results are presented from the application of the method to one synthetic spectrogram containing 4 tracks, the weakest having an SNR of -9.76 dB (time domain), which demonstrate good detection performance. Another qualitative detection within a real world spectrogram is also presented, but these detections are not quantitatively analysed.

2.4 Discussion

To recapitulate, this chapter has aimed to survey and review algorithms representative of the intersection between the areas of acoustic analysis and pattern recognition for the problem of spectrogram track detection. The intention of such a survey has been to ascertain which approaches have been taken to solve this problem and, in the process, to reveal their limitations, strengths and weaknesses and thus laying the foundations for the research that will be presented in the following chapters of this thesis. To accomplish this, a problem statement, set of evaluation criteria, taxonomy of algorithms and a review of each algorithm from within the taxonomy has been presented. This section presents an evaluation of the algorithms with respect to the defined criteria and a discussion of the algorithms' strengths and limitations.

2.4.1 Algorithm Evaluation

The reviewed algorithms have been evaluated with respect to specific criteria that are prerequisites for a reliable and successful spectrogram track detection algorithm. These criteria have been defined in Section 2.1 and the results are summarised in Table 2.3.

2.4.2 Technique Limitations

In addition to the benefits of each technique, and the insight into the nature of the data that the study of these methods gives us, several fundamental limitations of the techniques that have been presented are identifiable.

- Smoothing of the spectrogram using spatial filtering techniques cannot guarantee that two close tracks have not been merged. It can also cause instances where a detected track has been shifted from the true location through the use of such a filter. These problems carry over to methods employing some form of resolution reduction as a pre-processing stage.

- Di Martino et al. describe problems that follow from using multiple hypothesis testing methods [55], the first being that the number of possible solutions increases dramatically with search depth and, therefore, "thresholding during the search is necessary in order to avoid the combinatory explosion". Also that "the decision process is local and so very sensitive to initialisation".

- Thresholding and likelihood estimates are statistically powerful and simple methods. Nevertheless, when the SNR of a spectrogram is low the probability density functions overlap considerably. Consequently, a low threshold value will result in a high true positive rate but will also detect many false positives. Conversely, if the threshold value is set to a low value the resulting detection will contain few false positives but false negatives start to be the drawback. Another

Algorithm	C1 Low SNR	C2 Temporal Noise Variability	C3 Birth/Death Tracks	C4 Multiple Tracks	C5 Closely Spaced Tracks	C6 Crossing Tracks	C7 High Track Variability	C8 No *A Priori* Shape Assumption	C9 Track Association	C10 Computationally Inexpensive
Maximum Likelihood										
Single frequency [152]	N	Y	Y	N	N	N	Y	Y	Y	Y
Multi harmonic [20]	N	Y	Y	N	N	N	Y	Y	Y	Y
Correlation [8]	-	N	Y	Y	Y	Y	Y	N	N	N
Image Processing Techniques										
Likelihood ratio & morphological operators [3]	-	Y	Y	Y	Y	Y	N	Y	Y	-
Multistage decision process [55]	-	Y	Y	Y	Y	Y	Y	Y	N	Y
Steerable filter & multistage decision [56, 57]	N	Y	Y	Y	N	N	Y	Y	N	Y
Two-pass split-window [38]	N	Y	Y	Y	N	N	Y	Y	N	Y
Edge detector [69]	N	Y	Y	Y	N	Y	Y	Y	N	Y
Neural Networks										
ASM and MLP [99]	N	-	N	N	N	N	N	N	Y	-
Multi-layer perceptron [114]	N	-	Y	Y	Y	-	N	N	N	-
MLP using Ockham's networks [98]	N	-	Y	Y	N	Y	Y	N	N	Y
Kohonen self-organising map [54]	N	Y	Y	Y	N	N	N	Y	N	-
MNET1 [4]	N	-	N	N	N	N	N	N	Y	Y
MNET2 [4]	N	-	N	N	N	N	N	Y	Y	Y
RNET [4]	Y	-	N	N	N	N	N	N	Y	Y
Statistical Models										
Dynamic programming [162]	-	N	N	N	N	N	N	Y	Y	-
Viterbi & max amplitude [169]	-	N	Y	N	N	N	Y	Y	Y	-
Viterbi, "mixed" track & threshold [190]	-	N	Y	Y	Y	Y	N	Y	Y	-
Viterbi & "mixed" track [191]	-	N	Y	Y	Y	Y	N	Y	Y	-
Viterbi & double threshold [165]		Y	Y	Y	Y	N	N	Y	Y	-
Viterbi & PDA [88]	N	Y	N	N	N	N	N	Y	Y	-
Parallel, multi model detection [175]	-	N	Y	Y	-	-	Y	Y	Y	-
F-B linking, SNR estimate & track gradient [137]	Y	Y	Y	Y	-	Y	Y	Y	Y	N
F-B linking & SNR estimate [138]	Y	Y	N	N	N	N	Y	Y	Y	N
Viterbi & SNR estimate [138]	Y	Y	N	N	N	N	N	Y	Y	N
F-B linking & spectrum interpolation [74]	-	Y	N	N	N	N	Y	Y	Y	-
Tracking Algorithms										
Particle filter [163]	-	Y	Y	Y	N	N	Y	Y	Y	N
Relaxation Methods										
Simulated annealing [112]	-	Y	N	Y	Y	N	N	Y	Y	N
Expert Systems										
Double threshold & priority ranking [117]	-	-	Y	Y	Y	Y	Y	Y	N	Y

Table 2.3: Analysis of spectrogram track detection algorithms ('-' denotes the inability to make a judgement regarding the criteria for a specific method due to lack of information).

drawback of these techniques is the constant variation of the noise distribution present in real-world noise environments. This problem then lends itself to machine learning techniques that are adaptive to the environment.

- Although the reviewed RNET and MNET neural network architectures do not account for multiple tracks, track crossing and track birth/death, their HMM counterparts are able to. Due to their close similarity to HMM formulations, these properties should be easily transferred to their implementations.

- The representative work of probabilistic data association (coupled with the Viterbi line detector) and dynamic programming assume that one track is present at any one time frame of the spectrogram. This limitation has been overcome with methods implementing hidden Markov models, some of which incorporate information regarding the current FFT observation into the likelihood measurement, which enables them to model time varying signal-to-noise ratio levels. Nevertheless, many of the implementations that are shown to work in low SNR conditions are tested upon tracks that are relatively stationary (typical variations are 1 Hz over minutes/hours of data). Anderson et al. note that "the transition and measurement probabilities are derived effectively on the assumption that the actual tracks are piecewise constant, which is not at all the case" [11]. If the track varies too greatly the probabilities will not be able to represent the behaviour accurately and therefore the track will not be extracted to the accuracy needed for source classification. The representation of a probability distribution function as a set of particles, as in particle filtering, allows the modelling of non-parametric system state distributions that can be dynamic due to particle population re-sampling at each iteration. Nevertheless, this introduces added computational burden as many particles are needed to produce a good approximation and each of these needs to be updated at each iteration (along with their associated weights). With regard to the proposed HMM solutions, each perform specific aspects of the desired properties however, not one algorithm combines all of the desirable features to fully realise a viable solution.

- The representative work based upon simulated annealing assumes that the initial track position is known. Although experiments have shown that it need not be known accurately to result in the correct detection of a track, it is unclear how much error is allowed for the method to work effectively. This limits the method's application to spectrograms in which a track is known to exist.

- The fundamental SNR limit of current techniques seems to be in the region of 2–4 dB in the frequency domain for tracks that exhibit low shape variation (this is derived by converting time domain SNR levels using assumptions of common spectrogram derivation parameters). This is not sensitive enough for some applications.

- There appears to be a theoretical division in the literature present in this field. A number of methods concentrate on determining the presence of a track within a window of data, and therefore conduct classification, whereas the remainder concentrate on determining the presence of a track at a specific pixel location, and therefore conduct track detection. The practical effect of this divide is that classification mechanisms are applicable, and most often used, to 'clean'

spectrograms, that is, to present the operator with a reduced complexity task where noise is suppressed and "difficult to see" features are highlighted. On the other hand, a reliable track detection mechanism replaces the need for such an operator all together, allowing the output to be directly passed to higher level decision mechanisms (be it an operator or computational system) for further processing.

2.5 Research Directions

The research presented in this thesis will investigate and integrate several beneficial aspects of the work found in the literature and, it will be shown in Chapter 5, will overcome some of their limitations.

The active contour algorithm relies upon internal energies to guide its convergence [96]. In the original algorithm, these energies are defined to be the continuity and curvature of the contour [96] and, as such, are parallels of two of the features used to identify tracks derived by Di Martino et al. [54] from feature grouping theory: temporal continuity and frequential curvature. The additional features that Di Martino et al. propose, high average energy and higher point density, will be further discussed and investigated in Chapter 3 by means of a full investigation into low-level feature detection in this problem.

It is evident from this literature review that there exists a wide range of techniques that have been applied to this problem, all of which rely upon low-level feature detection methods. A vast majority of techniques that have been reviewed utilise information derived from single pixel values for their low-level feature detection, more specifically: those reviewed under the sections Maximum Likelihood Estimates; Statistical Models; Tracking Algorithms; Relaxation Methods; Expert Systems; and a number from the Image Processing [3, 55, 38]. Di Martino and Tabbone [57], Gillespie [69], and the contributions from the Neural Network background (excluding Adams and Evans [4]) build upon this to perform low-level feature detection within windows of the spectrogram, thus incorporating spatial information. Surprisingly, however, no study exists into the effects of low-level feature detection, and so this topic will be addressed in Chapter 3.

An early contribution to the field, proposed by Barrett and McMahon [20], is a simple maximum likelihood detector that integrates information at harmonic locations in the spectrogram to enhance the detection of tracks. In the paper, however, no experimental evidence or investigation is presented to suggest that this improves the detection of tracks in the presence of a harmonic series. Subsequent work on HMM algorithms has attempted to tackle the problem of detecting multiple tracks in spectrograms, however, these are assumed to originate from independent sources and, as such, no information is fused to increase their rate of detection. In Section 3.4 an investigation into the integration of harmonic information is conducted and empirical evidence supporting this claim is presented.

2.6 Conclusions

It is hard to present a direct performance comparison of the outlined techniques as there is a large variation in the type of results presented in the literature. Several papers lack quantitative results, favouring qualitative analysis of one or two spectrograms instead. Furthermore, where quantitative results exist, there is a lack of consistency in the type of data that each technique is tested upon. These inconsistencies include: testing upon synthetic data; real-world data or both; the type of structure variation that tracks exhibit and the SNRs (even the measure of SNR); and noise environment present in the data set. This greatly inhibits the ability to form any direct comparison of results between papers describing different techniques.

The representative work from hidden Markov models and image processing techniques demonstrate applicability to this problem (albeit from different directions), as each of the reviewed solutions demonstrate the ability to achieve one or more of the defined criteria. Nevertheless, it seems that there has been no effort to combine all of these properties into one viable solution and therefore there is still room for improvement in order to meet the challenges posed by present applications.

This survey has been concerned with surveying track detection methods applied to spectrogram images. Techniques exist that include phase information derived from the FFT but these are not reviewed here. For further reading the following is recommended [19, 11, 122, 123].

Chapter 3

Low-Level Feature Detection

"*Do not go where the path may lead,*
go instead where there is no path and leave a trail."

— Ralph Waldo Emerson, 1803–1882.

The first stage in the detection of any object within an image is to extract low-level features. For a spectrogram, this stage results in the identification of unconnected points that are likely to belong to a track, which are output in the form of another image [71]. In Chapter 2 it was found that the approaches to the low-level feature detection of spectrogram tracks present in the literature can be grouped into two categories. Abel et al. [3], Di Martino et al. [55], Scharf and Elliot [162] and Paris and Jauffret [137], to name but a few, take the approach of detecting single pixel instances of the tracks, therefore only intensity information can be exploited in the decision process, and Gillespie [69], Kendall et al. [98] and Leeming [114] take the approach of detecting track sections within windows of the spectrogram, and therefore, in addition to intensity information, information regarding the track structure is exploited in the decision process. Nevertheless, an empirical study of the differences and detection benefits between the two approaches is still lacking. It would be expected that when intensity information degrades, such as in low signal-to-noise ratio spectrograms, the structural information will augment this deficit and thus improve detection rates.

This chapter presents such a study. Firstly three low-level feature detectors are defined, each of which acts upon an increasing amount of information. These are termed 'optimal' detectors as they perform an exhaustive search of the feature space and retain all of the information provided to them by the feature model. The exhaustive search performed by these methods, however, means that they are computationally expensive and, as such, a number of 'sub-optimal' detectors are examined. All of these 'sub-optimal' feature detectors derive feature vectors from within a window and they therefore act upon intensity and structural information, however, they utilise machine learning techniques for dimensionality reduction and class modelling and therefore simplify the search space. The 'sub-optimal' detectors are split into two categories—data-based and model-based—to reflect the source of the training samples utilised by their supervised learning process. Finally, the performance of the model-based 'optimal' feature detector is compared against the model-based

'sub-optimal' feature detector to ascertain the degree of performance divergence between the two approaches.

In addition to this, the final section of this chapter presents a novel transformation that integrates information from harmonic locations within the spectrogram. This is possible due to the harmonic nature of acoustic signals and is defined with the aim of revealing the presence of an acoustic source at low signal-to-noise ratios by utilising all of the information available. The benefits of performing low-level feature detection whilst combining information from harmonic locations are shown at the end of this chapter through a comparison with the detection performance achieved by the low-level feature detectors when applied to the original spectrogram.

3.1 'Optimal' Feature Detectors

Detection methods that utilise dimensionality reduction techniques such as principal component analysis [92] to reduce the model or data complexity, lose information regarding the feature model in the process [28]. This information loss detracts from their ability to detect features and therefore they produce sub-optimal detection results. A method which models the data correctly and does not lose any information in the detection process will have the most discrimination power as a feature detector, under the condition that it correctly models the features to be detected. These types of detectors are more generally referred to as correlation methods in the image analysis domain. In order for such methods to detect features that vary greatly, a model has to be defined with parameters corresponding to each variation type that can be observed. An exhaustive search for the parameter combination that best describes the data is conducted by matching the model to the data by varying its parameters.In this section are defined three detection methods with the properties of an 'optimal' feature detector, i.e. no model reduction or approximation is performed during the search for the feature and therefore they can be termed to be 'optimal' detectors. Three modes of detection have been identified, each of which increases the amount of information available to the detection process from the previous mode: individual pixels; local intensity distribution; and local structural intensity distribution. Individual pixel classification performs detection based upon the intensity value of single pixels. By definition this method makes no assumption as to the track shape and consequently is the most general of the methods in terms of detecting variable structure. A track, however, "is a spectral representation of the temporal evolution of the signal" [54] and, therefore, "can be expressed as a function of the time" [54], i.e. it is composed of a collection of pixels in close proximity to each other. Performing the detection process using individual pixels ignores this fact. An extension to this detection process is therefore to model the pixel value distribution in a local neighbourhood, forming a detector that incorporates this information. Nevertheless, such a detector still ignores the information that can be derived from the arrangement of pixels in the neighbourhood. Such information will enable the detector to distinguish between a number of random high intensity pixels resulting from noise and an arranged collection of pixels that belong to a track.

3.1.1 Bayesian Inference

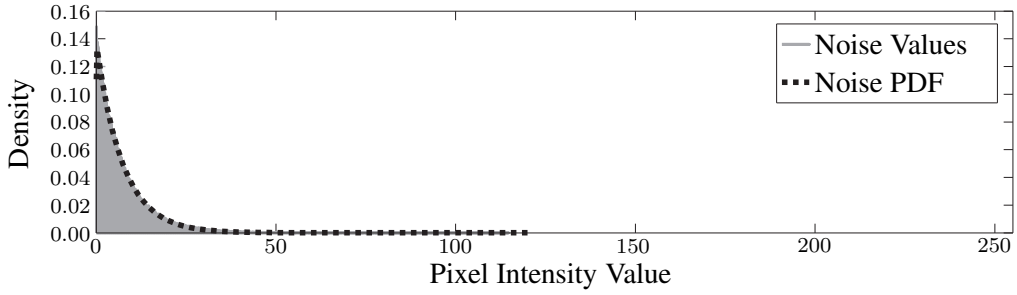
A common method used to model the distribution of individual pixel values makes use of probability density functions. A classification can then be made by testing the pixel's class-conditional membership to distributions describing each class, forming maximum likelihood classification, or, by extending this to act upon a Bayesian decision using the *a posteriori* probability. Assuming that the modelling is accurate, maximum *a posteriori* classification acts upon the optimal decision boundary [60]. In the former case, the class-conditional distribution to which the pixel value has the highest membership determines its classification. In the latter, the decision is made according to the Bayes decision rule and this has been shown to be optimal [60], i.e. it minimises the probability of error (subject to correct design choices).

In this case, Bayesian classification infers a pixel's class membership based upon the probability that it originates from a distribution model of the class' intensity values. The distribution of the intensity values of each class is determined prior to classification as a training stage; the model which best describes the data is chosen and this is fitted to the data by determining applicable parameter values. A similar approach was used by Rife and Boorstyn [152] and Barrett and McMahon [20] who applied maximum likelihood classification to pixel values, however, a very simple class model was used in that work; the maximum value in each spectrogram row was classified as a track position.

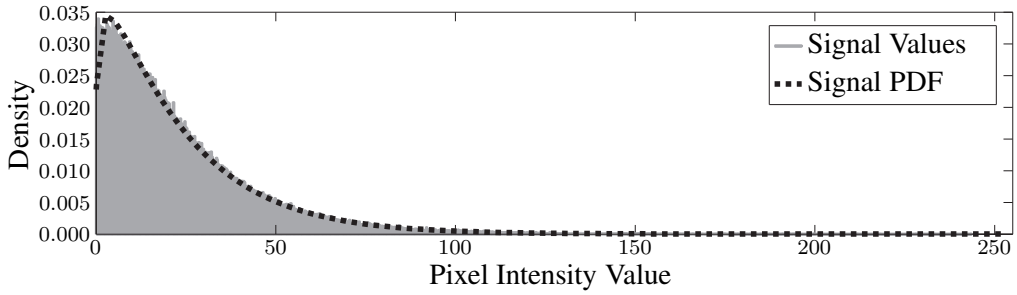
3.1.1.1 Intensity Distribution Models

There are three approaches to density estimation [28]: parametric, non-parametric, and semi-parametric. The first of these, the parametric approach, assumes a specific functional form for the density model, which is fitted to a data set by an appropriate choice of the model's parameters. A drawback of this approach is that the functional form of the model may not accurately represent the data. This limitation is alleviated in non-parametric density estimation, in which no functional form of the density is assumed. Instead, the density is determined by the data and, as a consequence, has the drawback that the number of parameters grows with the cardinality of the data set. This forms a complex model, which can be slow to evaluate for new data points [28]. The third approach balances the previous two and is typically applied as mixture models. These models allow a general class of functional forms in which the number of parameters increases with the complexity of the data and is independent of the size of the data [28]. In this problem, using synthetic data, it is possible to accurately estimate the data's density using the parametric approach, which usually allows the density function to be rapidly evaluated for new data points [28]. In other cases, however, it may be necessary to employ the non-parametric or semi-parametric approach. Nevertheless, the classification technique is equally valid when using different forms of density estimation.

To estimate the parameters of the class-conditional distribution for each class, histograms describing the frequency of intensity values were generated, one for each class, and parametric functions fitted to them. The number of pixel intensity values used to train the models was 266,643 samples of each of the noise and track classes (the data was scaled to have a maximum value of



(a) Noise modelled using an exponential PDF.



(b) Track modelled using a gamma PDF.

Figure 3.1: Class-conditional probability density function fittings for the single-pixel noise, modelled using an exponential PDF (a), and track, modelled using a gamma PDF (b), intensity value distributions. 266,643 randomly chosen pixel values for each class, taken from spectrograms having an SNR range of 0 to 8 dB were histogrammed into 1,000 bins linearly spaced between 0 and 255. The fittings for the signal and noise histograms have a root mean squared error of 0.00048 and 0.00084 respectively.

255 in the training set). These were then histogrammed into 1,000 equally space bins spanning the range 0–255 to form a histogram. As there was a large amount of training data available, the parameter values of each distribution function were determined by maximum likelihood estimation [60] as this has been shown to reach the Bayesian estimation under such conditions [28] and are simpler to evaluate [60] (under the case that there is little training data it may be more appropriate to use Bayesian estimation). The Gamma and Exponential probability density functions (PDF) were found to model the signal and noise distributions sufficiently well as they have a root mean squared error of 0.00048 and 0.00084 (mean error per histogram bin) respectively; histograms of intensity values and the resultant fittings for each class are presented in Fig. 3.1. As such, the class-conditional probabilities of a pixel value, given the hypotheses of noise h_1 and of signal h_2, are determined such that

$$\begin{aligned} P(h_1|s_{yx}) &= \lambda \exp\{-\lambda x\} \\ P(h_2|s_{yx}) &= x^{\alpha-1}\frac{\beta^\alpha \exp\{-\beta x\}}{\Gamma(\alpha)} \end{aligned} \tag{3.1}$$

where $x > 0$, the term Γ represents the gamma distribution and the distribution parameters were found to be $\alpha = 1.1439$, $\beta = 20.3073$ and $\lambda = 7.2764$ (with standard errors of 0.0029, 0.0576 and 0.0144 respectively).

The histograms presented in Fig. 3.1 highlight the fundamental limitation of these methods; there is a large overlap between the distributions of values from each class. This overlap is exaggerated as the SNR is reduced and it can be expected to impede the classification performance of this type of detector.

3.1.1.2 Decision Rules

The simplest form of Bayesian inference, referred to as Maximum Likelihood (ML) [129], is to calculate the class for which the pixel intensity value has the maximum membership. By defining a set of candidate hypotheses $H = \{h_1, h_2\}$, where h_1 and h_2 are the hypotheses that an observation is a member of the noise or signal class respectively, and the probability density functions corresponding to these hypotheses, given the data $s_{yx}, \forall x \in N \wedge y \in M$ (from Section 1.2.3), the likelihood that the data is a result of each hypothesis is determined, such that

$$h_{ML} = \arg\max_{h \in H} P(s_{yx}|h). \tag{3.2}$$

When all the hypotheses in H have equal likelihood of being true any convenient tie breaking rule can be taken [60], in this case a random classification is made.

A drawback of maximum likelihood classification is that it does not take into account the *a priori* probability of observing a member of each class $P(h)$. For example, in the case of taking a random observation with each hypothesis having an equal likelihood of being true, the observation should in fact be classified as belonging to the class that is most likely to be observed [60]. The *a posteriori* probability $P(h|s_{yx})$, which combines the class-conditional and prior, can be computed with Bayes formula,

$$P(h|s_{yx}) = \frac{P(s_{yx}|h)P(h)}{P(s_{yx})}. \tag{3.3}$$

The form of Bayesian decision that incorporates this information, the hypotheses prior probabilities, to form a decision is referred to as Maximum *A Posteriori* (MAP), such that

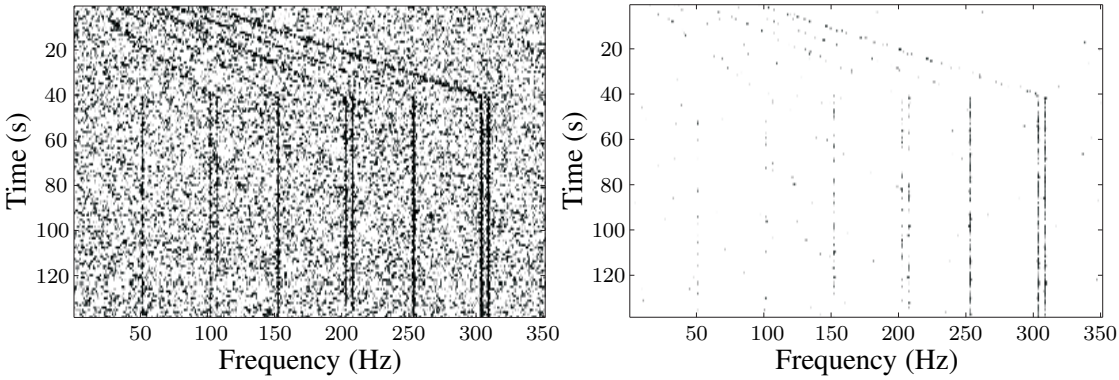
$$h_{MAP} = \arg\max_{h \in H} \frac{P(s_{yx}|h)P(h)}{P(s_{yx})}. \tag{3.4}$$

Note that the ML estimate can be thought of as a specialisation of the MAP decision in which the prior probabilities are equal. The term $P(s_{yx})$ is a normalisation term, which is independent of h, and therefore, does not influence the decision. It can therefore be dropped [60] and Eq. 3.4 reduces to

$$h_{MAP} = \arg\max_{h \in H} P(s_{yx}|h)P(h). \tag{3.5}$$

In the case that the prior probabilities are unknown, which is common, they can be estimated as the frequency of observing each hypothesis within a training set [28], irrespective of its value. In this case the prior probabilities were determined by calculating the frequency of pixels belonging to each class in the training set.

An example of a spectrogram's membership of the noise and track class is presented in Fig. 3.2,



(a) Noise class membership.

(b) Track class membership.

Figure 3.2: Likelihood of class membership, intensity represents likelihood of class membership (scaled to be within 0 and 255). The tracks in this spectrogram have SNRs of, from left to right; first three: 3 dB, middle three: 6 dB and the last three: 9 dB. The intensity of the each response is scale independently.

Figure 3.2a presents the noise membership values of each pixel. It can be seen that the majority of noise pixels have a large likelihood of belonging to the noise class. Nevertheless, the high noise values are found to have a lower likelihood and some of the low SNR tracks are found to have a high likelihood of belonging to this class. Figure 3.2b presents the likelihood of the pixels belonging to the track class and these emphasise the overlap between the two classes. The noise pixels are given a high likelihood of belonging to the track class and track pixels have a low likelihood of belonging to the track class. Taking the maximum membership of each pixel, as defined by Eq. 3.2, a classification of the spectrogram is obtained, Fig. 3.3. Most of the pixels that form a track are correctly classified, although gaps are present in low SNR tracks. The amount of noise in the spectrogram is reduced but there is still a large amount present and this is reflected in the classification percentages for the spectrogram pixels, 78.31% of noise and 71.51% of track is classified correctly.

3.1.2 Bayesian Inference using Spatial Information

Classification based upon single pixel values is limited to forming a decision using only intensity information. The definition of a track, as described in Chapter 1, is that a narrowband component of energy is present in a number of consecutive time frames. A consequence of this is that track pixels will be in close proximity to each other—a property that is not exploited using the classification methods defined above. An alternative method for classification is to determine a pixel's class membership based upon the distribution of pixel values in a local neighbourhood centred upon the pixel, thus exploiting both sources of information. This form of classification, applied to spectrogram track detection, has been investigated by Potter et al. [144] and Di Martino et al. [54] who demonstrate that it can produce high classification rates. A window function is now defined to enable the previously defined classifiers to perform this form of classification.

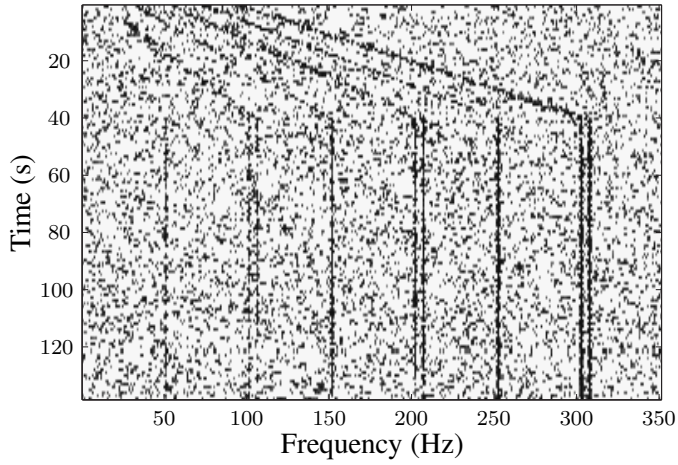


Figure 3.3: An example of maximum likelihood spectrogram pixel classification, in this image likelihood has been encoded as the inverse of intensity and scaled to have a maximum value of 255. The tracks in this spectrogram have SNRs of, from left to right; first three 3 dB, middle three 6 dB and the last three 9 dB.

3.1.2.1 Window Function

The spectrogram $\boldsymbol{S}$ (as defined in Section 1.2.3), can be broken down into I overlapping windows $\boldsymbol{W}$ of predefined size, such that

$$\boldsymbol{W}_{xy} = \begin{bmatrix} s_{y-\rho,x-\gamma} & \cdots & s_{y-\rho,x-1} & s_{y-\rho,x} & s_{y-\rho,x+1} & \cdots & s_{y-\rho,x+\gamma} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \iddots & \vdots \\ s_{y-1,x-\gamma} & \cdots & s_{y-1,x-1} & s_{y-1,x} & s_{y-1,x+1} & \cdots & s_{y-1,x+\gamma} \\ s_{y,x-\gamma} & \cdots & s_{y,x-1} & s_{yx} & s_{y,x+1} & \cdots & s_{y,x+\gamma} \\ s_{y+1,x-\gamma} & \cdots & s_{y+1,x-1} & s_{y+1,x} & s_{y+1,x+1} & \cdots & s_{y+1,x+\gamma} \\ \vdots & \iddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{y+\rho,x-\gamma} & \cdots & s_{y+\rho,x-1} & s_{y+\rho,x} & s_{y+\rho,x+1} & \cdots & s_{y+\rho,x+\gamma} \end{bmatrix} \tag{3.6}$$

$$\gamma = \left\lfloor \frac{n}{2} \right\rfloor, \qquad \rho = \left\lfloor \frac{m}{2} \right\rfloor$$

where $m \in \mathbb{N}$ and $n \in \mathbb{N}$ are odd numbers defining the size of the window (height and width respectively) such that $\gamma < x < f_{max} - \gamma$ and $\rho < y < M - \rho$. A row vector, $\boldsymbol{V}^{xy}$ of size $d = mn$, can be constructed from the values contained within window $\boldsymbol{W}_{xy}$ in a column-wise fashion where $\boldsymbol{C}_r^{xy}$ contains values from the rth column of $\boldsymbol{W}_{xy}$, such that

$$\boldsymbol{C}_r^{xy} = [s_{y-\rho,x-\gamma+r} \; s_{y-\rho+1,x-\gamma+r} \; \cdots \; s_{y+\rho,x-\gamma+r}] \tag{3.7}$$

where $r = 0, \ldots, n-1$, and thus

$$\boldsymbol{V}^{xy} = [\boldsymbol{C}_0^{xy} \; \boldsymbol{C}_1^{xy} \; \ldots \; \boldsymbol{C}_{n-1}^{xy}]. \tag{3.8}$$

3.1.2.2 Decision Rules

Using the window function, the ML hypothesis can be tested for the derived feature vector of pixel values. When the dependency relationships between the pixels are unknown, i.e. under no assumption of the track's shape, the pixels are assumed to be conditionally independent given each hypothesis [60], such that

$$h_{coML} = \arg\max_{h \in H} \prod_{k=1}^{d} P(\boldsymbol{V}_k^{xy}|h). \tag{3.9}$$

Similarly, the MAP classification is modified to take advantage of this information—forming the naïve Bayes rule,

$$h_{coMAP} = \arg\max_{h \in H} \prod_{k=1}^{d} P(h|\boldsymbol{V}_k^{xy}) \tag{3.10}$$

$$= \arg\max_{h \in H} \prod_{k=1}^{d} P(\boldsymbol{V}_k^{xy}|h)P(h) \tag{3.11}$$

where $d = |\boldsymbol{V}^{xy}| \triangleq mn$ is the cardinality of the feature vector $\boldsymbol{V}^{xy}$.

Nota bene to avoid the problem of underflows during the calculation of h_{coML} and h_{coMAP}, the sum of the log likelihoods is taken instead of the product of the likelihoods [60].

3.1.3 Bar Detector

The two previous detectors have been defined to exploit intensity information and also the frequency of intensity values within a window. A final piece of information that can be exploited in the classification process is the arrangement of intensity values within the local window of spectrogram pixels. The independence assumption made in the co-Bayes methods, defined previously, means that they only take into account the presence of multiple track pixels within the window and not the arrangement of these pixels. Thus two disjoint pixels in a window that have high membership to the track distribution will be classified just as two pixels of the same value arranged in consecutive locations. The latter of the two is most likely to be the result of a track being present in the window and the former the result of random noise. This section describes a feature detector that exploits all the information that has been so far outlined. A simple exhaustive line detection method is described that is able to detect linear features at a variety of orientations and scales (width and lengths) within a spectrogram. In accordance with the detectors in this section, this detector can also be viewed as 'optimal' because it detects all variations of the parameters defining the arrangement of pixels belonging to a track within a window in an exhaustive fashion.

First, consider the detection of an arbitrary fixed-length linear track segment and the estimation of its orientation (subsequently this will be extended to include the estimation of its length). The process of detection and inference proceeds as follows: a rotating bar is defined that is pivoted at one end to a pixel $\boldsymbol{g} = [x_g, y_g]$, in a spectrogram $\boldsymbol{S}$, such that $\boldsymbol{g} \in \boldsymbol{S}$ where $\boldsymbol{s} = [x_s, y_s]$, and extends in the direction of the l previous observations, see Fig. 3.4. The values of the pixels that are

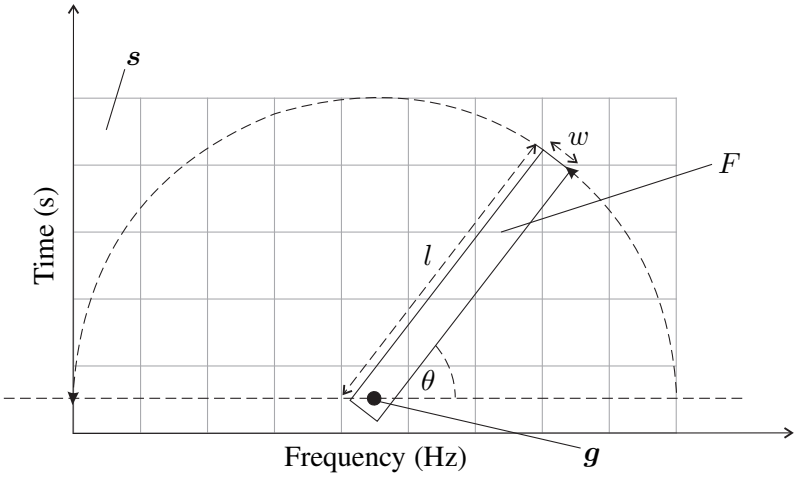


Figure 3.4: The bar operator, having the properties; width w, length l and angle θ.

encompassed by the bar template are defined by the set $F = \{\boldsymbol{s} \in \boldsymbol{S} : P_l(\boldsymbol{s}, \theta, l) \wedge P_w(\boldsymbol{s}, \theta, w)\}$, where

$$\begin{aligned} P_l(\boldsymbol{s}, \theta, l) &\iff \quad 0 \leq [cos(\theta), sin(\theta)][\boldsymbol{s} - \boldsymbol{g}]^T < l \\ P_w(\boldsymbol{s}, \theta, w) &\iff \quad \left|[-sin(\theta), cos(\theta)][\boldsymbol{s} - \boldsymbol{g}]^T\right| < \tfrac{w}{2}, \end{aligned} \tag{3.12}$$

and where θ is the angle of the bar with respect to the x axis (varied between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$ radians), w is the width of the bar and l is its length. The pixels in F are summed, such that

$$B(\theta, l, w) = \frac{1}{|F|} \sum_{\boldsymbol{f} \in F} \boldsymbol{f}. \tag{3.13}$$

To reduce the computational load of determining $P_w(\boldsymbol{s}, \theta, l)$ and $P_l(\boldsymbol{s}, \theta, l)$ for every point in the spectrogram, $\boldsymbol{s}$ can be restricted to $x_s = x_g - (l+1), \ldots, x_g + (l-1)$ and $y_s = y_g, \ldots, y_g + (l-1)$ (assuming the origin is in the bottom left of the spectrogram) and a set of templates can be derived prior to runtime to be convolved with the spectrogram. The bar is rotated through 180 degrees, $\theta = [-\frac{\pi}{2}, \frac{\pi}{2}]$, calculating the underlying summation at each $\Delta\theta$.

Normalising the output of $B(\theta, l, w)$ forms a brightness invariant response $\bar{B}(\theta, l, w)$ [131], which is also normalised with respect to the background noise, such that

$$\bar{B}(\theta, l, w) = \frac{1}{\sigma(B)} \left[B(\theta, l, w) - \mu(B)\right] \tag{3.14}$$

where σ is the standard deviation of the response and μ its mean.

Once the rotation has been completed, statistics regarding the variation of $B(\theta, l, w)$ can be calculated to enable the detection of the angle of any underlying lines that pass through the pivoted pixel $\boldsymbol{g}$. For example, the maximum response, such that

$$\theta_l = \arg\max_{\theta} \bar{B}(\theta, l, w). \tag{3.15}$$

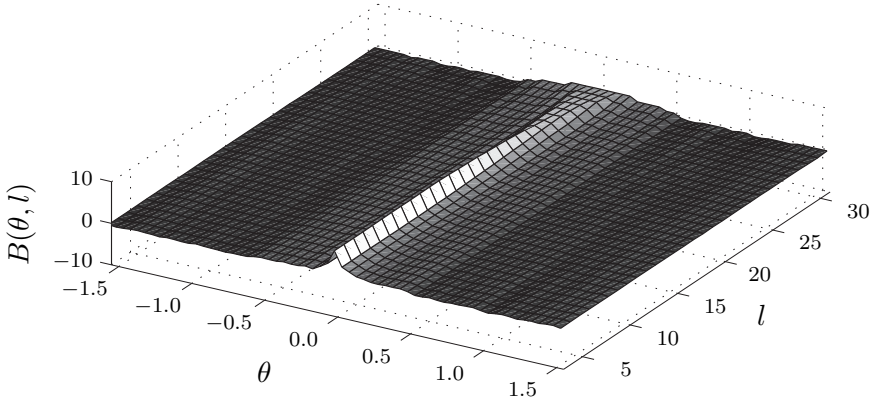


Figure 3.5: The mean response of the rotated bar operator centred upon a vertical line 21 pixels in length (of varying SNRs). The bar is varied in length between 3 and 31 pixels.

Assuming that the noise present in a local neighbourhood of a spectrogram is random, the resulting responses will be low. Conversely, if there is a line present, the responses will exhibit a peak in one configuration, as shown in Fig. 3.5. Thresholding the response at the angle $\bar{B}(\theta_l, l, w)$ allows these cases to be detected. This threshold will be chosen such that it represents the response obtained when the bar is not fully aligned with a track segment.

Repeating this process, pivoting on each pixel $\boldsymbol{g}$ in the first row of a spectrogram and thresholding, allows for the detection of any lines that appear during time updates.

This process will now be extended to facilitate the detection of the length l. For simplicity, and without loss of generality, the line's width is set to unity, i.e. $w = 1$. To estimate the line's length Eq. (3.15) is replaced with

$$\theta_l = \arg\max_{\theta} \sum_{l \in L} \bar{B}(\theta, l, w), \tag{3.16}$$

where L is a set of detection lengths, to facilitate the estimation of the angle over differing lengths. Once the line's angle θ_l has been estimated $\bar{B}(\theta_l, l, w)$ is analysed as l increases to estimate the line's length.

The response of $\bar{B}$ is dependent on the bar's length, as this increases, and extends past the line, it follows that the peak in the response will decrease, as illustrated in Fig. 3.5. The length of a line can therefore be estimated by determining the maximum bar length in which the response remains above a threshold value: $l_l = \max(L_p)$, where L_p is defined such that

$$L_p = \{l \in L : \bar{B}(\theta_l, l, w) > \frac{3}{4} \max(\bar{B}(\theta_l, l, w))\}. \tag{3.17}$$

An arbitrary threshold of $3/4$ of the maximum response found in $\bar{B}(\theta_l, l, w)$ is taken (the threshold value could alternatively be learnt in a training stage).

3.1.3.1 Length Search

The estimation of a line's length using the linear search outlined above is particularly inefficient and has a high run-time cost. To reduce this, the uniform search strategy is replaced with the more

Algorithm 3.1 Bar length binary search
Input: l_{low}, the minimum length to search for, l_{high}, the maximum length to search for, T, a threshold, θ_l, the line's orientation, S, a spectrogram image
Output: l_l, the length of an underlying line.

```
1:  if B̄(θ_l, l_low, w) > T then
2:      p_low ← l_low + 1
3:      p_high ← l_high + 1
4:      while p_low ≠ l_low ∧ p_high ≠ l_high do
5:          p_low ← l_low
6:          p_low ← l_high
7:          l ← ⌊(l_low + l_high)/2⌋
8:          if B̄(θ_l, l, w) > T then
9:              l_low ← l
10:         else {the line's length has been exceeded}
11:             l_high ← l
12:         end if
13:     end while
14:     l_l ← l_low
15: else {a line does not exist}
16:     l_l ← 0
17: end if
18: return l_l
```

efficient binary search algorithm outlined in Algorithm 3.1. Implementing the search in this way reduces the associated search costs from $\mathrm{O}(n)$ to $\mathrm{O}(\log n)$, allowing searches to be performed for a large number of line lengths. The same algorithm can be used to search for the line's width, further reducing the cost.

3.2 'Sub-Optimal' Feature Detectors

A limitation of the 'optimal', correlation detection methods is that they are computationally feasible only for models with few parameters. As the number of parameters increase, the size of the search space increases exponentially—forming an intractable solution. For example, a simple deterministic sinusoidal model contains five free parameters: fundamental frequency position; scaling; track amplitude; phase and frequency; which requires a solution of $\mathrm{O}(n^5)$ complexity.

Dimensionality reduction techniques remove potentially unneeded information and therefore reduce the search space by simplifying the model or, alternatively, the data. This is an important step in the classification process as the act helps to avoid the curse of dimensionality [60]; a problem that states that for each additional dimension, exponentially more samples are needed to span the space. Moreover, data that has some underlying low-dimensional structure may be embedded in high-dimensional space and the additional dimensions are likely to represent noise [28]. If these additional dimensions can be removed, leaving the low-dimensional structure intact, the problem is simplified.

As outlined earlier, these methods should not achieve the performance of the 'optimal' de-

tectors due to information loss. Nevertheless, the increase in computing performance, and the non-specificity that occurs as a result of the problem simplification ('optimal' detectors are specific to detecting structures that are dictated by their models) merits their use.

A low dimension subspace is typically learnt by supervised learning methods and as such can be derived in two ways: data-based and model-based. Data-based methods determine the subspace using real examples of the data to be classified by constructing a training set. This training set could contain noise and random variations of the feature that occur in the real world, however, it is often difficult to construct a training set that fully represents these complex variations. On the other hand, model-based methods generate the data used for training from a model and, therefore, are limited to the model's ability to represent the complexity of the problem. This section presents feature detection methods that are examples of both methods.

3.2.1 Data-Based Subspace Learning

It is common in the area of machine learning that a classification, or decision, is based upon experience [128]. The experience can take the form of a data set, a training set, which contains examples of the data to be classified and labels describing the class to which the examples belong. This is what is referred to as data-based learning. This data set should encompass the primary variations that are possible in the data so that the classifier is able to learn the underlying process that generates the data [60]. In the problem of remote sensing, data is scarce and it may not be possible to construct such a training set. Consequently, techniques that utilise such machine learning methods may be limited in their ability to generalise to unseen complex track structures.

The window function outlined in Section 3.1.2.1 splits the spectrogram into overlapping windows and constructs high dimensional feature vectors from the intensity values contained within these windows. Feature vectors from multiple windows concatenated together form a set of data that can be used to train and test the classification algorithms presented in this section.

3.2.1.1 Explicit Dimension Reduction

Dimensionality reduction techniques have been investigated throughout the history of pattern recognition. They offer the ability to visualise high dimensional data and to simplify the classification process, for reasons previously outlined.

There has recently been a renewed interest in the development of dimensionality reduction techniques, with particular application to high dimensional data visualisation. Recent algorithm contributions include: ISOMAP [170], Locally Linear Embedding (LLE) [154], Stochastic Neighbourhood Embedding (SNE) [82], Laplacian Eigenmapes (LE) [24], Kernel Matrix [181], local tangent space aligning (LTSA) [198], Essential Loops [113], Neural Networks [83], t-SNE [176], and general graph based frameworks to unify different dimensionality reduction techniques [195]. Nevertheless, implemented as batch techniques, these methods require all training and testing samples to be given in advance. Embedding a novel data point into the space requires a complete recalculation of the subspace—a computationally expensive process. In recent years there has been a move to address this issue and researchers are introducing incremental learning algorithms

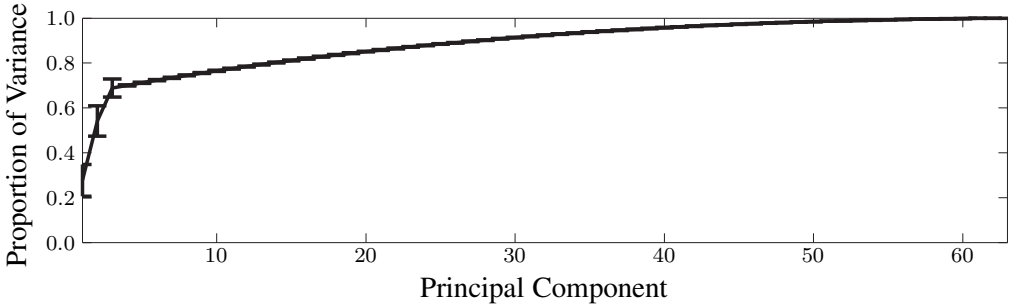


Figure 3.6: Windowed spectrogram PCA eigenvalues. The eigenvalues were determined using a data set of 1,000 samples data samples of each class taken from spectrograms having a mean SNR of 8 dB.

such as: incremental versions of MDS, ISOMAP, LLE and LE [26]; LLE [106]; ISOMAP [109]; LTSA [116]; and incremental Laplacian Eigenmaps [90]. It is beyond the scope of this thesis to evaluate these methods with application to this data and therefore this section concentrates on evaluating the well established techniques of principal component analysis (PCA) [140, 67], linear discriminant analysis (LDA) [23] and neural networks. These methods are suitable for classification problems as they calculate basis vectors that allow novel data points to be projected into the low dimensional space with no added computational burden.

Statistical methods such as PCA and LDA attempt to determine a subspace in which a measure of the data's variance is maximised. The key difference between the two methods is that they measure the variance in different manners: PCA takes the data's global variance, and LDA the within and between class variances. Consequently, both methods determine subspaces that represent different features of the data, PCA globally extracts the most significant features from the data set whereas LDA attempts to extract the most significant features that separate the classes. Neural networks incrementally determine a subspace in which the sum-of-squares error of a training or validation set is at a minimum [28]. If the correct network and activation functions are applied to the data, this translates into a projection in which the properties of the data that are most relevant to learning the target function are captured [129].

The eigenvalues obtained by applying principal component analysis to a training set comprising 1,000 data samples (3×21 pixel window instances) of each class randomly selected from a spectrogram having a SNR of 8 dB are presented in Fig. 3.6. A majority of the data's variance is contained within the first three principal components and the remaining components have little variance. Figure 3.7 presents the distribution of windows containing vertical tracks and noise (selected randomly from spectrograms having SNRs of 3 dB and 6 dB) after projection onto the first two principal components. In this form the classes are neatly clustered. A high proportion of the noise is clustered in a dense region and three protrusions from this cluster contain the data samples from the track class—each of the protrusions corresponds to each of the three possible positions of a straight vertical track in a window three pixels wide. As the SNR of the track contained within a window increases, its projected distance from the noise class increases proportionally. There is some overlap between low SNR track data points and the noise cluster, which emphasises the

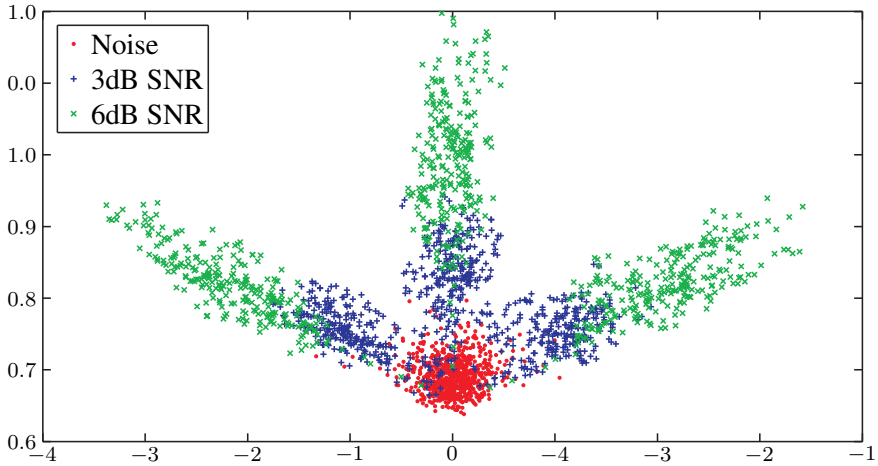


Figure 3.7: Windowed spectrogram projected onto the first two principal components.

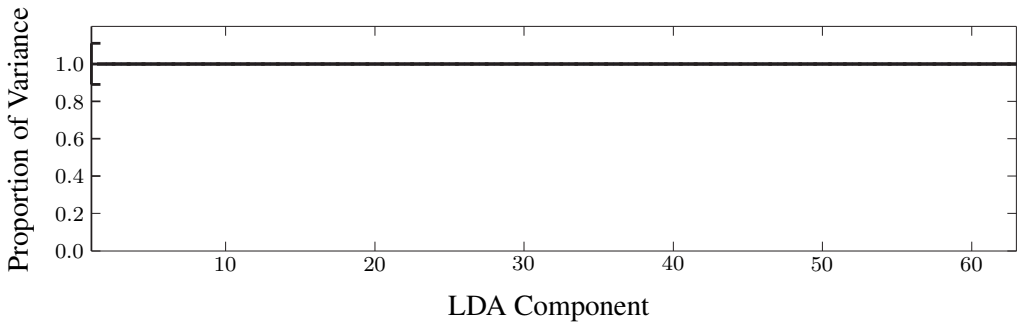


Figure 3.8: Windowed spectrogram LDA eigenvalues. The eigenvalues were determined using a data set of 1,000 samples data samples of each class taken from spectrograms having a mean SNR of 8 dB.

problems of separation between these two classes found earlier in the investigation. The windows containing high SNR tracks (greater than 3 dB) are well separated from the noise in this projection.

Figure 3.8 presents the eigenvalues derived through LDA. The eigenvalues of LDA when applied to the same data set as used previously for PCA indicate that all of the variance can be represented with one component. The result of projecting the data onto the first two components is presented in Fig. 3.9. The samples from different locations of the window are not as cleanly separated as was found with PCA. The most likely explanation for this is that LDA maximises the between-class variation and not the data's global variance. Nevertheless, the separate class clusters are preserved in the projection. As with PCA, LDA cannot separate the overlap between the low SNR track samples and the noise cluster, but high SNR samples are still well separated from the noise.

3.2.1.2 Implicit Dimension Reduction

Neural networks perform dimensionality reduction when in specific topologies [95]—a three-layer Multi-Layer Perceptron (MLP) that has a hidden layer with fewer nodes than the input and output layers compresses the data—thus implicitly reducing the data's dimensionality [28]. The same

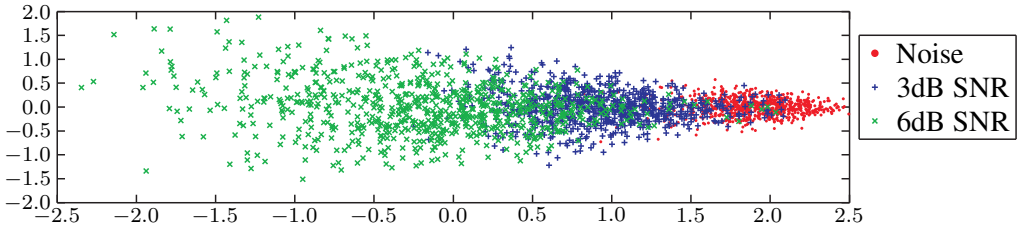


Figure 3.9: Windowed spectrogram projected onto the first two LDA principal components.

is true for the Radial Basis Function (RBF) network, in which radial basis functions are used as the activation functions. The self-organising map (SOM) [103, 104] performs dimensionality reduction in a very different manner. The SOM reduces the dimensionality in a manner similar to the explicit dimensionality reduction techniques discussed in the previous section. It often takes the form of a two-dimensional array of nodes that use a neighbourhood function to model the low dimensional structure in high dimensional data.

3.2.1.3 Classification Methods

To quantitatively evaluate the effectiveness of dimensionality reduction and to determine the applicability of classifiers to this problem, the performance of a range of classifiers is evaluated in this section. Each of the classifiers will be evaluated using the original, high dimensional, data in addition to the low dimension data.

The following classifiers are evaluated in this section: the Radial Basis Function (RBF); Self Organising Map (SOM); k-Nearest Neighbour (KNN); and Weighted k-Nearest Neighbour (WKNN). In addition to these, simpler distance based classification schemes are also evaluated. The class c that minimises the distance d, for each feature vector $\boldsymbol{V}^{xy}$, is taken to be the classification of the feature vector, such that

$$c^k = \arg\min_{c \in C} d(\boldsymbol{V}^{xy}, \boldsymbol{\mu}_c). \tag{3.18}$$

The distance measure d can be taken to be the Euclidean distance d_1, or the Mahalanobis distance d_2, such that

$$d_1(\boldsymbol{V}^{xy}, \boldsymbol{\mu}_c) = \sqrt{(\boldsymbol{V}^{xy} - \boldsymbol{\mu}_c)^T (\boldsymbol{V}^{xy} - \boldsymbol{\mu}_c)} \tag{3.19}$$

$$d_2(\boldsymbol{V}^{xy}, \boldsymbol{\mu}_c) = \sqrt{(\boldsymbol{V}^{xy} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\boldsymbol{V}^{xy} - \boldsymbol{\mu}_c)} \tag{3.20}$$

where $\boldsymbol{V}^{xy}$ and $\boldsymbol{\mu}_c$ and $\boldsymbol{\Sigma}_c^{-1}$ are the mean vector and the inverse of the covariance matrix of each class c in the training set respectively. When the Mahanalobis distance is in use and the covariance matrix is diagonal, the normalised Euclidean distance is formed, which will be evaluated as the third distance measure d_3.

Furthermore, the structure observed in the low dimensional representations obtained using PCA and LDA suggest that the noise class can be modelled using a multivariate distribution. An additional classifier is therefore formed by modelling the noise class with a multivariate Gaussian

distribution,

$$G(\boldsymbol{V}^{xy}) = \frac{1}{2\pi^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{V}^{xy} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{V}^{xy} - \boldsymbol{\mu})\right\}, \tag{3.21}$$

where $|\boldsymbol{\Sigma}|$ and $\boldsymbol{\Sigma}^{-1}$ are the determinant and inverse of the noise classes' covariance matrix respectively. The output of which can be thresholded to determine the feature's membership to the noise class, such that

$$h = \begin{cases} h_1 & \text{if } G(\boldsymbol{V}^{xy}) > \epsilon, \\ h_2 & \text{otherwise.} \end{cases} \tag{3.22}$$

The data used during this experiment was as follows; the training set consisted of $6{,}732$ samples of 3×21 pixel windows taken from spectrograms that contain vertical tracks having SNRs of $0\,$dB. This window size was chosen as during preliminary experiments it was found to provide acceptable results (see Appendix A, Figure A.2). The test set, containing the same number of samples and window configuration, contained examples of tracks having an SNR of 0, 3 and $6\,$dB. It was found during preliminary experimentation that the multilayer perceptron neural network does not perform well compared with the RBF and SOM networks and therefore results obtained using this classifier are not included in this chapter.

Each of the classifier's parameters were chosen to maximise generalisation performance and were determined through preliminary experimentation, these are as follows. The KNN and WKNN classifier used ten nearest neighbours to determine the class of the novel data point. In the event of a tie, a random classification was made. An RBF classifier with five Gaussian activation functions and two training iterations has been implemented as this was found to perform well in preliminary experimentation. The RBF basis centres were determined by k-means clustering [28]; the variance of the basis functions were taken as the largest squared distance between the centres. The RBF weights were determined using the pseudo inverse of the basis activation levels with the training data [28]. A rectangular lattice of SOM nodes was used—the size of which was determined automatically by setting their ratio to be equal to the ratio of the two biggest eigenvalues of the data set [104]. The Gaussian model defined in Eq. (3.21) was fitted to the noise class by calculating its mean and standard deviation.

The classification performance of each classifier applied to the original data and the same data projected into a low dimensional subspace determined through PCA and LDA is presented in Table 3.1 (and the standard deviations attributed to these results are presented in Table 3.2). These results demonstrate that classification performance using these features can reach 84% with a standard deviation of 4% when applied to the test dataset (using the RBF classifier in a three dimensional subspace derived through PCA). The classification performance using the training data set is lower than that observed using the test data set as the classifiers were trained using more complex data than that with which they were tested. The training data comprised of instances of windows containing noise and track having an SNR of $0\,$dB and, upon this data, the majority of classifiers obtain a classification percentage between 71 and 78% with standard deviations between 2% and 5%. These results demonstrate that the dimensionality reduction techniques extract meaningful information from the data even at low SNRs. By testing the classifiers upon a dataset

Classifier	Window	PCA 2D	PCA 3D	PCA 4D	PCA 5D	LDA 2D	LDA 3D	LDA 4D	LDA 5D
KNN — tr	77.8	75.9	**79.5**	78.5	79.0	78.4	78.0	78.4	78.0
KNN — te	81.5	78.5	**83.3**	82.7	83.1	80.1	80.6	80.8	79.6
WKNN — tr	77.5	76.1	**79.7**	79.5	79.5	79.1	78.0	77.1	78.0
WKNN — te	80.8	77.0	**83.4**	83.1	82.2	81.0	80.6	80.3	80.5
RBF — tr	75.6	73.0	**77.3**	76.6	76.0	76.5	75.6	76.6	75.6
RBF — te	81.8	81.9	**84.4**	83.8	83.3	81.8	82.1	81.8	80.8
SOM — tr	80.4	78.8	81.3	**81.5**	80.5	80.3	80.2	79.2	80.2
SOM — te	79.6	74.3	**80.8**	79.9	80.5	77.5	78.3	77.0	76.1
Euclid. (d_1) — tr	76.4	63.1	74.0	74.5	75.6	**76.7**	75.4	76.6	76.3
Euclid. (d_1) — te	81.1	66.4	81.2	81.5	81.0	**82.3**	81.4	80.5	80.9
Mahalanobis (d_2) — tr	54.9	60.2	71.2	69.4	67.3	**75.8**	71.6	71.1	69.4
Mahalanobis (d_2) — te	54.6	65.3	81.2	77.5	77.0	**81.8**	79.7	79.1	75.8
N. Euclid. (d_3) — tr	52.4	59.8	68.9	66.0	62.6	**75.7**	73.2	71.2	68.8
N. Euclid. (d_3) — te	54.0	63.3	78.6	74.4	69.9	**82.0**	81.0	78.6	77.1
Gaussian ($G(\boldsymbol{V}^{xy})$) — tr	50.1	66.1	71.8	73.5	**74.8**	61.0	65.6	67.4	69.5
Gaussian ($G(\boldsymbol{V}^{xy})$) — te	50.3	76.1	81.5	82.0	**82.2**	68.1	72.3	74.4	74.8

Table 3.1: Classification percentage on training (tr) and test (te) data using the proposed features. The highest classification percentage for each classifier is highlighted in bold and the highest percentage for each feature is underlined. The standard deviations of these results are presented separately in Table 3.2.

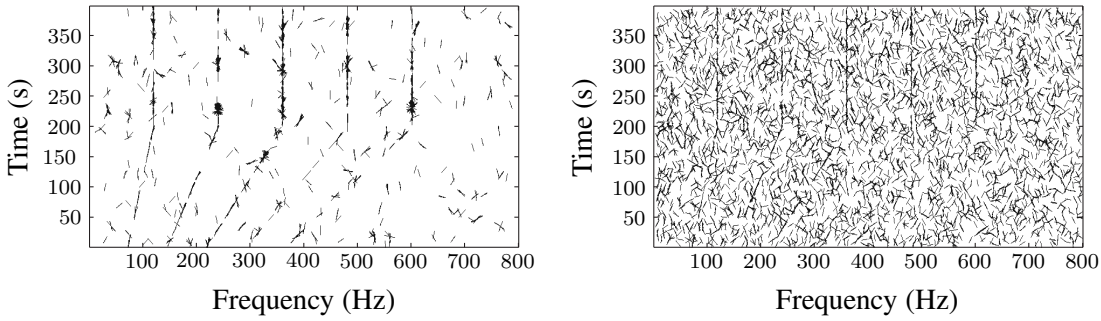
Classifier	Window	PCA 2D	PCA 3D	PCA 4D	PCA 5D	LDA 2D	LDA 3D	LDA 4D	LDA 5D
KNN — tr	2.50	4.77	2.72	4.24	2.73	3.15	2.95	2.61	3.83
KNN — te	3.44	8.78	2.72	3.29	2.84	2.92	3.52	3.61	3.79
WKKN — tr	3.87	5.07	2.79	4.17	3.69	2.69	2.66	3.21	4.13
WKNN — te	4.44	7.44	1.97	3.58	2.51	4.53	2.37	4.48	3.67
RBF — tr	4.40	5.16	4.19	4.02	4.47	2.45	2.91	2.40	2.68
RBF — te	2.92	5.31	2.77	2.97	2.83	3.73	3.11	2.64	4.54
SOM — tr	1.74	3.06	2.41	2.67	1.97	3.22	3.08	2.73	3.52
SOM — te	4.63	7.00	3.80	3.55	5.29	6.84	5.35	3.78	4.55
Euclid. (d_1) — tr	2.08	11.03	2.77	3.13	3.02	2.59	3.57	3.17	3.90
Euclid. (d_1) — te	2.56	13.11	3.50	2.29	3.29	1.42	3.66	2.99	3.01
Mahalanobis (d_2) — tr	2.47	14.06	2.90	3.35	3.80	3.27	2.94	4.38	3.45
Mahalanobis (d_2) — te	3.12	19.96	2.92	2.00	4.52	2.21	3.06	4.14	5.50
N. Euclid. (d_3) — tr	1.57	10.14	4.17	5.68	4.66	3.37	3.49	4.75	3.43
N. Euclid. (d_3) — te	3.05	14.09	4.54	7.64	10.69	2.10	3.77	4.83	3.19
Gaussian ($G(\boldsymbol{V}^{xy})$) — tr	0.32	6.74	2.82	4.09	3.30	5.92	5.80	4.75	5.00
Gaussian ($G(\boldsymbol{V}^{xy})$) — te	0.50	10.69	2.59	4.80	2.07	2.84	5.77	5.47	3.07

Table 3.2: Standard deviation of the classification performance presented in Table 3.1.

comprising windowed instances of noise and tracks that have an SNR greater than or equal to 0 dB (in this case 0, 3 and 6 dB) it is possible to demonstrate that the dimensionality reduction techniques allow the classifiers to generalise to higher, unseen, SNRs while trained upon track instances that have very low SNRs.

Several of the classifiers perform badly when applied to the original windowed data; the normalised Euclidean, Mahalanobis, and Gaussian classifiers all have a classification performance between 50% and 55% upon the original test data. Nevertheless, when the data is projected into a lower dimension subspace derived through PCA or LDA this performance increases to between 63% and 76%. This indicates that the dimension reduction techniques have removed noise present in the original feature vectors and have allowed the, relatively simple, classifiers to correctly model the data's structure. Furthermore, this has reduced the performance gap between these and the more complicated classifiers.

It was shown by Kendall et al. [98] that the generalisation performance of a neural network classifier, when applied to this problem, can be further improved through Ockham's networks



(a) Bar detection method.

(b) Parametric manifold detection method.

Figure 3.10: Spectrogram detections (2.18 dB SNR in the frequency domain) using the proposed bar method and the parametric manifold detection method.

[97]. These experiments, however, were conducted, and shown to perform best, on a low number of training samples (121 examples) and therefore this technique was not tested in this section.

3.2.2 Model-Based Subspace Learning

The previously evaluated techniques determine a low dimension subspace using examples of the data to be classified and in which the classification performance of this data is optimised. An alternative approach to calculating the subspace is by utilising a model describing the data and not the data itself—a feature detector in this vein is described by Nayar et al. [131]. In such techniques the data used to train the detection mechanism is generated from a model that is constructed such that it describes each observable variation that can exist in the problem. Training the detection mechanism in this way allows the exact underlying nature of the problem to be captured by the learning technique.

The feature detector proposed by Nayar et al. [131], like the bar detector proposed in Section 3.1.3, is a model-based feature detection method. The primary difference between the two is that Nayar et al. propose to construct a sampled manifold in a feature space derived through PCA. Detection is achieved by calculating the closest point on the manifold to a sample taken from an image (nearest neighbour classification) and thresholding the distance if necessary. The bar detector performs the detection without the construction of the manifold, instead, the image sample's responses as the model is varied are analysed and the best fit is found from the match between sample response and model. This avoids the loss of information that is an effect of dimensionality reduction. This equivalence justifies a direct comparison between the two methods and, more importantly, a comparison between an 'optimal' and a 'sub-optimal' detector that model the data equivalently and differ only in the presence and absence of a dimension reduction step.

The execution times of the proposed method and that outlined by Nayar et al. were measured within one 398×800 pixel spectrogram using Matlab 2008a and a dual-core 2.0 GHz Intel PC. As the method proposed by Nayar et al. is not multi-scale the length of the bar is fixed $L = 13$ to facilitate a fair comparison, additionally, the parametric manifold was constructed using the same parameter range and resolution as used in the bar model. The bar detector performed the detection

in 5.5 min whereas the comparison performed the detection in 3.4 min and the resulting detections can be seen in Fig. 3.10. Although this is far from an exhaustive test it does highlight a benefit of dimension reduction—the duration of the detection process is reduced with the complexity of the model. In the detection results presented the threshold for each method was chosen such that a true positive rate of 0.7 was achieved. This allows equivalent false positive rates to be compared and it becomes apparent that the speed offered by the 'sub-optimal' method is achieved at the price of detection performance—the detector utilising the dimension reduction technique results in a false positive rate of 0.163 and the bar detector a false positive rate of 0.025.

A more exhaustive comparison between all the feature detectors described in this chapter is presented in the next section.

3.3 Evaluation of Feature Detectors

The feature detectors that are outlined in this chapter have been evaluated along with several common line detection methods found in the literature; the Hough transform [59] applied to the original grey-scale spectrogram; the Hough transform applied to a Sobel edge detected spectrogram; convolution of line detection masks [71]; Laplacian line detection [71]; and pixel value thresholding [71]. Due to its simplicity and comparable performance to more complex methods, the classification scheme that combines PCA and the Gaussian classifier outlined in Section 3.2.1.3 will be evaluated here.

During preliminary experimentation it was found that forming a six dimensional subspace using -0.5 dB (mean SNR) samples provides the best detection performance (to improve readability these results are presented in Appendix A, Figure A.1) and, as discussed in Section 3.2.1.3, that using a window size of 3×21 provided acceptable results (Appendix A, Figure A.2).

The performance of each feature detector can be characterised by determining its Receiver Operating Characteristic (ROC) [63]. A two-dimensional ROC graph is constructed in which the True Positive Rate (TPR) is plotted in the x-axis and False Positive Rate (FPR) is plotted in the y-axis. The TPR (also known as sensitivity, hit rate and recall) of a detector is calculated such that

$$TPR = \frac{TP}{TP + FN} \tag{3.23}$$

where TP is the number of True Positive detections and FN is the number of false negative detections. The FPR (also known as the false alarm rate) is calculated such that

$$FPR = \frac{FP}{FP + TN} \tag{3.24}$$

where FP is the number of False Positive detections and TN is the number of True Negative detections. For a full introduction to ROC analysis the reader is referred to Fawcett [65], which appears in a special issue of Pattern Recognition Letters dedicated to ROC analysis in pattern recognition.

Track Type	Parameter	Values
Vertical	Signal Duration (seconds)	100
	SNR (dB)	-1–7
Oblique	Track Gradient (Hz/s)	1, 2, 4, 8 & 16
	Signal Duration (seconds)	100
	SNR (dB)	-1–7
Sinusoidal	Period (seconds)	10, 15 & 20
	Centre Frequency Variation (%)	1, 2, 3, 4 & 5
	Signal Duration (seconds)	200
	SNR (dB)	-2–6

Table 3.3: Parameter values spanning the synthetic data set.

3.3.1 Experimental Data

Using the signal model outlined in Section 1.2.1, a set of spectrogram images is generated for use in the evaluation of the proposed low-level feature detectors. The spectrograms are formed by generating synthetic acoustic signals and transforming these to form spectrograms using the process described. Time-series signals are created and contain a fundamental frequency of $\omega_0^t = 120\,\text{Hz}$ (at constant speed), a harmonic pattern set $\boldsymbol{P}_s = \{1, 2, 3, 4, 5\}$, and have a sampling rate of $f_s = 4{,}000\,\text{Hz}$ (to ensure high fidelity in the representation of frequency modulations). The fundamental and harmonic series are chosen to be representative of values true to small boat observations. Spectrograms are generated from these using a time resolution of one second with a half second overlap, and a frequency resolution of 1 Hz per STFT bin. The three variations of track appearance that are commonly seen in this problem are: sinusoidal, representing a Doppler shifted signal; vertical, representing a constant engine speed; and oblique, representing an accelerating engine. A number of noise-only spectrograms were also included in the data set. A description of the parameter variations used for these three signal types is outlined in Table 3.3. For each parameter combination, one spectrograms is generated to form a test set, and another to form a training set to facilitate the application of the machine learning techniques. The parameters described in Table 3.3 determine the appearance of each type of track and are defined as:

Period — The time in seconds between two peaks of a sinusoidal track;

Centre Frequency Variation — The amplitude of a sinusoidal track relative to its frequency location, expressed as a percentage of the track's frequency;

SNR — The frequency domain SNR, described by Eq. 1.16;

Track Gradient — The amount of change in the track's frequency relative to time.

The values of these parameters are chosen to cover meaningful real-world observations. The effects of these upon the appearance of the sinusoidal track class are illustrated in Fig. 3.11. To ensure an accurate representation of the SNR, the final value is calculated within the resulting spectrogram and therefore may deviate from the value specified (all SNRs quoted within this thesis are calculated in this manner).

Ground truth spectrograms were created by generating a spectrogram for each parameter combination that have high SNRs (approximately 1,000 dB), and then thresholding these to obtain

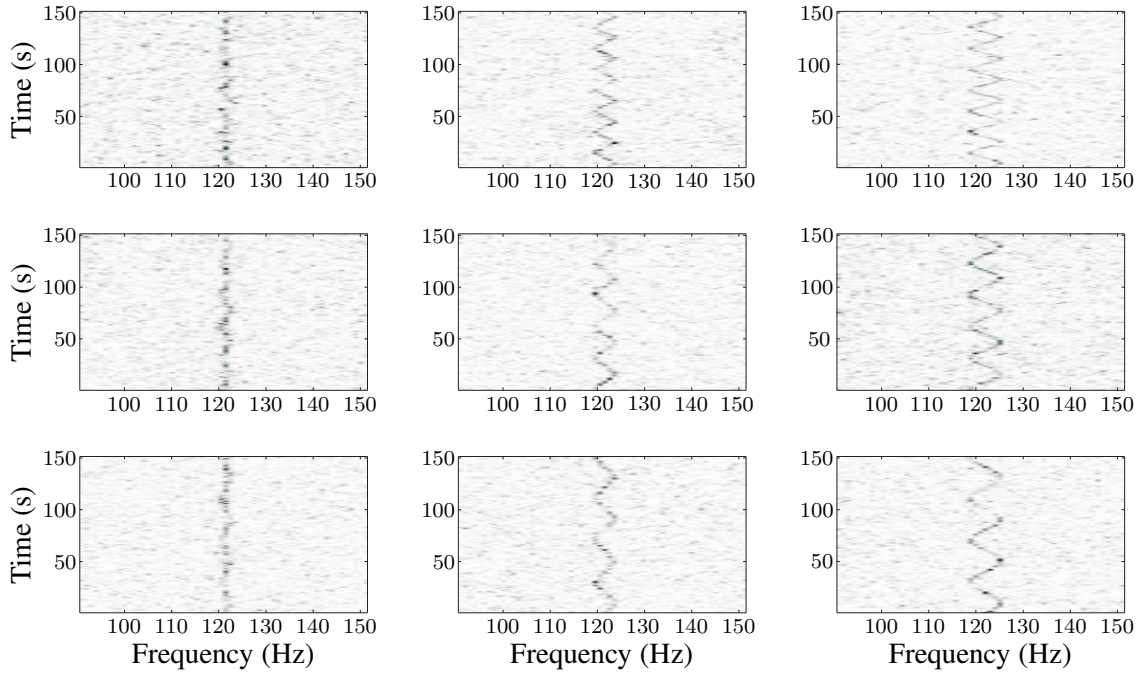


Figure 3.11: The effects of the parameter values upon the appearance of sinusoidal tracks. The top row represents sinusoidal tracks that have a 10 second period, the middle row a 15 second period and the bottom row a 20 second period. The tracks in the left column have a centre frequency variation of 1%, the middle 3% and the right 5%. All of the tracks have a mean SNR of 7.5 dB (rounded to the nearest 0.5 dB), which has been simulated for illustration purposes. The intensity is proportional to power in voltage-squared per unit bandwidth, that is V^2/Hz, and is linearly scaled to have intensity values between 0 and 255.

binary bitmaps. These have the value one in pixel locations where a track is present in the related spectrogram, and zero otherwise. The data set is scaled to have a maximum value of 255 using the maximum value found within the training set, except when applying the PCA detector, when the original spectrogram values are used.

3.3.2 Results

In this section are presented the results obtained during experimentation upon the data set described above. The parameters used for each method are described in Table 3.4 and the Gaussian classifier using PCA was trained using examples of straight-line tracks and noise.

The ROC curves were determined by varying a threshold parameter that operates on the output of each method—pixel values above the threshold were classified as signal and otherwise noise. The ROC curves for the Hough transforms were calculated by varying the parameter space peak detection threshold. The TPR and FPR for each of the methods were calculated using the number of correctly and incorrectly detected track and noise pixels.

Detection Method	Parameter	Value
Laplacian Convolution	Filter size (pixels)	3×3
	Threshold value range	0–255 (step 0.2)
Bar (fixed-scale)	width w (pixels)	1
	length l (pixels)	21
	angle θ (radians)	$-\frac{\pi}{2}$–0 (step 0.05)
	Threshold value range	0–255 (step 0.5)
Bar (muti-scale)	width w (pixels)	1
	length l (pixels)	$6, 7, 8, 9, 10, 12, 14, 16, 18$ & 20
	angle θ (radians)	$-\frac{\pi}{2}$–0 (step 0.05)
	Threshold value range	0–255 (step 0.5)
Pixel Thresholding	Threshold value range	0–255 (step 0.2)
PCA	Window size (pixels)	3×21
	Threshold value range	0–1 (step 0.001)
	Data dimensionality	2
Nayar	width w (pixels)	1
	length l (pixels)	21
	angle θ (degrees)	0–180 (step 0.05)
	Threshold value range (distance to manifold)	0–10 (step 0.1)
	Data dimensionality	8
MLE & MAP	λ	7.2764
	α	1.1439
	β	20.3073
co-MLE & co-MAP	Window size (pixels)	3×3
	λ	7.2764
	α	1.1439
	β	20.3073
Hough	Threshold value range (peak detection threshold)	0.5–1 (step 0.001)

Table 3.4: The parameter values of each detection method that were used during the experimentation.

3.3.2.1 Comparison of 'Optimal' Detection Methods

One of the hypotheses proposed by this chapter is as follows: as the amount of information made available to the detection process is increased, the detector's performance will also increase. Evidence for the validity of this hypothesis is presented in the form of performance measurements for each detector described in this chapter, each of which acts upon a different amount and type of information, which is presented in Fig. 3.12.

The MAP and ML detectors, operating on single pixel values, achieve a TPR of 0.051 and 0.643, and a FPR of 0.002 and 0.202 respectively (as no thresholding is performed ROC curves for these methods are not presented). These results highlight the high class distribution overlap and variability in this problem. The ML detector performs better than the MAP detector (although it also results in a higher FPR) due to the very low *a priori* probability of observing the track class—the detector requires a very high conditional probability for the decision to be made that the pixel belongs to the track class. These rates increase to a TPR of 0.283 and 0.489, and FPR of 0.016 and 0.074 when the MAP and ML detectors are evaluated within 3×3 pixel neighbourhoods (respectively). Again, the low *a priori* probability of the track class hinders the MAP detector's ability to detect tracks within the spectrograms as it does not reach the TPR level of ML detector on single pixels. Nevertheless, the MAP detector's TPR is increased when integrating spatial

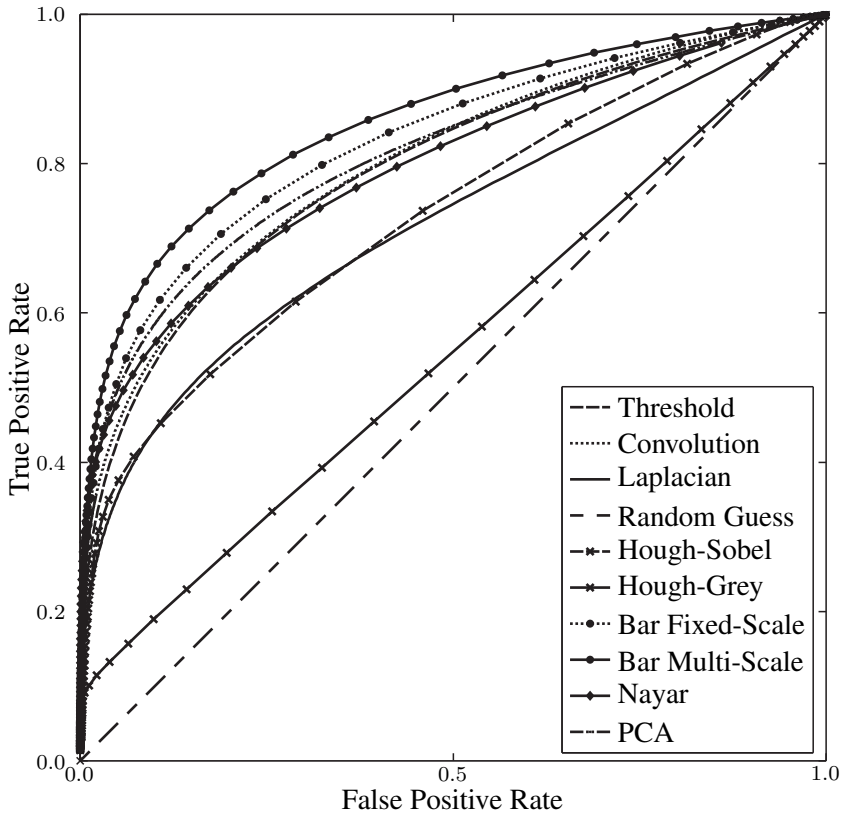
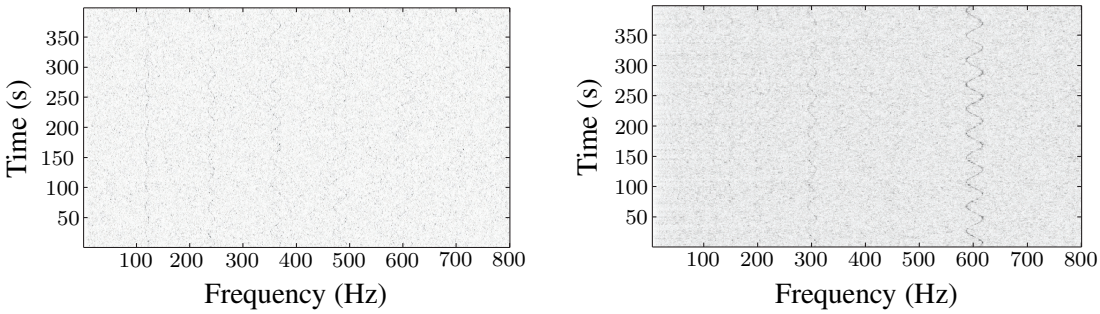


Figure 3.12: Receiver operating characteristic curves of the evaluated detection methods.

information (at the expense of a slight increase in FPR). Moreover, spatial integration has reduced the FPR of the ML detector quite dramatically, however, this is at the expense of a vast reduction of the TPR. Therefore, spatial integration does increase the detector's performance, however, due to the simplicity of the detection strategies, this increase is manifested in either a large reduction in the FPR or a large increase in the TPR, but not both. Finally, the bar detector was defined to exploit all of the information available to a detector: the intensity, local frequency, and structure of the pixel values. Preliminary tests were performed using a fixed length implementation. The maximum of the rotated bar's response, $\bar{B}(\theta_l, l, w)$, where $l = 21$ and θ_l, defined by Eq. (3.15) was taken as the output pixel's value to produce a response for each pixel. This was then thresholded to perform the detection and forms a feature detection mechanism that outperforms all other detection methods. The multi-scale abilities of the proposed method allow it to better fit piecewise linear features and approximate curvilinear features. These properties translate to a ROC curve that has greater separation from existing line detection methods than the fixed length implementation, and thus it achieves much higher TPRs and lower FPRs. Taking an example TPR of 0.7 the best detectors are, in order of increasing performance: Convolution (FPR: 0.246), PCA (FPR: 0.213), Bar Fixed-Scale (FPR: 0.181) and Bar Multi-Scale (FPR: 0.133). These results show that the combination of intensity information and structural information, rather than relying on intensity information alone, increases detector reliability.



(a) Original spectrogram.

(b) Result of the harmonic transform.

Figure 3.13: An example of the harmonic transform applied to a spectrogram. Intensity is proportional to power in voltage-squared per unit bandwidth, that is V^2/Hz.

3.3.2.2 Comparison of 'Sub-Optimal' Detection Methods

The second hypothesis proposed in this chapter was that 'optimal' detection methods will outperform 'sub-optimal' detection methods. It was found that the feature detector proposed by Nayar et al. and the fixed-scale bar detector would allow this comparison to be made, as they both utilise equivalent data models. It can be seen in Fig. 3.12 that the detection performance of the fixed-scale bar detector outperforms that proposed by Nayar et al. over the full range of TPRs and FPRs, confirming the validity of this hypothesis. It was found instead that the 'sub-optimal' detection method that achieves the closest performance to the bar-method was the Gaussian classifier using PCA. This indicates that the learning method is capturing the correct type of information in the data set and results in a form in which it is faithfully represented and modelled using the Gaussian distribution.

Of the other evaluated methods, the threshold and convolution methods achieve almost identical performance over the test set. With the Laplacian and Hough on Sobel line detection strategies achieving considerably less and the Hough on grey scale spectrogram performing the worst. It is possible that the Hough on edge transform outperformed the Hough on grey scale due to the reduction in noise occurring from the application of an edge detection operator. Nevertheless, both of these achieved detection rates that are considerably less than the other methods. None of the existing methods that were evaluated had comparable performance to the 'optimal' or 'sub-optimal' methods outlined in this chapter.

3.4 Harmonic Integration

An additional source of information that the detection process can exploit, other than local information as previously explored, arises from the harmonic nature of acoustic energy. Enhancing the detection robustness using this information was first explored by Barrett and McMahon [20], however, subsequent research has ignored this and instead has focussed on detecting individual tracks.

As described in Section 1.2.1 the acoustic signal emitted by a source comprises of a fundamen-

tal frequency and its harmonic series at frequencies that are integer multiples of the fundamental. Within a spectrogram these harmonic frequencies result in multiple tracks at specific positions. Recall that noise is an uncorrelated phenomenon and is therefore not harmonic in nature. A transformation can be defined upon the spectrogram, or output of a detector, that integrates the energy or detection from harmonically related positions, such that

$$s'_{yx} = \frac{1}{n}\sum_{k=1}^{n} s_{y,[kx]} \tag{3.25}$$

for $y = 1, 2, \ldots, M$ and $x = 1, 1\frac{1}{n}, 1\frac{1}{2n}, \ldots, N$ and where $[kx] \leq N$, the transformed spectrogram is $\boldsymbol{S}' = [s'_{yx}]_{M\times nN}$, the notation $[x]$ denotes the nearest integer function and the term n controls the number of harmonics that will be integrated in the transformation. The x-axis of the transformation output is related to fractional frequencies in the original spectrogram, this accounts for the frequency quantisation that occurs during the FFT process. Quantisation rounds fractional frequencies into the nearest quantisation bin and therefore the position of tracks harmonically related to a fundamental frequency may not reside in bins that are integer multiples of the fundamental frequency. An example of the output of this transformation when applied to a spectrogram is presented in Fig. 3.13. It results in a more prominent fundamental frequency, however, the transformation has actually decreased the spectrogram's SNR from 6.56 dB to 6.23 dB. The reason for this is concealed in the distribution statistics of the intensity values. The mean values of the two classes are transformed closer together—being 41.48 and 7.50 in the original spectrogram and 39.82 and 7.66 after the transformation (signal and noise respectively)—and the ratio between these forms the SNR estimate (Eq. 1.16). Nevertheless, the SNR estimate does not take into account the variance of the two classes and the transformation has a large effect on this. The standard deviations of the classes' intensity values in the original spectrogram are 25.50 and 7.55 and in the transformed spectrogram these values are roughly halved to 12.00 and 3.85—the transformation has reduced the overlap between the two classes, aiding in their separability.

3.4.1 Results

To demonstrate the effectiveness of this simple transformation, the previous experiment is repeated using the top performing detector, the multi-scale bar detector, and this is applied to the transformed spectrograms, $\boldsymbol{S}'$, as defined by Eq. 3.25 instead of the original spectrograms. As the harmonic set is integrated, the detector's performance is evaluated on the detection of the track corresponding to the fundamental frequency and not all the frequency tracks as in the previous experiment. The results of this experiment, in comparison to the detector's previous performance, are presented in Fig. 3.14 and they demonstrate the vast improvement in the detector's performance that is afforded by this relatively simple transformation.

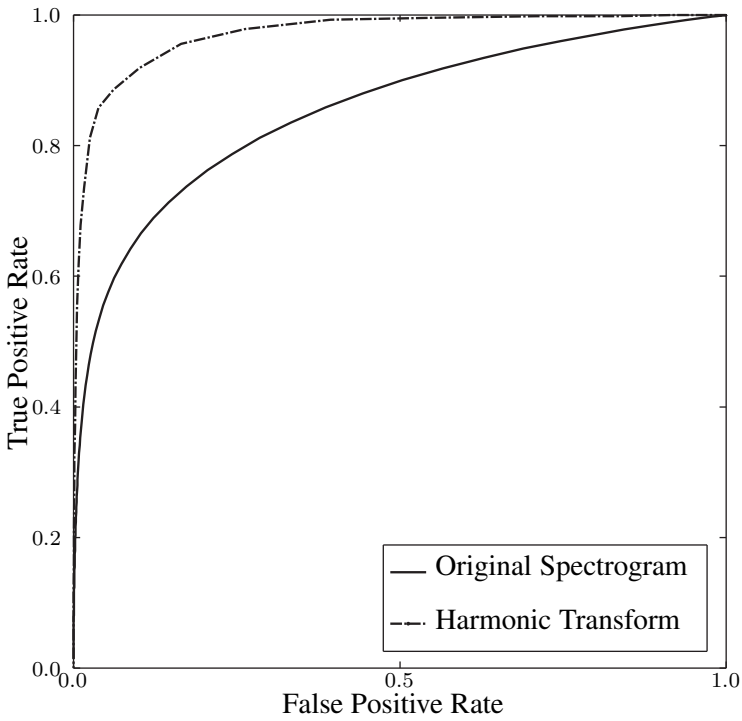


Figure 3.14: Receiver operating characteristic curves of the bar detector with and without harmonic integration

3.5 Summary

This chapter has presented a performance comparison within a group of novel and existing low-level feature detection methods applied to spectrogram track detection. Initially, a group of 'optimal' feature detectors were defined so that each utilised increasing amounts of information from the spectrogram when performing the detection and these were compared with each other. The information sources utilised by each of these were: the intensity of an individual pixel, the intensity distribution within a window, and the structural arrangement of pixels within a window. It was found that the 'bar' feature detector, which utilises the structural and intensity information from within a window (and therefore incorporates all of the available information), performed most favourably. Nevertheless, because of its exhaustive search, in combination with a complex model, it was found to be computationally expensive. A consequence of these findings is that the methods that are defined to operate on single pixel values, for example the solutions utilising the HMM, multi-stage decision process and simulated annealing, that were seen in the literature review cannot reach the performance of methods that utilise more information in the low-level detection process.

Subsequently, a group of 'sub-optimal' feature detectors were defined that utilise machine learning principles to simplify the detection process. These were also defined to utilise the maximum amount of information available to facilitate their comparison to the 'bar' detector and were grouped into the categories of model-based and data-based feature detectors; reflecting the source

of the training samples used by their supervised learning process. Due to the loss of information that is incurred by dimension reduction techniques these feature detectors were not able to perform comparably to the 'optimal' 'bar' detector. Nevertheless, a novel data-based feature detector that utilises principal component analysis was found to be the best performing 'sub-optimal' detector, in addition to reducing the computational complexity inherent in the 'bar' detector. This detector tackled the detection problem by specifically modelling the noise class, thus bypassing some of the generalisation limitations that are inherent when applying machine-learning techniques to limited training data (although the principal components are still dependent upon the track structure represented by the training set). Furthermore, a comparison between an 'optimal' and a 'sub-optimal' model-based feature detector, which have equivalent data models, found that the dimension reduction technique used in the 'sub-optimal' detector, whilst reducing computational complexity, vastly reduces detection abilities.

The final section of this chapter presented a harmonic transformation for spectrograms. This allowed for an empirical comparison between low-level feature detection with and without integrating information from harmonic locations. It was shown that the transformation does not increase the separation between the means of the track and noise classes but instead reduces the standard deviations of the classes—reducing the overlap between the distributions. This effect was shown to offer a vast performance improvement when detecting low-level features.

Chapter 4

A Track Detection Algorithm

"The field of computer vision has its sights set on nothing less than enabling computers to see."

— Blake and Isard [29].

It was shown in Chapter 3 that the detection of low-level features can be improved by exploiting structural information, in addition to intensity information, during the detection process. It has also been shown that, assuming a harmonic series is present, including information from harmonic positions can further improve detection rates. This chapter extends the active contour algorithm to create a model that incorporates these findings, forming a high-level track detection algorithm. The active contour model is a well known image analysis algorithm that achieves non-parametric feature detection within an image through energy minimisation. Several of its features, however, prevent its application to this problem and these are identified and overcome with novel solutions. Integrating a low-level feature detector, derived from machine learning and classification techniques, into a flexible track detection algorithm, that can model any structure (dependent upon its internal energy constraints), allows for generalisation to unseen track structure. Furthermore, this generalisation is enhanced by formulating the potential energy to be dependent upon noise class membership alone. The model extends the notion of a harmonic series to allow for the detection of defined patterns of narrow-band spectra—further enhancing detection at low SNRs. The proposed algorithm is subjected to an analysis of its computational complexity to ensure its suitability to the real-time applications that are the concern of this thesis.

The first section of this chapter introduces and presents the definition of the active contour algorithm and a review of the literature relevant to it, with respect to this problem. General limitations of the algorithm are identified and solutions present in the literature are discussed. The second section of this chapter discusses the limitations of the algorithm that prevent its application to this problem. Novel solutions are then proposed to overcome these limitations. These are presented in the context of the active contour framework for spectrogram track detection. A complexity analysis is then presented to demonstrate the framework's suitability to real-time implementation and finally the chapter's summary is drawn.

4.1 The Active Contour Algorithm

The Active Contour algorithm (also known as a snake) is a special form of deformable energy model proposed by Kass et al. [96] and allows for non-parametric feature detection within an image —ideal in problems, such as remote sensing, where *a priori* shape information is not strictly defined. The active contour is driven by an energy minimisation process and is constrained by internal energy forces, which ensure that its shape follows certain criteria; these criteria are typically defined as curvature and connectivity. It is guided by potential energy, which attracts the active contour towards features by following local changes in energy gradient. As these gradients are calculated on a local basis the contour needs to be initialised close to the desired feature to ensure correct convergence. The active contour converges on a minimum of the weighted combination of its internal and potential energies within the spatial domain of the image. The potential energy constraints translate this convergence to be a local gradient maxima in the image. In the original formulation the energy minimisation is performed using variational calculus. This model has been successfully applied to object detection and segmentation problems in a wide range of image analysis applications such as brain segmentation; artery, thyroid, cell and cortex detection; and road detection in space-borne SAR images.

The original active contour model, as proposed by Kass et al. [96], is as follows. A collection of k contour vertices defined on a finite grid, $\boldsymbol{v}(t) = [x(t), y(t)]$, $t \in \{0, 1, \ldots, k-1\}$, forms a deformable contour where $x(t)$ and $y(t)$ are the contour vertex's position in the image $\boldsymbol{S} = [s_{ij}]_{M \times N}$ such that $x(t) \in \{0, 1, \ldots, N-1\}$ and $y(t) \in \{0, 1, \ldots, M-1\}$. The contour has the energy

$$\mathcal{E}(\boldsymbol{v}) = \sum_{t=1}^{k-2} \Big(\alpha |\boldsymbol{v}'(t)|^2 + \beta |\boldsymbol{v}''(t)|^2 + \mathcal{P}(\boldsymbol{v}(t)) \Big) \tag{4.1}$$

where the terms α and β control the first-order continuity and second-order curvature of the contour respectively, and the term $\mathcal{P}$ is the potential induced by the image (also known as the external, or 'image' energy). The continuity is defined as the distance between two adjacent vertices, therefore ensuring that they remain close together, and the curvature as the second order distance, ensuring that the contour has low curvature to conform with the features of interest. Setting their weights, α and β, to have large values increases the influence of these energies.

The potential is defined to attract the contour to salient features in the image. The simplest features for this are the image intensity, such as

$$\mathcal{P}(\boldsymbol{v}(t)) = -\gamma s_{y(t),x(t)} \tag{4.2}$$

or gradient

$$\mathcal{P}(\boldsymbol{v}(t)) = -\gamma \big| \nabla s_{y(t),x(t)} \big|^2. \tag{4.3}$$

where ∇ is the gradient operator and γ is the potential's weight.

The active contour model is often applied to image analysis problems, which are inherently discrete. In such cases the first and second derivatives, which constitute the first and second terms in Equation (4.1), being the continuity and curvature internal energies, are discretely approximated

by finite differences [184]. The approximation of the first derivative is taken to be

$$\boldsymbol{v}'(t) \approx |\boldsymbol{v}(t) - \boldsymbol{v}(t-1)|^2 \tag{4.4}$$

and the second derivative is discretely approximated as

$$\boldsymbol{v}''(t) \approx |\boldsymbol{v}(t-1) - 2\boldsymbol{v}(t) + \boldsymbol{v}(t+1)|^2. \tag{4.5}$$

This poses a problem when implementing an open ended active contour as it is not possible to calculate these approximations at each end of the contour, $\boldsymbol{v}(0)$ and $\boldsymbol{v}(k)$ (as $\boldsymbol{v}(-1)$ and $\boldsymbol{v}(k+1)$ do not exist!). A simple solution is to 'mirror' the contour vertex on the opposite side of each of these vertices, such that $\boldsymbol{v}(k+1) = \boldsymbol{v}(k-1)$ and $\boldsymbol{v}(-1) = \boldsymbol{v}(1)$. This allows an approximation to their derivatives to be calculated in accordance with Equations (4.4) and (4.5).

4.1.1 Algorithm Background

The original formulation for the active contour algorithm, as described previously, was introduced by Kass et al. in 1988. Since its introduction the algorithm received a lot of attention from the image processing community as it allowed for flexible modelling of object boundaries, which forms a key step in object detection, recognition and segmentation problems, some of which have been mentioned above. Nevertheless, as the algorithm was investigated it became apparent that there were several limitations that needed to be overcome to allow successful application to these problems. The most prominent and relevant aspects of the algorithm will be discussed in this section, which will be organised according to the constituent parts of the algorithm and developments relevant to each of these will be discussed. These aspects are: contour initialisation, internal energy representation, potential energy representation, contour energy minimisation, and multiple contour models.

4.1.1.1 Contour Initialisation

One of the major limitations of the original algorithm is the contour's sensitivity to the initialisation location. For the contour to converge accurately it was necessary to initialise it close to the desired solution, which was often achieved through user intervention. A relaxation to this criteria was introduced by Neuenschwander et al. [133] in the form of perturbation snakes, a form of snake that only requires the two end-points of the object to be specified by the user. Alternative approaches were proposed with the aim of removing user intervention altogether; the first from Cohen [43]. This was termed the 'balloon' force and applied to closed contour formulations. It was noted that if the initial contour is placed in a constant gradient area—a part of the image with no potential force—the curve shrinks on itself until it becomes a singular point. To counteract this the balloon force acts as though filling the contour with air, expanding it from an initial state. If the contour is initialised somewhere within the object to be detected this balloon force expands the contour until it encounters the object's boundary. This relaxes the initialisation criteria from specific points on (or very near) the object boundary to any point within the object's boundary, and combats

the additional problem of the contour not being able to enter concave boundary sections. An incidental benefit of this force is that it introduces some resilience to noise in the image as the force pushes the contour over weak detections. Similar solutions to the balloon force exist: gravitation external force field [164], constant normal force [194], 'blown' force [187] and the 'wrapping' force [27]. Nevertheless, although some of these methods move towards a solution, they all suffer from the problem of striking a balancing between the strength of the additional force and that of the gradient induced by the potential energy. An imbalance between these can result in the contour over-running the object boundary. To combat this an auto-adaptive dynamic force is needed that will guide the contour towards the object [89]. An additional drawback of these methods is the need to determine in which direction the pressure should act. This is dependent upon whether the contour is initialised within or without the object boundary and this point is addressed with the dual contour. Dual contours [75] consist of one contour initialised inside and a second outside the boundary and their energies are minimised in absence of a balloon force. Once each of the contours has converged, a force is added to the contour with the highest energy, which attracts it towards the other. More recent developments, that aim to overcome the initialisation problem, have been proposed by Xu and Prince [192; 193], referred to as Gradient Vector Flow (GVF), and subsequently, Normal Gradient Vector Flow (NGVF) proposed by Jifeng et al. [91]. The initialisation problem, as discussed, can be thought of as a limitation resulting from the potential energy force. The extent to which this force can influence the contour is limited to a local region surrounding the object boundary. Gradient vector flow overcomes this limitation by calculating a diffusion field of the gradient vectors derived from the image. The field then extends far away from the boundary of the image and is independent of whether the contour is initialised within or outside the object boundary. The GVF also has the effect of overcoming the boundary concavity [50] problem. Nevertheless, the method does have its limitations; the calculation cannot be formulated using the standard energy minimisation framework, instead, it is specified directly from a force balance equation.

4.1.1.2 Potential Energy

Other than the local influence of the potential energy, limitations still exist with applications in which features are not defined, or reliably defined, by gradient change or intensity [37]. If the image is too noisy then a large amount of smoothing is required, which will smooth the boundary edges. Complex backgrounds are likely to produce strong edges, which can be mistaken as object boundaries [164]. Alternative potential energies have been proposed to overcome these limitations; Davis et al. [52] combine the output of several edge detectors, including: the Sobel detector; the Canny detector; maximum likelihood detection; the Mero-Vassey detector [126]; and a bi-directional morphological edge operator. The solution improves edge detection and is applied to computerised tomography and magnetic resonance medical images, however, the increase in the number of parameters complicates the active contour model and its parameter selection. Wu et al. [188] use a single potential energy and propose using the zero-crossings of wavelet-frames to offer noise resistance. It is concluded in the paper that the solution is effective down to an SNR of 10 dB, and below this the performance deteriorates. Davatzikos and Prince [51] propose an energy based

upon the probability distribution within a region of specific thickness. Minimising the variance of the region draws the contour towards thick, homogeneous boundaries in the image. Shih and Zhang [164] combine the original energy term with a regional similarity energy that minimises the difference between the intensity of the region encompassed by the contour and the intensity within the contour vertex's local neighbourhood and apply this to locate objects in complex backgrounds. Similar approaches, termed region-snakes, have been taken by Chesnaud et al. [41] and Slabaugh et al. [166], however, as the name implies, these are methods designed to segment regions of an image and therefore tackle the problem of object extraction from a different viewpoint. An additional model that has been proposed with respect to image segmentation is called the Active Contour Without Edges (ACWE) [37]. In this model two regions are defined as 'inside' and 'outside' of a closed contour and the mean intensity values of these regions are calculated. The difference between their average and the expected region intensity is minimised, and this minimum translates to the detection of an image segment having different image intensity. Savelonas et al. [159] point out that, with respect to texture segmentation, this formulation cannot discriminate regions of different textures that have equal average intensity values. To address this the ACWE model has been extended to use vector-valued images [36], these may be separate RGB images, where each pixel is a vector comprising red, green and blue components, or multi-spectral images where each pixel is a vector of components representing different wavelengths. Alternatively, vector-valued images can be formed by deriving features from a Gabor transformation [135, 158, 156], Wavelet transformation [16] or LBP distributions [159]. These region based active contours use information regarding a region to define the contour's convergence and therefore are not applicable in this research.

4.1.1.3 Internal Energy

The effect of the contour collapsing upon itself in the absence of potential energy is a side effect of the internal energy formulation. Although these energies force the contour to form a smooth shape and to have vertices that are in close proximity to each other, they also force the contour to collapse into a point [184]. This is a side effect of the continuity energy as it minimises the distance between each vertex and therefore forcing the contour to coalesce upon strong edges in the image [9]. Williams and Shah [184] move towards addressing this issue by exploring better approximations to curvature. It is noted in the paper that the original internal energies are not normalised with respect to distance, although the value of the energy's weight can be chosen to correct this invariance. A curvature estimate is proposed based upon the mathematical definition of curvature (the angle between the x-axis and the tangent to the curve), however, this measure is computationally expensive and is scale variant. Two additional measures are proposed, which are based upon the original curvature measure. Squaring the curvature's value forms a measure that is dependent upon the distance between vertices and results in a curvature measure can be non-zero when the contour is straight. Normalising the two vectors before taking the difference removes this the length dependence and, consequently, the curvature measure is based solely upon the angle between the vectors. Perrin and Smith [141] argue that a contour that is a smooth circle and has contour vertices equally spaced, fulfils the original hypothesis for the internal energy and therefore

should be a minimum of the internal energy functions. An internal energy based upon geometric properties is defined in the paper that fulfils this specification. Furthermore, this representation of curvature combines the continuity and curvature energies into one term, simplifying the internal energy's computation in addition to the active contour model.

4.1.1.4 Energy Minimisation

Within the original formulation proposed by Kass et al. energy minimisation is achieved using a variational approach by solving a pair of Euler equations iteratively. As such, it requires that the energies are defined by differentiable functions and Finite Differences or Finite Elements are used to discretely approximate the continuous energy functions. This requirement does not allow hard constraints to be defined upon the contour, such as imposing a fixed minimum distance between the vertices [10]. Additionally, this method suffers from numerical instability due to the compound effects of the iterative approximations followed by a further geometrical approximation from the continuous plane, in which the optimisation is performed, to the discrete grid of the image [9]. Search-based approaches to energy minimisation which allow the inclusion of hard constraints, such as simulated annealing [73] and dynamic programming [9], have been proposed. These techniques solve the energy minimisation iteratively, however, the simulated-annealing approach is supervised and, as such, requires user intervention to identify the contour's end points. Furthermore, these techniques are computationally expensive as they perform exhaustive searches within the search space [89] and add additional complications such as choosing the correct parameter combination [75]. An additional iterative algorithm derived from the dynamic programming method [89] that allows the inclusion of hard constraints and avoids exhaustive searches is proposed by Williams and Shah [184]—the greedy algorithm. The authors have shown that this greedy algorithm produces comparable results to the more complex methods, in addition to which, it is much faster than the dynamic programming method proposed by Amini et al. [9], having a complexity of $O(nm)$ as opposed to $O(nm^3)$, where n is the number of contour vertices and m is the number of points in each contour vertex's neighbourhood. The complexity has been further reduced to form the fast greedy algorithm [107], which reduces the computation time by 30% by evaluating the energy function in alternative search patterns within each vertex's neighbourhood. Further improvements have been realised in the optimally fast greedy algorithm [127]. More recently, genetic snakes have been proposed that use a genetic algorithm to minimise the energy function [17]. Using a genetic algorithm in this setting proved to be useful in overcoming the problems of initialisation, parameter selection and local minima in the energy function. In the case that points on the object boundary can be supplied to the algorithm, alternative energy minimisation approaches can be taken and, because of the restricted space in which they work, these techniques may lead to globally optimal solutions. The minimal path approach [44] is one such method that guarantees to find the global minimum of the energy function and requires only one point on the object boundary to be identified.

4.1.1.5 Multiple Contours

An extension to the active contour model that allows for the simultaneous detection of multiple objects within an image has also been presented in the literature. This approach can offer improved detection rates in a number of different applications, in particular when multiple objects that have similar appearance exist within a single image. For this purpose, Srinark and Kambhamettu [167] propose a framework that contains an additional energy term called the group energy. This energy measures the variance of properties of the objects being detected and is used to promote the correct detection of 'weak' objects (i.e. poorly defined in the image) by the detection of 'strong' objects and, as such, the objects must be of the same shape. An alternative multiple snake formulation is proposed by Chalana et al. that utilises two snakes to extract cardiac boundaries within ultrasound images [35]. In this model, however, the structure that one contour can model is independent of the other. A further class of active contour model that utilises multiple contours has been developed to detect a single object using multiple contours [1, 199], however, these are applied to detecting regions and not boundaries and are therefore not applicable here.

This review has concentrated on aspects relevant to the problem posed by this thesis and as such several unconnected developments regarding the active contour model, have been omitted. For example, extensions relevant to three-dimensional object detection [42, 81], video analysis [136], Active Shape Models [45] and Geodesic Active Contours [34]—to name but a few.

4.2 Track Detection Framework

For the detection of features in the proposed domain two of the issues discussed previously limit the active contour's application (in its original form): its sensitivity to initialisation and the assumption that features are defined by local intensity gradient. These limitations are addressed in this section.

It is the concern of this thesis to detect vertical curvilinear features within spectrogram images. Therefore the active contour model described here is an open ended contour where the first and last vertices are fixed to the top and bottom of the spectrogram (or window within the spectrogram) such that $\boldsymbol{v}(0) = [x(0), \rho]$ and $\boldsymbol{v}(k-1) = [x(k-1), M-\rho]$, where ρ is the height dimension of the potential energy defined below. Movement of the contour vertices is restricted in the y-axis to ensure an even search along this axis. This has the additional benefit of reducing problems that can occur due to the internal energy's sensitivity to the distance between contour vertices [157] and avoids the need to dynamically resample the contour by preventing the vertices from becoming too disparate.

4.2.1 Gradient Potential

Since its introduction it has been evident that the active contour model is limited by its sensitivity to the initialisation location and it is therefore required to be initialised close to the desired feature to ensure correct convergence. This is an effect of the potential energy gradients being calculated on a local basis and is overcome by Cohen in the closed contour case by introducing a balloon

force that expands the contour, allowing it to be initialised anywhere within the feature boundary. In the proposed model a similar force, the gradient potential $\mathcal{W}$, is implemented, such that

$$\mathcal{W}(\boldsymbol{v}(t)) = \begin{bmatrix} -c \\ 0 \end{bmatrix} \boldsymbol{v}(t). \tag{4.6}$$

This creates an uniform energy gradient across the spectrogram, and therefore across the contour's search space, biasing the contour to move in a given direction with force c. In effect, forcing the contour to perform an even search throughout the spectrogram after being initialised at a low frequency. As previously discussed with respect to the balloon force, the gradient potential also prevents the contour from being trapped by spurious isolated edge points, allows it to pass over weak edges [164] and reduces the contour's sensitivity to its initial configuration [42]. Additionally, this supplementary force enables the contour to move into concave sections of the track [89].

4.2.2 Potential Energy

Chan and Vese [37] state that a general edge-detector can be defined by a positive and decreasing function g, that is dependent upon the intensity gradient of an image, such that

$$\lim_{z \to \inf} g(z) = 0. \tag{4.7}$$

The original potential energy function fulfils this condition but it has been shown in Chapter 3 that, with respect for the application, where low SNRs are commonly encountered, the performance of a simple edge detector such as that expressed in the original potential energy function is insufficient. A problem that has also been encountered by Chalana et al. [35] with application to cardiac boundary detection.

It was also shown in Chapter 3 that a detection mechanism can be defined using machine learning techniques to exploit more of the information that is available in the spectrogram. Such a feature detector combines intensity information with spatial information to allow for detection in low SNR images and along broken (weak) tracks. In addition to the property defined by Chan and Vese, in this application it is required that the detection mechanism has a low computational burden and it has been shown in Chapter 3 that exhaustive 'optimal' detectors do not have this property. Nevertheless, a detector has been defined with low computational burden, which performs favourably in comparison with the 'optimal' detector and also has the property of a general edge-detector defined by Eq. 4.7.

These properties were fulfilled by a feature detector that will now form the potential energy of the active contour. The potential energy is therefore derived from intensity values taken within a $\xi \times \rho$ pixel window $\boldsymbol{W}_{ij}$, centred on vertex $\boldsymbol{v}(t) = [x(t), y(t)]$ where $i = x(t)$ and $j = y(t)$ using Eq. (3.6). The method of spectrogram windowing is outlined in Section 3.1.2.1. Here, the windows are taken dynamically under each vertex of the contour as it evolves within the spectrogram. The intensity values are arranged column wise into a vector $\boldsymbol{V}^{ij}$ using Eq. (3.8) and PCA is utilised to derive a compact feature vector that represents the window (avoiding the 'curse of dimensionality'

[28]). Its similarity to noise can be measured by testing its membership of a noise model. To allow its use as a potential energy, the measure has to be formulated to take a maximum value when the window contains a signal track and minimum value when the window contains noise.

A multivariate Gaussian distribution is used as the noise model and this is fitted to examples of the noise class within a space defined by d principal component basis vectors, such that

$$G(\boldsymbol{v}(t)) \quad = \quad \frac{1}{(2\pi)^{\frac{d}{2}}|\hat{\boldsymbol{\Sigma}}^n|^{\frac{1}{2}}} e^{-\frac{1}{2}\hat{\boldsymbol{Q}}^T(\hat{\boldsymbol{\Sigma}}^n)^{-1}\hat{\boldsymbol{Q}}} \tag{4.8}$$

for $\hat{\boldsymbol{Q}} = \boldsymbol{U}^T\dot{\boldsymbol{V}}_l^{ij}$, where $\hat{\boldsymbol{\Sigma}}^n$ is the standard deviation of the low-dimensional noise cluster, and $\dot{\boldsymbol{V}}_l^{ij}$ is the feature vector after removing the high-dimensional noise cluster's mean (both of these are determined during a training phase). When subtracting the mean it is necessary to observe the following condition; if a component of the vector is zero, its corresponding value in the mean removed vector is also zero. This enables the contour to effectively 'ignore' previously detected pixels by setting their values to zero, a condition that is also physically meaningful—if no power is present in the pixel, no signal can be contributing to its value. Therefore, during the training phase the noise cluster will be centred on the origin of the low-dimensional space by subtracting the mean of the noise cluster in the high-dimensional space. The vector $\dot{\boldsymbol{V}}^{ij}$ is therefore calculated such that

$$\dot{\boldsymbol{V}}_l^{ij} = \begin{cases} 0 & \text{if } \boldsymbol{V}_l^{ij} = 0, \\ \boldsymbol{V}_l^{ij} - \boldsymbol{\mu}_l^n & \text{otherwise} \end{cases} \tag{4.9}$$

where $\boldsymbol{V}_l^{ij}$ is the lth component of the vector $\boldsymbol{V}^{ij}$ and $\boldsymbol{\mu}_l^n$ is the lth component of the vector $\boldsymbol{\mu}^n$.

4.2.2.1 Noise Model Training

The parameters $\boldsymbol{U}$, $\boldsymbol{\mu}^n$ and $\hat{\boldsymbol{\Sigma}}^n$ are determined during a training phase and are then stored for use during the algorithm's application. First it is necessary to derive the subspace spanned by the orthogonal basis vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d$ in which the noise model will be defined, where d is the rank of the matrix $\boldsymbol{U}$. The bases for the new space are derived through unsupervised learning using PCA [28] and therefore a training set $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j]^T$ where $\boldsymbol{x}_l = \boldsymbol{V}^{ij}$ is formed such that it includes equal numbers of examples of windows containing a feature and those containing only noise. The principal components $\boldsymbol{u}_k$ of this training set, are found by maximising the quantity λ_k, such that

$$\lambda_k = \frac{1}{n}\sum_{i=1}^{j} \boldsymbol{u}_k^T(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^2$$

where $\bar{\boldsymbol{x}}$ is the mean vector of $\boldsymbol{X}$, subject to the orthogonality constraint

$$\boldsymbol{u}_l^T\boldsymbol{U}_k = \delta_{lk}$$

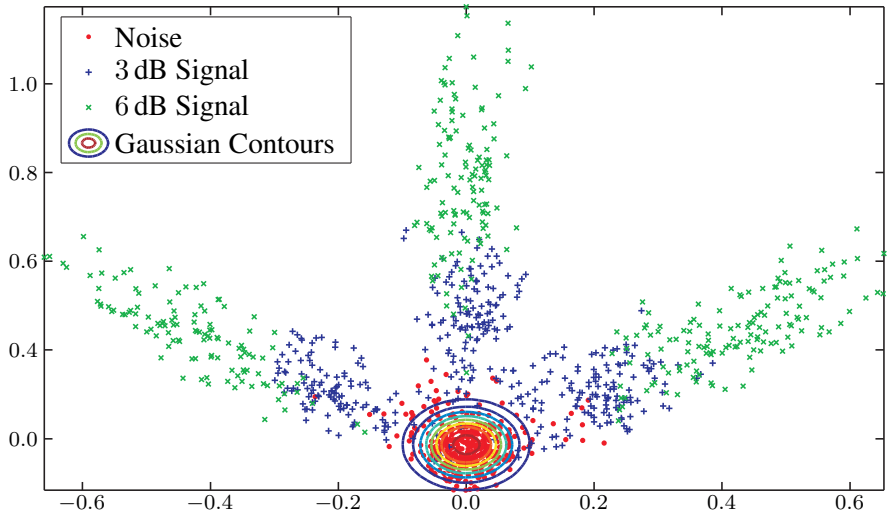


Figure 4.1: Windowed spectrogram feature vectors projected onto the first two principal components (window size 3×21 pixels). The noise class is represented by red circular points, the two signal classes, having an SNR of 3 and 6 dB, are represented by blue and green crosses respectively and the contours represent the Gaussian distribution. Increasing the track's SNR increases its distance from the noise class. The three pronged fan structure results from the track being present in the left, middle or right of the window.

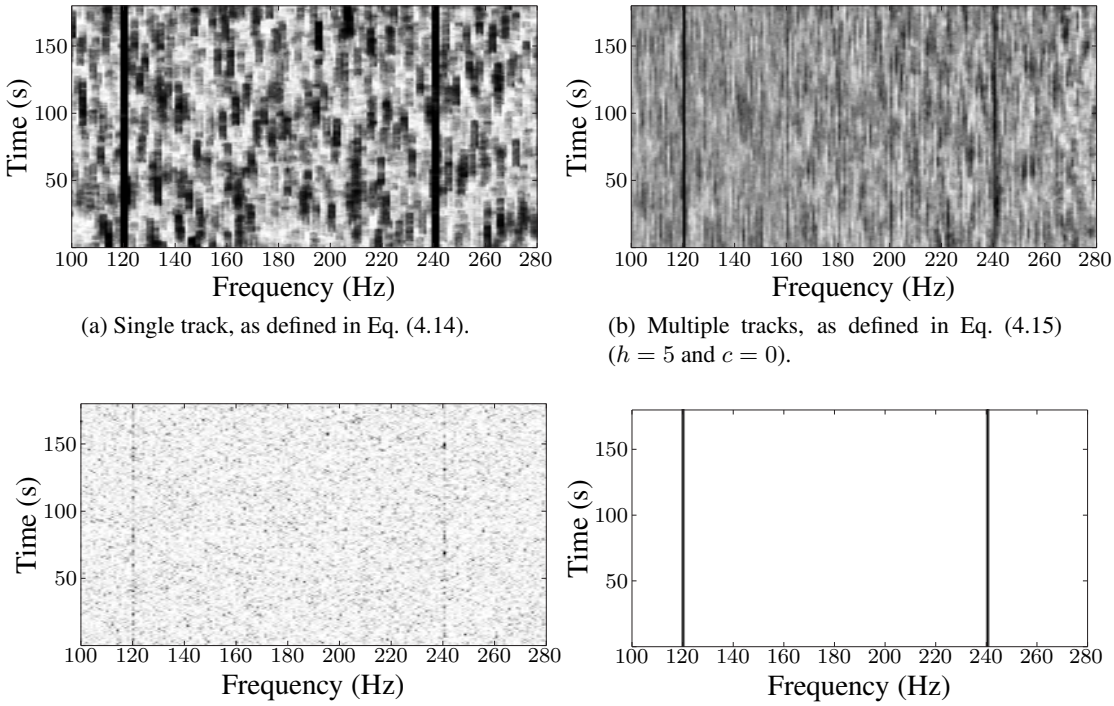
where δ_{lk} is the Kroneker delta. A subset of the basis vectors, $\boldsymbol{U}$, is selected as the first d principal components to form the low dimensional space, such that

$$\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d]. \tag{4.10}$$

By storing the basis vectors, the window vectors derived as the active contour evolves can be projected into the same space. Projecting the training set onto these basis vectors results in a distribution similar to that presented in Fig. 4.1, in which there is a clear separation of the classes and in which the noise is modelled using the Gaussian distribution. A consequence of the dimensionality reduction process is that the number of basis vectors used to model the subspace (the space's dimensionality) is a parameter to be determined. In the case of PCA the dimensionality should reflect the proportion of the training data's variance that is to be represented and the fraction of training data available to the algorithm's training process. Consequently, the number of dimensions to be used is dependent upon the training data used to derive the principal components. Therefore the value of d that is applicable to the proposed application and data will be determined during the algorithm's training process.

Now that the space in which the noise model is to be defined has been derived, the model can be fitted to the data. The noise samples are extracted from the training set $\boldsymbol{X}$, such that $\boldsymbol{X}^n \subset \boldsymbol{X}$ where $\boldsymbol{X}^n \neq \emptyset$, and their mean is calculated in the original high-dimensional space, such that

$$\boldsymbol{\mu}^n = \frac{1}{l} \sum_{i=1}^{l} \boldsymbol{x}_i^n \tag{4.11}$$



(a) Single track, as defined in Eq. (4.14).

(b) Multiple tracks, as defined in Eq. (4.15) ($h = 5$ and $c = 0$).

(c) Original spectrogram.

(d) Ground truth.

Figure 4.2: Potential energy topologies for a 180×180 pixel section of a spectrogram. The x-axis represents frequency, the y-axis time and, in the original spectrogram, intensity is proportional to power in voltage-squared per unit bandwidth, that is V^2/Hz. For easier interpretation the values in (a) and (b) are $1 - E(\boldsymbol{v}(t))$, making the valleys peaks and vice versa. A window size of 3×21 pixels was used to generate this data.

where l is the number of vectors within the set $\boldsymbol{X}^n$. This mean is removed from the training set to form $\boldsymbol{X}_c^n = \boldsymbol{X}^n - \boldsymbol{\mu}^n$. The term $\boldsymbol{\Sigma}$ is defined as the standard deviation of the noise cluster within the low dimensional subspace. The mean centred noise cluster is projected into the low dimensional space such that $\hat{\boldsymbol{X}}_c^n = \boldsymbol{U}^T \boldsymbol{X}_c^n$ and therefore $\hat{\boldsymbol{\Sigma}}^n$ can be calculated using the maximum likelihood estimate [60], such that

$$\hat{\boldsymbol{\Sigma}}^n = \frac{1}{l} \sum_{i=1}^{l} (\hat{\boldsymbol{x}}_i^n - \hat{\boldsymbol{\mu}}^n)(\hat{\boldsymbol{x}}_i^n - \hat{\boldsymbol{\mu}}^n)^T \tag{4.12}$$

where

$$\hat{\boldsymbol{\mu}}^n = \frac{1}{l} \sum_{i=1}^{l} \hat{\boldsymbol{x}}_i^n \tag{4.13}$$

and where $\forall \hat{\boldsymbol{x}}^n \in \hat{\boldsymbol{X}}_c^n$ is a vector containing only noise and l is the number of such vectors within the set $\hat{\boldsymbol{X}}_c^n$. The contours of level response resulting from such a model are depicted in Fig. 4.1.

4.2.2.2 Individual Track Detection

The noise model's response can be combined with the gradient potential that has been defined in Section 4.2.1 to replace the original potential energy, P, in the energy formulation of the active

contour model, Eq. (4.1), such that

$$P(\boldsymbol{v}(t)) = \mathcal{W}(\boldsymbol{v}(t)) + \gamma G(\boldsymbol{v}(t)) \tag{4.14}$$

where γ is the potential energy's weighting.

The feature space topology resulting from Eq. (4.14) is similar to that presented in Fig. 4.2a, demonstrating that the combination of spatial and intensity information produces few spurious detections and a large gradient change at track locations—desirable properties for feature detection using the active contour algorithm. Nevertheless, if each simultaneous track is a component originating from a common source, and therefore is part of a signature pattern, the local nature of the energy term results in multiple detections, one for each component, and not a single detection for the whole pattern. In addition to this, some of the false positive detections have the same magnitude as true positives and in short time frames these are hard to separate with a simple threshold.

4.2.2.3 Multiple Track Detection

To overcome these issues when performing low-level feature detection it has been necessary to define the harmonic transformation that was described in Section 3.4. This transformation integrates information from harmonic locations in the spectrogram to form a single, more distinguished, track in the output and it also has the effect of suppressing false positive detections. A similar transformation will be integrated into the potential energy of the active contour to alleviate the issues found in the single track formulation. The transformation will be extended to integrate information from locations defined by the characteristics of the target to be detected and as such the harmonic transformation previously defined is a special case of this in which the track relationships are defined by integer multiples.

It was discussed in Chapter 1 that simultaneous tracks originating from a common source can have some underlying linear relationship, for example, periodic signals are made up of harmonic frequencies and produce tracks in a spectrogram at harmonic locations. This *a priori* knowledge can be represented by a pattern set $\boldsymbol{P}_s = \{m_1, \ldots, m_h\}$, where $m_i \in \mathbb{R}^+$ is a multiple of the fundamental frequency, and can be integrated into the potential energy function, Eq. (4.14), such that

$$P(\boldsymbol{v}(t)) = \mathcal{W}(\boldsymbol{v}(t)) + \frac{\gamma}{h}\left[\sum_{i=1}^{h} G\left(\begin{bmatrix} m_i & 0 \\ 0 & 1 \end{bmatrix} \boldsymbol{v}(t)\right)\right] \tag{4.15}$$

where $m_1 = 1$ (the fundamental frequency) and the term $h \geq 1$ is the number of relative frequencies in $\boldsymbol{P}_s$. Window samples in Eq. (4.15) are taken from relative locations as defined in $\boldsymbol{P}_s$ and the potential energy forms a pattern-based active contour search—an active 'mesh' (Fig. 4.3) that stretches under the influence of the harmonic relationships as its fundamental position increases in frequency.

The modified feature space incorporating harmonic information is similar to that presented in Fig. 4.2b. The averaging of detections in several locations vastly reduces the unwanted effects of

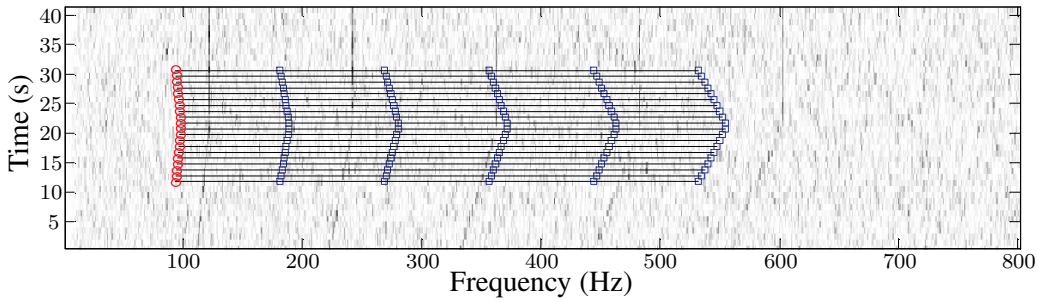


Figure 4.3: The contour mesh, the contour 'body' in circles, its harmonic set locations defined by $\boldsymbol{P}_s$ in squares and lines depicting the connection of potential energy.

the energy term defined in Eq. (4.14). In particular, a track's response is now located at a single, more localised, position corresponding to the fundamental frequency of the signature pattern and this is easily distinguished from the weaker harmonic response. Gaps in the track, a result of weaker signal sections, have been interpolated with information from higher harmonics and false positive detections are weaker due to the random, uncorrelated, nature of noise. In this example, these are now easily distinguished from true detections in short time periods.

A final point regarding the potential energy; it was outlined in the literature review presented at the start of this chapter that some formulations of the active contour model require that the potential energy is differentiable. For example, this is the case when variational calculus is used to minimise the contour's energy. Although it has not been presented as such within this thesis, the potential energy force can be utilised as a separate transformation stage and applied to the spectrogram prior to the active contour. In this case the original potential energy (the pixel's intensity value) can be utilised and therefore any minimisation technique that is applicable in the original formulation is also applicable in this case—thus it is equivalent. This bypasses the need for a differentiable potential energy, however, in this thesis the greedy energy minimisation technique will be used and therefore the condition of differentiable energies is not necessary.

4.2.2.4 Noise Model

A single Gaussian distribution is used to model noise in the proposed algorithm. Noise excursions above a certain threshold are classified as 'signal'. In a supervised learning situation this improves generalisation to unseen cases and is a useful property with respect to the concerned application, and any that has similar characteristics, in which the feature's appearance, for example the track's shape, can vary significantly against a background of uniform noise. Consequently, it may be necessary to model noise with different characteristics, however, the formulation of this algorithm facilitates this. In a similar vein to the generalisation made by Chan and Vese regarding the edge detector, it is possible to state that any noise model that can be defined as a positive increasing function and one that is dependent upon the noise in a sample, can be adopted in the proposed algorithm. In fact, all probabilistic models fulfil this criterion and have the additional benefit that their outputs do not require scaling to fall within the unit interval. Furthermore, this generalisation permits the inclusion of a wider range of classification methods such as those explored in the

previous chapter.

In situations where the track class is well defined, i.e. where it has little variability, the problem allows for the modelling of the track class. It is therefore possible to augment the noise model with information derived from the track model. Under this formulation, where s is a positive and increasing function of membership to the track class, the term G in Eq. 4.15 should be supplemented such that

$$G(z) = 1 - s(z). \tag{4.16}$$

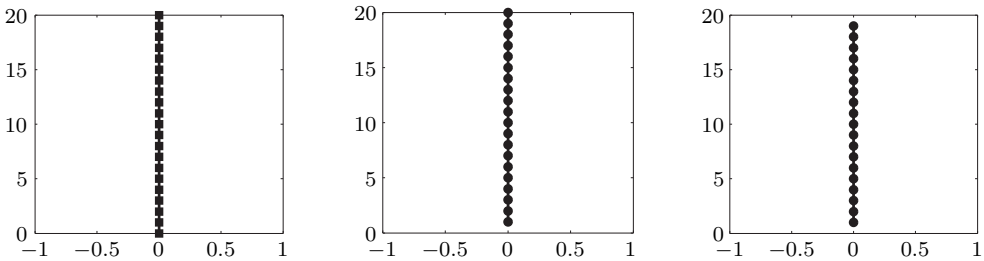
Some problems may exhibit temporal variability of the noise distribution—a characteristic that is not explicitly accounted for with the proposed noise model. Nevertheless, an extension to explicitly model temporal variation of the noise's mean intensity can be incorporated into the proposed model. This aspect is not fully explored in this thesis but two solutions to the problem will be briefly presented here. The first solution achieves this invariance by removing the mean from the window vectors prior to processing, similar to the technique of achieving lighting invariance in photometric image analysis [71], which results in vectors that model each pixel's deviation from the their mean and not their absolute intensity values. The second is a more complex method that exploits the sparseness of frequency tracks in a spectrogram; in most cases a source will not be present in a spectrogram, however, in the case where there exists a source that emits say ten narrowband frequency components and the spectrogram is the result of a $1{,}000$ frequency bin FFT process, the track class represents only 1% of the data. In this context the frequency tracks have a very small skew on the maximum-likelihood estimate of the data's mean and can therefore be ignored or accounted for through bias estimation [60]. As such, the mean of the noise model can be updated at each time step to be equal to the mean value of the current spectrogram frame.

4.2.3 Internal Energy

The internal energies for the active contour model are defined by the first and second derivatives along the contour, Eq. (4.1). These energies force the contour vertices to remain close to each other, a condition enforced by the first derivative, and to have low curvature, enforced by the second derivative constraint. Williams and Shah demonstrate that for a closed contour under no influence from a potential energy, these internal energies force the contour to collapse into a point.

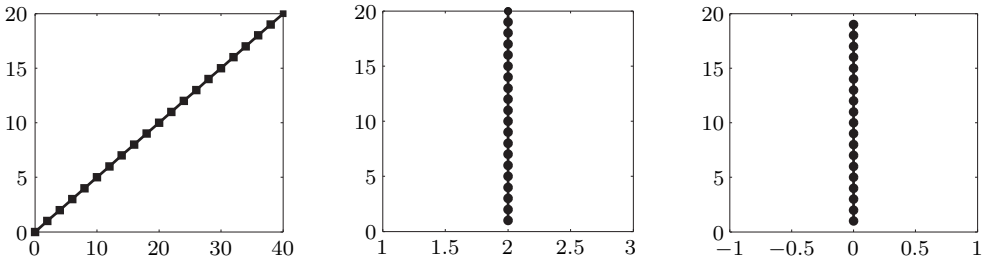
In the case of an open-ended contour, these energies force the contour to have a straight, vertical configuration. With reference to Figs. 4.4, 4.5 and 4.6; an analysis of the internal energy values under three, ideal, configurations demonstrate this effect. The three cases under investigation are: configurations in which the contour is vertical; oblique; and sinusoidal. It can be seen that it is in only one of these cases that the sum of the internal energies is minimum (the case in which the contour is vertical). The consequence of this is that the internal energies bias the contour to form a vertical configuration when detecting tracks that have any of the other configurations.

Perrin and Smith [141] alleviate this problem in the closed contour case by defining an internal energy based upon local geometric properties of the neighbouring contour vertices. The energy is calculated as the distance from the current contour vertex position to the point on the perpendicular bisector of the two surrounding vertices that has an exterior angle equal to the mean of



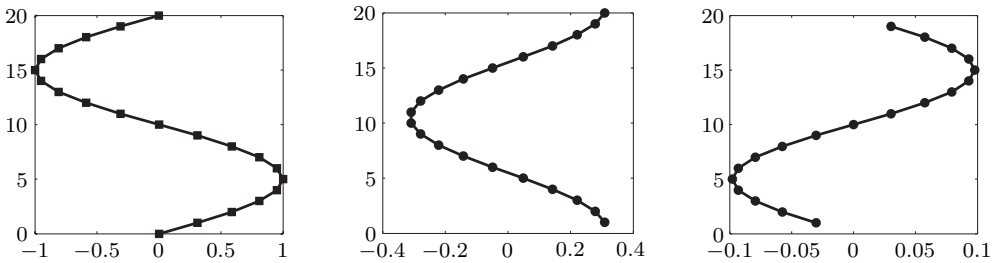
(a) Vertical Track. (b) First Derivative. (c) Second Derivative.

Figure 4.4: The original internal energies' values when modelling a straight vertical track.



(a) Oblique Track. (b) First Derivative. (c) Second Derivative.

Figure 4.5: The original internal energies' values when modelling an oblique track.



(a) Sinusoidal Track. (b) First Derivative. (c) Second Derivative.

Figure 4.6: The original internal energies' values when modelling a sinusoidal track.

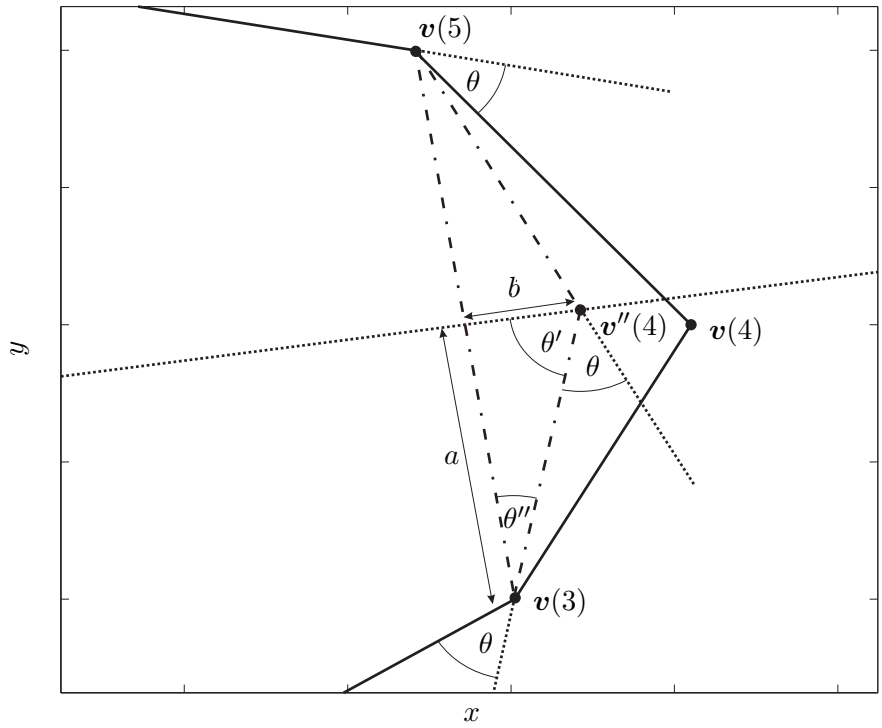


Figure 4.7: An illustration of the optimal contour vertex position as defined by the internal energy introduced by Perrin and Smith.

the exterior angles between the five neighbouring vertices. In their paper, however, the solution is presented in a descriptive manner and not mathematically or algorithmically. In this section this point is addressed and a mathematical formulation of the energy is derived. This problem can be formulated as calculating the lengths of sides a and b of an isosceles triangle, see Figure 4.7, and can be solved through simple geometrical properties as follows.

The length of the base of the isosceles triangle is $\|\boldsymbol{v}(t-1) - \boldsymbol{v}(t+1)\|$ and the two equal length sides have lengths $\|\boldsymbol{v}(t-1) - \boldsymbol{v}'(t)\| = \|\boldsymbol{v}(t+1) - \boldsymbol{v}'(t)\|$. The case in which the base of the triangle is parallel to the x-axis will be considered first and then this will be generalised to the case in which the triangle is arbitrarily rotated. The ideal vertex position, $\boldsymbol{v}'(t)$, for $\boldsymbol{v}(t)$ is at $\boldsymbol{v}(t-1) + [a\ b]^T$ where a and b are equal to half the length of the isosceles triangle's base and its height, respectively. The component a is therefore calculated such that

$$a = \frac{1}{2}\|\boldsymbol{b}(t)\| = \frac{1}{2}\|\boldsymbol{v}(t-1) - \boldsymbol{v}(t+1)\| \tag{4.17}$$

where $\boldsymbol{b}(t)$ is the length of the triangle's base, and the midpoint between $\boldsymbol{v}(t-1)$ and $\boldsymbol{v}(t+1)$ is simply

$$\boldsymbol{b}_m(t) = \boldsymbol{v}(t+1) + \frac{1}{2}\boldsymbol{b}(t). \tag{4.18}$$

The component b is the distance between $\boldsymbol{b}_m(t)$ and the ideal vertex position $\boldsymbol{v}'(t)$ on the perpendicular bisector. This point lies on the perpendicular bisector such that the angle between $\boldsymbol{v}(t-1) - \boldsymbol{v}'(t)$ and $\boldsymbol{v}'(t) - \boldsymbol{v}(t+1)$ is equal to the mean angle $\theta(t)$ of the three surrounding

contour edges, such that

$$\theta(t) = \frac{1}{3}\cos^{-1}\left(\frac{\boldsymbol{u}(t)\cdot\boldsymbol{u}(t-1)}{\|\boldsymbol{u}(t)\|\,\|\boldsymbol{u}(t-1)\|}\right) \tag{4.19}$$

where $\boldsymbol{u}(t) = \boldsymbol{v}(t-1) - \boldsymbol{v}(t)$ for any non-zero vector $\boldsymbol{u}(t)$ [143]. Therefore angle $\theta''(t)$ (see figure 4.7) is

$$\theta'(t) = \frac{180 - \theta(t)}{2} \tag{4.20}$$

and subsequently

$$\theta''(t) = 180 - 90 - \theta'(t). \tag{4.21}$$

The distance b is calculated through basic trigonometry, such that

$$b = \frac{1}{2}\|\boldsymbol{b}(t)\|\tan\theta''(t) \tag{4.22}$$

and the ideal vertex position $\boldsymbol{v}'(t)$ is therefore $\boldsymbol{v}(t-1) + [a\ b]^T$ for the special case that the base of the triangle $\boldsymbol{b}(t)$ is parallel to the x-axis. To generalise this, the vector $[a\ b]^T$ needs to be rotated by θ^* degrees, where θ^* is the angle at which the triangle's base intercepts the x-axis, such that

$$\boldsymbol{v}'(t) = \boldsymbol{v}(t-1) + \begin{bmatrix} a \\ b \end{bmatrix}\begin{bmatrix} \cos(\theta^*) & \sin(\theta^*) \\ -\sin(\theta^*) & \cos(\theta^*) \end{bmatrix} \tag{4.23}$$

where $\theta^* = \frac{\boldsymbol{u}(t)\cdot[1\ 0]}{\|\boldsymbol{u}(t)\|}$. The energy term, as defined by Perrin and Smith [141], is thus the distance between $\boldsymbol{v}'(t)$ and $\boldsymbol{v}(t)$, such that

$$E_{\text{int}}(\boldsymbol{v}(t), \boldsymbol{x}) = \|[\boldsymbol{v}(t) + \boldsymbol{x}] - \boldsymbol{v}'(t)\| \tag{4.24}$$

where $\boldsymbol{x} \in \text{neighbourhood}(\boldsymbol{v}(t))$.

This energy term preserves the curvature criteria, defined originally by the second derivative along the contour, by enforcing that the angles between the contour's edges are equal. It also preserves the continuity criteria, defined originally by the first derivative along the contour, by forcing each contour vertex towards a point on the perpendicular bisector of the surrounding vertices. The energy term proposed by Perrin and Smith therefore combines the properties of the two original internal energy terms into one and thus reduces the number of internal energy parameters by the same factor.

4.2.4 Energy Minimisation

The minimum of the active contour's energy, as defined by Eq. (4.1), translates to the detection of a feature within a spectrogram. The iterative greedy algorithm proposed by Williams and Shah is used as the energy minimisation technique, as it has a low computational complexity $O(nm)$ and it relaxes the constraints upon the forms of the energy functions imposed by other minimisation techniques. Specifically, the algorithm allows the energy terms to have non-differentiable forms, such as those of the internal and potential energies outlined in this chapter, without loss of accuracy [184]. In terms of execution time, it has also been shown to outperform energy minimisation

Algorithm 4.1 Contour energy minimisation
Input: $\boldsymbol{S}$, spectrogram; f_1 and f_2, search range.
Output: $\hat{\boldsymbol{v}}$, contour positions.

1: $s \leftarrow 1$
2: initialise contour $\boldsymbol{v}^s$ at -1
3: initialise contour $\boldsymbol{v}^{s-1}$ at f_1
4: **while** $\forall \boldsymbol{v}^s(t) < f_2$ **do**
5: **while** $\forall \boldsymbol{v}^s(t) \neq \boldsymbol{v}^{s-1}(t)$ **do**
6: **for** $t = 0, 1, \ldots, k-1$ **do**
7: **if** $\boldsymbol{v}^{s-1}(t) < f_2$ **then**
8: $\boldsymbol{v}^s(t) = [i, j]^T$ where $\underset{(i,j) \in \text{neighbourhood}(\boldsymbol{v}^{s-1}(t))}{\arg\min} \mathcal{E}([i, j]^T)$
9: **else**
10: $\boldsymbol{v}^s(t) = \boldsymbol{v}^{s-1}(t)$
11: **end if**
12: **end for**
13: **end while**
14: **if** $\exists \boldsymbol{v}^s(t) < f_2$ **then**
15: store $\boldsymbol{v}^s(t)$ in detections such that $\hat{\boldsymbol{v}}^j(t) = \boldsymbol{v}^s(t)$
16: $j \leftarrow j + 1$
17: **for** $t = 0, 1, \ldots, k-1$ **do**
18: $\boldsymbol{v}^{s+1}(t) = \boldsymbol{v}^s(t) + [2, 0]^T$
19: $s_{y(t),x(t)} = 0$, where $[x(t), y(t)] = \boldsymbol{v}^s(t)$
20: **end for**
21: $s \leftarrow s + 1$
22: **end if**
23: **end while**
24: **return** detections $\hat{\boldsymbol{v}}$

using finite differences and LU decomposition as utilised by Cohen and Cohen [89]. The greedy algorithm is a gradient descent method for energy minimisation and the pseudo-code that describes this process in detail is presented in Algorithm 4.1. A cautionary note; due to the iterative nature of this energy minimisation process it is possible that the algorithm cycles between two low energy states indefinitely and therefore this occurrence should form an additional stopping condition when implementing the algorithm.

The process updates each contour vertex's position to the minimum within its local neighbourhood (determined by the function neighbourhood$(\boldsymbol{v}(t))$). The neighbourhood is normally taken to be the 3×3 square neighbourhood centred on the contour vertex. This is repeated for each vertex until the search range has been exceeded (or no movement occurs)—at this stage the contour has converged to a minimum of the energy function $\mathcal{E}$. The position of each contour vertex at the minimum is stored as a detection (see Section 4.2.5 below) and the contour re-initialised at higher frequency bins that are out of range of the current detection; in the case of a 3×3 pixel neighbourhood each contour vertex is re-initialised two pixels higher in the frequency axis (line 18 in Algorithm 4.1). In this way, the contour does not miss any tracks that are close to the first. Although the contour is re-initialised two pixels after a detection, the space between the position of re-initialisation and the detection is captured within each contour vertex's local neighbourhood,

and so this space is not ignored. It is however harder for the contour to reach this space due to the gradient potential biasing it in the increasing frequency direction. Once the maximum frequency defined by the search range is reached the algorithm terminates, returning the detections found during the energy minimisation process. The search range $f_1 \in \mathbb{N}$ and $f_2 \in \mathbb{N}$ can be taken as the complete frequency range available in the spectrogram, i.e. $f_1 = 0$ and $f_2 = M - 1$, so that an exhaustive search for tracks within the spectrogram is conducted. Alternatively, if it is known that the tracks to be detected are located within a specific frequency range, a contiguous subset of the spectrogram corresponding to that range can be specified such that $f_1 < f_2 < M - 1$.

A drawback of performing energy minimisation for the detection of features is that if a weak feature and a strong feature reside in close proximity to each other (within each other's local neighbourhood) and the weak feature is encountered prior to the strong feature, it will be missed. This is because the contour is drawn away from the weak feature as the minimum within the local neighbourhood moves to the position of the stronger feature—note that this will not occur if the strong feature is encountered first as, according to line 19 of Algorithm 4.1, it is removed before the contour is re-initialised.

4.2.4.1 A Note on the Vertices' Neighbourhood

Now that the potential energy is formulated to make use of pixels taken from within a window the configuration of these windows in the vertex's neighbourhood should be considered. The original potential energy makes use of the pixel values in the vertex's neighbourhood and these pixels can be thought of as a special case of a window that has the size 1×1 pixels. Extending this to a window of a larger size results in windows that are centred upon each point in the neighbourhood. If, for example, the window has a width of 3 pixels and these windows are centred upon each point in the neighbourhood, the information derived from several points in the neighbourhood will overlap. To correct for this, neighbourhood positions to the left of the vertex should be associated with the rightmost column of the window, those to the right associated with the leftmost column of the window and those in the centre of the neighbourhood should be associated with the centre column of the window. This results in a configuration in which the evolution for each side of the contour is driven solely by information from that side.

4.2.5 Rolling Window

Thus far, the spectrogram has been treated as a stationary image, however, in real applications this is not always the case. The spectrogram can be constructed in real-time and, as such, updated as each observation arrives. The short-term Fourier transform is applied to the observation and a new row of the spectrogram becomes available. Conventionally, at this time the oldest row of the spectrogram is removed and a "rolling window" or "waterfall display" of a fixed height is formed. Track detection can be repeated within this scrolling window as the data is updated. This leads to the consideration of how often the detection is performed and how the output of the algorithm (the positions in each rolling window at which a track is detected) is interpreted. Consequently, two configurations arise, each having separate approaches to interpreting the output.

The first configuration is as follows; the detection is performed within the rolling window as each new row becomes available and therefore the detection process integrates past (and perhaps future) information to enhance the detection at each time point. In this way each row supports k detections as it flows down a rolling window k pixels in height and the active contour has k vertices corresponding to each row in the rolling window. In this configuration several approaches to interpreting the algorithm's output exist, each of which produces a different system behaviour, these are:

a) The set of detection locations D_t composed of the co-ordinate positions $[l, t]^T$ of the first contour vertex of each detection within the rolling window (for each update of the rolling window), such that

$$D_t = \left\{ [l, j]^T \middle| [l, j]^T = \left[i, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \hat{\boldsymbol{v}}_i^j(0) \right]^T \right\} \tag{4.25}$$

where $i = k, k+1, \ldots, N-1$ is each row of the spectrogram and $\hat{\boldsymbol{v}}_i^j(0)$ is the location in the first row of the jth detection (convergence of the contour) within the rolling window that has row i of the spectrogram as its first row. The multiplication of $\hat{\boldsymbol{v}}_i^j(0)$ with the vector $[1\ 0]^T$ simply extracts the x-axis co-ordinate from the first row. This can be interpreted as the detection process utilising past information to enhance the detection in the current time step.

b) The mean position along the x-axis of all the contour vertices of each detection within the rolling window, such that

$$D_t = \left\{ [l, j]^T \middle| [l, j]^T = \left[i - \left\lfloor \frac{k}{2} \right\rfloor, \frac{1}{k} \sum_{t=0}^{k-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \hat{\boldsymbol{v}}_i^j(t) \right]^T \right\}. \tag{4.26}$$

This configuration could be beneficial if smoothing of the detection output is needed, the averaging of locations smoothes detection irregularities.

c) A combination of the output from the k detections that each row supports, such that

$$D_t = \left\{ [l, j]^T \middle| [l, j]^T = \left[i - t, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \hat{\boldsymbol{v}}_i^j(t) \right]^T \right\}, \qquad t = 0, 1, \ldots, k-1 \tag{4.27}$$

which allows an initial, quick, estimate to be made based upon the detection in one time step, which is refined throughout subsequent iterations. This interpretation requires post-processing of the set D_t to combine the detections from the multiple iterations.

This configuration is characterised by an initial detection delay after the arrival of the first observation equal to the time that it takes to fill the rolling window. Once this period has passed, detection can take place each time that a new observation arrives.

The second detection configuration simplifies the process by removing the overlap between rolling windows and therefore detection is performed each time that the rolling window is com-

pletely updated. In this configuration each row supports exactly one detection and the delay in detection is the amount of time in which it takes to refresh the whole rolling window. The output of the detection process for each row in the rolling window is stored in each iteration of the algorithm, such that

$$D_t = \left\{ (l,j) \middle| [l,j]^T = [i-t, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \hat{\boldsymbol{v}}_i^j(t)] \right\}, \qquad t = 0, 1, \ldots, k-1 \tag{4.28}$$

where $i = k-1, 2k-1, \ldots, \lfloor \frac{N}{k} \rfloor$ is the row of the spectrogram and N is the number of rows in the spectrogram.

4.3 Complexity Analysis

In the context of a spectrogram track detection algorithm's application, it is important that detections are made in real-time. In a majority of situations the algorithm would be expected to be used on-line and therefore its complexity should reflect this. As such, any training costs will be ignored and the analysis of complexity will be concerned with the algorithm's on-line execution costs. It is widely accepted that a linear or quadratic time complexity is acceptable as a tractable solution under these considerations. Therefore, an analysis of the algorithm's complexity in terms of time and space, with regard to the potential energy and both formulations of the internal energy, is conducted to establish whether or not it is a tractable solution (it has been shown that the greedy energy minimisation algorithm has complexity O(mn), Section 4.1.1.4). The notation that will be used throughout this analysis is big O, where n is the number of elementary arithmetic operations (add, subtract, multiply and divide).

It will be assumed that vector multiplication (and therefore the dot product) has complexity O(n) as, using schoolbook matrix multiplication [48], multiplying a matrix of size $m \times n$ with a matrix of size $n \times p$ has the complexity O(mnp). As such, two vectors that have the sizes $1 \times x$ and $x \times 1$, and therefore $m = 1$, $n = x$ and $p = 1$, result in a multiplication complexity of O$(1n1) =$ O(n).

4.3.1 Original Internal Energy

A study into the algorithm's time and space complexity reveals that they are both linear with respect to all parameters except the dimensionality of the potential energy's feature space. This non-linearity is the result of the computation and storing of the inverted matrix $\boldsymbol{\Sigma}^{-1}$ in Eq. (4.8), which has a time complexity of O(n^3) using Gaussian elimination [60] and a space complexity of O(n^2). Although, as matrix inversion and multiplication are computationally equivalent [5], the more efficient Strassen [168] and Coppersmith-Winograd [46] algorithms reduce this complexity. Regarding the time complexity, as $\boldsymbol{\Sigma}^{-1}$ does not vary during the algorithm execution its value can be determined prior to execution and stored for subsequent use—reducing the execution time from O(n^3) to O(n^2). Also, the matrix multiplication $\boldsymbol{Q}^T\boldsymbol{\Sigma}^{-1}$ in Eq. (4.8) is processed in O(n^2) time as $\boldsymbol{Q}^T$ has a size of $1 \times s$ and $\boldsymbol{\Sigma}^{-1}$ is a matrix of size $s \times s$ (the complexity of matrix multiplication

between a $n \times m$ matrix and a $m \times d$ matrix is O(nmd) using the schoolbook method [48]—as $m = d$ and $n = 1$ in this case, the order is O(n^2)). Fortunately a low PCA dimensionality (typically 3–6 dimensions) is sufficient to capture enough information for accurate track detection and therefore a non-linearity in s is acceptable.

4.3.2 Perrin Internal Energy

In terms of the potential energy, the same formulation is used in both versions of the algorithm and therefore the complexity remains O(n^2) due to the matrix multiplication between $\boldsymbol{Q}^T$ and $\boldsymbol{\Sigma}^{-1}$, as described previously. What remains to be shown is that the Perrin internal energy formulation does not have a greater complexity than this. It can be seen that the only components of the Perrin calculation (outlined in Section 4.2.3) that are not linear are: $\cos$, sin, $\cos^{-1}$, $||\,||$, and the matrix multiplication in $\boldsymbol{v}'(t)$. Analysing these in turn; the elementary trigonometric functions, $\cos$, $\sin$, and $\cos^{-1}$, can all be computed in $O(M(m)\log(m))$,where m is the number of digits precision and $M(m)$ is the number of single precision operations required to multiply m-bit integers [31], which within this analysis are assumed to be constant. The square and square root involved in the distance function $||\,||$ can all be computed with a complexity of O($M(m)$) using Newton's method [31]. The matrix multiplication involved in calculating $\boldsymbol{v}'(t)$ can be calculated using schoolbook matrix multiplication, which has complexity O(mnp), where m, n and p are the matrix dimensions, in this case $m = 1$, $n = 2$ and $p = 2 = n$ and therefore the complexity is O(n^2) for a fixed $n = 2$ as the size of these matrices do not change. As this is the component that has the highest complexity within the calculation of the Perrin internal energy, and its complexity is equal to that of the original formulation, it can be concluded that the Perrin energy introduces no additional complexity to the algorithm.

4.4 Summary

This chapter has presented an active contour framework for the detection of single and multiple tracks in spectrograms. A discussion of the original active contour algorithm, its limitations and alternative methodologies, has also been presented. This has led to the identification of issues that prevent the algorithm in its original form from being applied to spectrogram track detection. Novel solutions to these problems have been proposed in this chapter. The performance of the original algorithm is dependent upon the location in which it is initialised and the gradient potential energy function was proposed to overcome this. The potential energy force, which defines the feature's location in the image, relies upon intensity information from a single point and this was found in Chapter 3 to be insufficient for this problem. To rectify this, a novel potential energy formulation based upon supervised learning techniques has been proposed to take advantage of structural and intensity information to increase detection rates. Moreover, this potential energy has been extended to integrate information from harmonically related positions in the spectrogram to improve detection at low SNRs. The potential energy explicitly models the noise, which improves generalisation to unknown track structures when using machine learning techniques. The conditions under which the noise model can be augmented with information from the track class, in problems

where it is well defined, have been identified in Section 4.2.2.4. It has also been shown that the original internal energies bias the contour towards a configuration that does not accurately model the variation of structure observed in spectrogram tracks. These energies were therefore replaced by the internal energy proposed by Perrin and Smith, which removes the 'geometrical' bias. Within the proposed framework the greedy energy minimisation algorithm was preferred over the variational calculus approach as it relaxes the restrictions imposed upon the forms of the energy functions as well as having a reduced computational complexity. An analysis of the framework's computational complexity has shown that it is applicable to real-time implementations.

Chapter 5

Algorithm Evaluation

"The true method of knowledge is experiment."

— William Blake, 1757–1827.

It has been demonstrated in Chapter 3 that, when detecting low-level features, structural and harmonic information can be integrated into the process to improve track detection rates. In Chapter 4 these findings have been realised within an active contour algorithm for high-level track detection. The active contour algorithm allows generalisation to unseen track structures due to the flexibility of the model in the absence of rigid shape parameters.

In this chapter the above algorithm is assessed using the synthetic data set described in Section 3.3.1. The first section of this chapter presents, and discusses, the evaluation measure that will be used throughout the experimentation. The subsequent section presents an analysis of several of the algorithm's parameters for which values can be determined from the experimental results of Chapter 3. Following this analysis, several empirical comparisons are made between the detection performance achieved using variants of the algorithm. These comparisons aim to demonstrate the algorithm's applicability to the problem of weak feature detection, and more specifically, to determine the validity of the following hypotheses:

- The internal energy proposed by Perrin and Smith models the track structure observed in this problem better than the original internal energy.
- The potential energy formulation proposed in this thesis increases track detection performance when compared with the original potential energy.
- The detection of harmonic patterns of tracks, as opposed to individual tracks, increases high-level detection performance.

Preceding each evaluation is presented a study into the stability of each variant of the algorithm in relation to a variety of parameter value combinations. This leads to the selection of appropriate combinations of values for use in each experiment, which are designed to test the various aspects of the algorithm by applying it to a number of test scenarios. A discussion of the findings from each evaluation is presented, related to existing literature, and finally, the chapter's conclusions are drawn.

5.1 Evaluation Measure

The evaluation measure chosen for use during this evaluation is the Line Location Accuracy (LLA) proposed by Pratt [145], and used by Di Martino and Tabbone [57] for this application. The accuracy is evaluated by a figure of merit that is at its maximum when all track pixels are detected and no false positive detections exist. A set of all detected pixel locations D_t can be constructed from the output of the track detection model, as defined in the previous chapter, and there also exists ground truth data. This data is in the form of a set of actual pixel locations $P_t = \{(i,j)|s_{ij} \text{ belongs to a track}\}$, therefore the figure of merit is formulated such that

$$F = \frac{1}{\max(|P_t|, |D_t|)} \sum_{(i,j)\in D_t} \frac{1}{1 + \lambda \min\limits_{(l,k)\in P_t} (||[i,j] - [l,k]||^2)} \tag{5.1}$$

where $|P_t|$ and $|D_t|$ are the cardinalities of the actual and detected track pixel set respectively, $||[i,j] - [l,k]||$ is the Euclidean distance between the detected track pixel $[i,j]$ and the actual track pixel $[l,k]$ and $\lambda \in \mathbb{R}^+$ where, throughout this chapter, $\lambda = 1$.

The parameter λ controls the influence of the distance from detection to true location, values below one suppress the measure's degradation resulting from inaccurate detections or false positives. It is therefore possible that two methods which achieve high LLAs when $\lambda = 1$ may have equal LLAs when $\lambda = 0$ as, in this state, the figure of merit is simply measuring the ratio between the number of detections and the number of true locations. Furthermore, an algorithm that results in fewer, more accurate, detections will result in a higher LLA than one which produces the correct number of inaccurate detections when $\lambda = 1$. When, however, λ is sufficiently small, it is possible that the LLA of the former becomes less than that of the latter.

It can be observed in Eq (5.1) that the occurrence of a number of conditions drive the value below its maximum. In the case that a detection is close to, but not at the same location as the true occurrence, the Euclidean distance between the detection and the true position, $|[i,j] - [l,k]|^2$, reduces the function's value. The difference between the number of detections and the number of true occurrences, $|D_t|$ and $|P_t|$, also reduces the function's value when false negative or false positive detections occur. The LLA measure therefore aggregates a number of commonly measured factors into one figure of merit.

5.1.1 Experimental Data

The training and test sets described in Section 3.3.1, upon which the low-level feature detectors were evaluated, are combined to form the training set used during the experiments presented in this chapter. The training set therefore comprises two spectrograms generated from each of the parameter combinations previously outlined in Section 3.3.1. Furthermore, a new test set is generated, in the same manner as previously described, which contains ten spectrograms generated from each of the parameter combination.

5.2 Parameter Selection

It has been shown in Section 3.3 that by calculating the PCA vectors using low SNR data samples, the signal detection rate is maximised. The same procedure is used to calculate the PCA vectors for use in the active contour's potential energy to provide the high-level algorithm with sufficient information to detect the features. The lower dimensional subspace in which the noise is modelled using Eq. (4.8) is therefore derived using PCA and $1{,}000$ feature vectors containing noise and $1{,}000$ feature vectors containing track and noise, each extracted from within a 3×21 pixel window from spectrograms having a mean SNR of $-0.5\,\text{dB}$. Previously, however, it has been found that the best ROC performance is achieved using a subspace having six dimensions. The ROC measure used to determine this is a balance between the false positive detection rate and false negative detection rate. The active contour algorithm is sensitive to false positive detections, which can cause a local minimum of the contour's energy gradient within its search space and therefore result in false positive detections within the high-level detection process. It can be seen in Appendix A, Fig. A.1, that increasing the dimensionality of the subspace, whilst increasing the track detection performance, also decreases the noise detection performance. By analysing the track detection performance as the dimensionality increases it can be seen that a good balance is achieved at a dimensionality of three. Using fewer than three dimensions results in a large decrease of the track detection performance and incurs an increase in its variability. Adding further dimensions, although increasing the track detection performance, only does so by approximately 1–2% per dimension. The noise detection performance is reduced by a much smaller amount, however, it should be noted that a typical spectrogram is largely composed of noise. For example, in the conditions of this evaluation the percentage of each spectrogram that forms part of a track is 0.63%—the remaining 99.37% is noise. It is therefore much more beneficial for high-level algorithms, such as the active contour algorithm, to have fewer false positive detections made at the low-level. The use of three dimensions is further justified by analysing the PCA eigenvalues, which are presented in Fig. 5.1 and show that, of all the principal components, the first three account for the largest portion of the data's variance and, by definition, these three principal components minimise the data's mean square error.

Surface views of the principal component vectors, which can be seen in Fig. 5.2, confirm PCA's ability to capture salient information from this data. The first is similar to the Prewitt, first derivative, edge detector [147]; the second, a second partial derivative edge detector, similar to Eq. (5.2); and the third the inverse of that.

$$s''_{ij} = s_{i-1,j} - 2s_{ij} + s_{i+1,j} \tag{5.2}$$

Finally, the potential energy term, presented in Eq. (4.15), allows *a priori* information regarding the position of relative frequencies to be built into the detection process through the use of the pattern set $\boldsymbol{P}_s$. This additional information enables the potential energy term to sample multiple concurrent features and, therefore, increase the reliability of detection. Within the data set, five frequency tracks are present that are described by the pattern set $\boldsymbol{P}_s = \{1, 2, 3, 4, 5\}$ (plus the

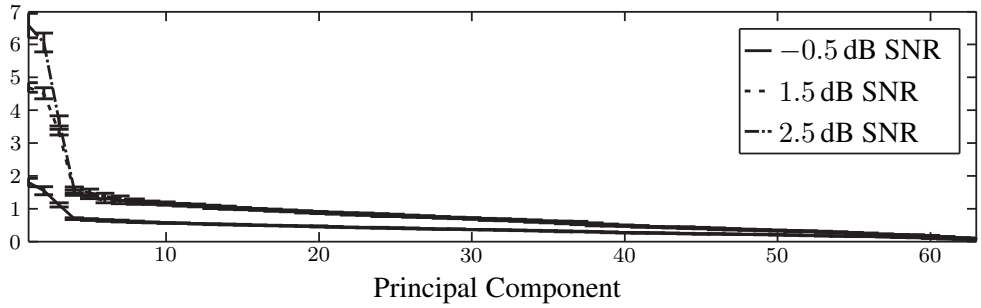
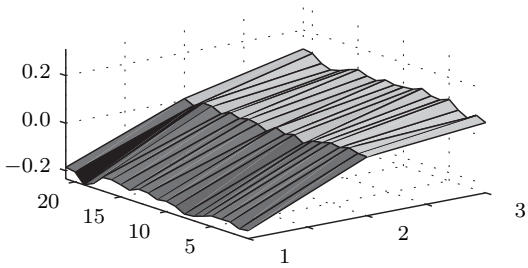
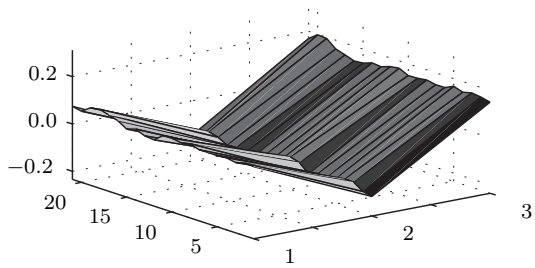


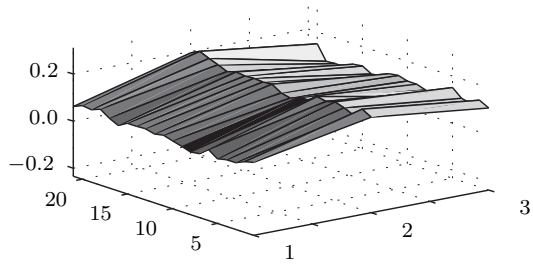
Figure 5.1: The eigenvalues associated with the principal components derived by averaging over 10 random training sets, each containing 1,000 examples of each class. The top line represents the eigenvalues for 2.5 dB SNR examples, the middle 1.5 dB SNR and the bottom −0.5 dB SNR and error bars of 2 standard deviations (SNRs have been rounded to the nearest 0.5 dB).



(a) First Principal Component.



(b) Second Principal Component.



(c) Third Principal Component.

Figure 5.2: The first three principal component vectors viewed as 3×21 point surface plots.

fundamental frequency). The search was optimised by initialising the contour within 10% of the expected frequency position for a particular source. Throughout all of the experiments presented in this chapter, the active contour's length is set to $k = 20$ and all SNRs quoted in this chapter have been rounded to the nearest 0.5 dB and calculated according to the definition presented in Section 1.2.3.

5.3 Comparison of Internal Energies

The first of the hypotheses presented by this chapter is that the internal energy proposed by Perrin and Smith models the track structure observed in this problem better than the original internal energy. It has been discussed in Chapter 4 that the original internal energies bias the contour to

take on a straight and vertical configuration and that the internal energy proposed by Perrin and Smith removes this bias. First, using the potential energy proposed in this thesis, each algorithm's sensitivity to varying weighting values is analysed, a suitable combination of weights for each is selected, and the analysis is presented.

5.3.1 Parameter Sensitivity

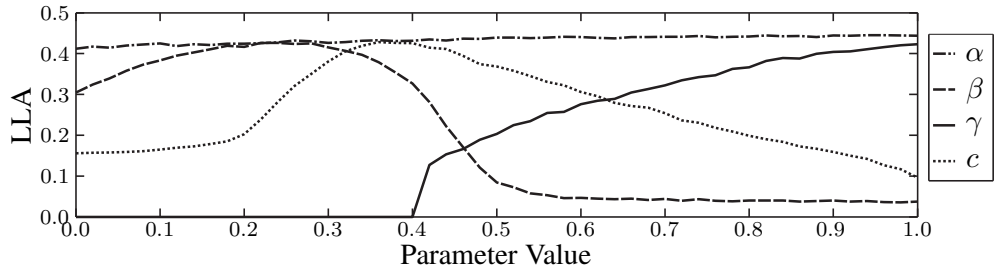
The weighting parameters of the internal and potential energies of the active contour algorithm and the gradient potential, affect its ability to locate features in the spectrogram and to model the local deformation of these features. Ballerini explains that "large values for the continuity and curvature weights will discourage convergence to a 'busy' contour" and that "small weights may allow the contour to be trapped into false edges or leak out through gaps in the boundary" [17]. These parameters, therefore, form the balance of energies that determine the final contour convergence. As such, it is difficult to specify optimal parameter values using heuristics. Instead, a good set of parameter values is searched for by varying each parameter in turn throughout its range of values. During this search the remaining parameters are fixed at values that have been found to lead to good convergences during preliminary experimentation. Each parameter combination is evaluated using the training set and the values that lead to the maximum performance are chosen as those for use in the evaluation of the test set. In this search, therefore, the interactions between the energies that these parameter values control are ignored. This assumption means that the results are likely to be sub-optimal, however, it greatly simplifies the optimisation process.

In this comparison two variants of the algorithm will be evaluated, both of which will employ the definition of potential energy presented in this thesis. The difference between the two variants will be that one has the original internal energy and the other the internal energy proposed by Perrin and Smith. Plotting the line location accuracy as a function of each parameter's value also allows the algorithm's sensitivity to parameter values to be analysed. Figure 5.3 presents the results of this empirical investigation and affords an insight into the role of each parameter.

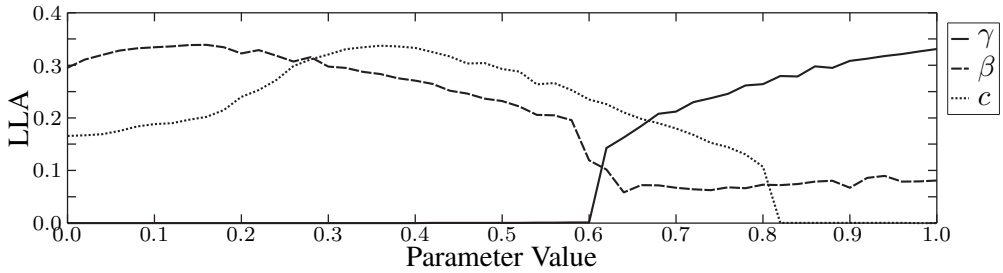
The potential energy is the algorithm's sole source of information to allow for the accurate location of features in an image. Its weight is controlled by the value of the parameter γ, and as this increases the active contour gains more information from the spectrogram. This fact is directly reflected in both of the observed functions; as γ increases the LLAs also increase.

The gradient potential parameter c enables the active contour to locate features that lie outside its local gradient topology and to pass over false positive detections that result from the potential energy. It is observed in both of the algorithm variants that as c increases, i.e. the contour moves over false positives with a greater force, the LLAs also increase. If the value, however, is too great (above 0.36) the contour begins to be forced over true positives and the detection rates decrease. In both variants of the algorithm, the functions of c have, in general, the same form and the peak in performance is observed at the same value, indicating that the gradient potential balances the effects of the potential energy and is, in the most part, independent of the contour's internal energy.

The internal energy parameters control the contour's ability to deform and to model the track's structure. When the original internal energy variant is considered, it can be observed that the value of α (which controls the continuity of the contour) has very little effect on the detection



(a) Original internal energy algorithm variant. The mean standard deviations for each function are $\alpha = 0.0042$, $\beta = 0.0052$, $\gamma = 0.0044$ and $c = 0.0057$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.1a. Whilst varying each of the parameters the remainder took the following values: $\alpha = 0.10$, $\beta = 0.20$, $\gamma = 1.00$ and $c = 0.41$.



(b) Perrin internal energy algorithm variant. The mean standard deviations for each function are $\beta = 0.0062$, $\gamma = 0.0028$ and $c = 0.0047$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.1b. Whilst varying each of the parameters the remainder took the following values: $\beta = 0.20$, $\gamma = 1.00$ and $c = 0.41$.

Figure 5.3: The mean line location accuracy of training set detection as functions of each variant of the algorithm's parameter values. The results were obtained from five repetitions of the experiment using the potential energy proposed in this thesis.

rate. It seems that the information captured through the contour's second derivative, controlled by the parameter β (which controls the curvature of the contour), overlaps that captured by the first derivative. The parameter β has a far greater effect; at low values the contour has sufficient freedom to model track variations and evolve, however, when the influence is too great (above a value of approximately 0.4) this ability is restricted and the contour is not able to evolve and model the tracks. A similar behaviour is observed in the variant that utilises the internal energy proposed by Perrin and Smith, at low values of β (less than 0.2) the maximum of the LLA is reached and the performance is relatively stable. Above this range, however, the contour's ability to evolve and model the track's structure is restricted and the LLA degrades accordingly.

To confirm that choosing the parameter values by analysing the algorithm's performance as a function of LLA is sensible, several additional measures were taken during the experiments. These were designed to evaluate each compositional measure that the LLA aggregates into one measure. The additional measures were the proportion of true track occurrences detected (those that are within five pixels distance of the true location) and the mean probability of false positive detections per row of the spectrogram (the number of additional detections within the five pixel range plus those outside of this range). The results are presented in Appendix A, Fig. A.3 and Fig. A.9, and show that, in the large part, the parameter values for each algorithm behave in accordance

with the corresponding LLA. In both cases, as the potential energy parameter γ is increased, the detection performance also increases and the false positive rate remains low (below 0.005 in the Perrin algorithm variant and 0.034 in the original variant). The internal energy parameter β of the Perrin variant reaches its LLA maximum at 0.16. It is evident from Figure A.3 that the detection performance remains fairly stable at low values of β and is inversely proportional to β as its value increases. The false positive rate exhibits similar behaviour; at low values of the parameter β a small number of false positive detections are observed and these decline as β's value is increased. The parameter controlling the effect of the gradient potential, c, has a distinct LLA maximum at a value of 0.36. Again, analysing the detection rate reveals that the proportion of true positive detections is inversely proportional to the value of c, however, the mean probability of false positive detections per row is also inversely proportional to this value. The LLA measure has, therefore, chosen a balance between these two measures, and the maximum corresponds to the value at which the mean probability of false positive detections per row is low and the proportion of true positive detections remains relatively high. Therefore, in the case of the Perrin algorithm variant, the LLA measure has determined an acceptable balance between minimising the probability of false positive detections and maximising the number of true positive detections.

Regarding the original algorithm variant, the proportion of true positive detections and the mean probability of false positive detections per row is presented in Appendix A, Figure A.9. As discussed earlier, the value of α shows no effect on the detection performance, which is confirmed by the number of observed true and false positive detections. The internal energy weighting parameter of the contour's second derivative, β, reaches its maximum LLA at a value of 0.22. This corresponds to the point at which the mean probability of false positive detections per row starts to increase. The proportion of true positive detections is very close to its maximum value and therefore the LLA measure has provided an adequate balance between these two measures. In confirmation of the previous observation, the gradient potential parameter, in all measures, exhibits similarly behaviour to the gradient potential parameter in the Perrin algorithm variant, although the responses are at higher values. Again, the LLA measure has reached a compromise between minimising false positive detections and maximising the proportion of true positive detections, occurring at a value of 0.36.

In accordance with the results found during this investigation, and for the remainder of the experiments in this section, the following parameter values are used; for the original internal energy variant, the internal energy parameters are set to: $\alpha = 0.96$ and $\beta = 0.22$, the potential energy parameter to $\gamma = 1.00$ and the gradient potential to $c = 0.36$; and for the Perrin and Smith internal energy variant, the internal energy parameter is set to $\beta = 0.16$, the potential energy parameter to $\gamma = 1.00$ and the gradient potential to $c = 0.36$. These values also correspond to points of low standard deviation of the results, as presented in Appendix A.2.6, Table A.1.

5.3.2 Performance

Using the parameter value combination that has been determined for each variant of the algorithm, it is now possible to analyse the algorithms' performance on each type of track configuration found in the test dataset. The line location accuracy of each variant, applied to each track configuration,

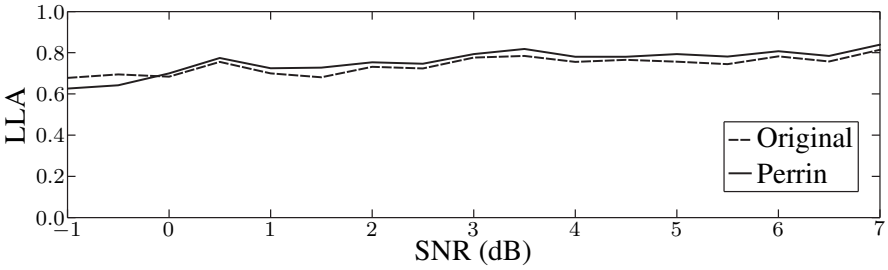
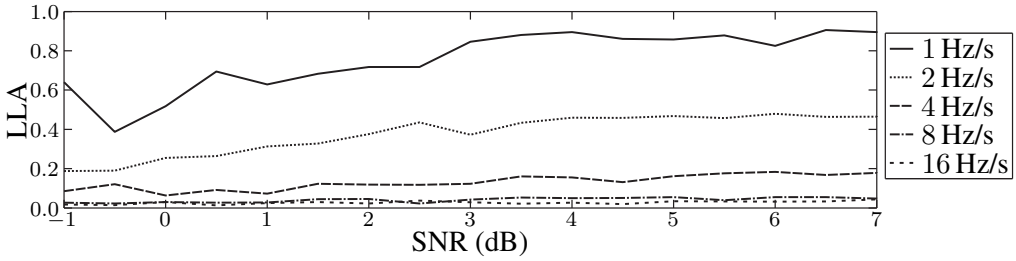


Figure 5.4: The mean line location accuracies of vertical track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. The mean standard deviations for the Original detections is 0.0060 and the Perrin detections 0.0064, a full list of the standard deviations is presented in Appendix A.2.6, Table A.2. The results were obtained from ten repetitions of the experiment using the potential energy proposed in this thesis.

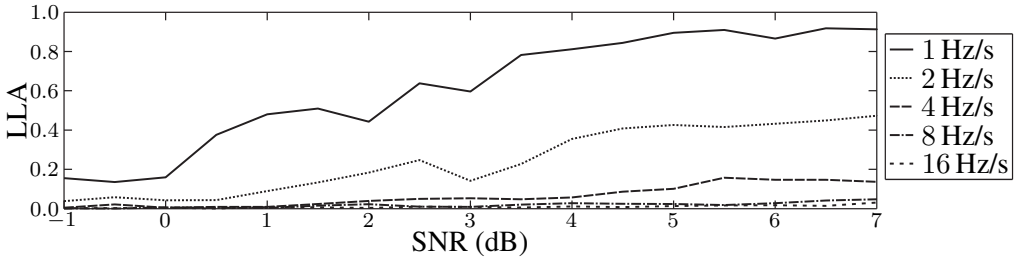
is measured as a function of the SNR of the track. Not only does this allow for the construction of a detailed comparison between the two variants with regards to their ability to detect differing track structures but also for the analysis of the performance as the track's SNR degrades.

The first of these comparisons, with regards to the vertical track structure, is presented in Figure 5.4. At SNRs of $0\,\text{dB}$ and above the Perrin variant outperforms the original by a mean LLA measure of 0.0260. Below this point the performance of both variants degrade, however, the original internal energy is marginally more resilient and has an LLA measure that is, on average, 0.0298 higher than the Perrin internal energy variant in the SNR range -1 to $-0.5\,\text{dB}$. This is possibly an effect of the shape bias that is exhibited by the original internal energy. As it has been shown in Chapter 4, the original internal energy biases the contour to form a straight vertical track, a fact that could explain the apparent difference in performance at low SNRs. In this setting the potential energy's influence is diminished and therefore the internal energy's role is accentuated. This change in the balance of energies allows the internal energy's bias to have greater influence on the contour, resulting in a convergence that coincidentally matches the track's shape.

The proportion of true positive detections and mean probability of false positive detections per row for each algorithm variant are presented in Appendix A, Figure A.4 and Figure A.10. The mean probability of false positive detections per row resulting from the original internal energy is over ten times that of the Perrin internal energy variant, however, the proportions of true positive detections for each variant are comparatively close to each other. The reason that the LLAs of both variants are similar is a combination of two effects. The first is a consequence of the true positive measure, which deems a detection within five pixels of the true location to be a true detection. If false positive detections are located within this distance (and a true positive detection not made), this would artificially inflate the measure. The second effect is due to the LLA incorporating a measure of the detection's distance from the true location and these results indicate that the Perrin variant, although detecting as much, or possibly more, of the tracks than the original variant, it has done so at the expense of location accuracy. The number of true positive detections of each algorithm variant show that there is a relatively small mean difference of 0.0425 between them. This indicates that the original internal energy models the tracks more closely and therefore its resulting detections have a smaller distance from the true location. This finding would be



(a) Original internal energy algorithm variant. The mean standard deviations for the each function is: 1 Hz/s = 0.0128, 2 Hz/s = 0.0075, 4 Hz/s = 0.0052, 8 Hz/s = 0.0031 and 16 Hz/s = 0.0031, a full list of the standard deviations is presented in Appendix A.2.6, Table A.3a.

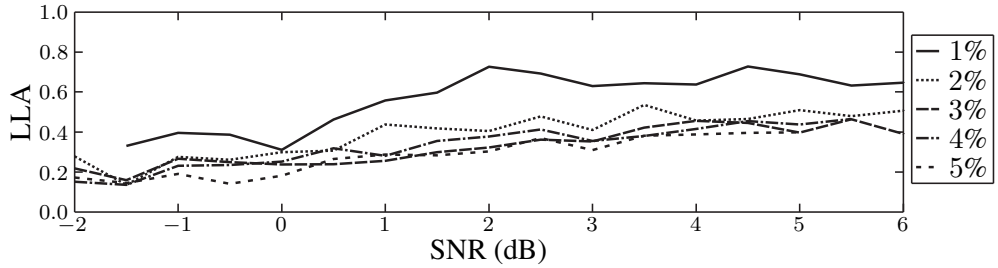


(b) Perrin internal energy algorithm variant. The mean standard deviations for the each function is: 1 Hz/s = 0.0126, 2 Hz/s = 0.0098, 4 Hz/s = 0.0053, 8 Hz/s = 0.0029 and 16 Hz/s = 0.0015, a full list of the standard deviations is presented in Appendix A.2.6, Table A.3b.

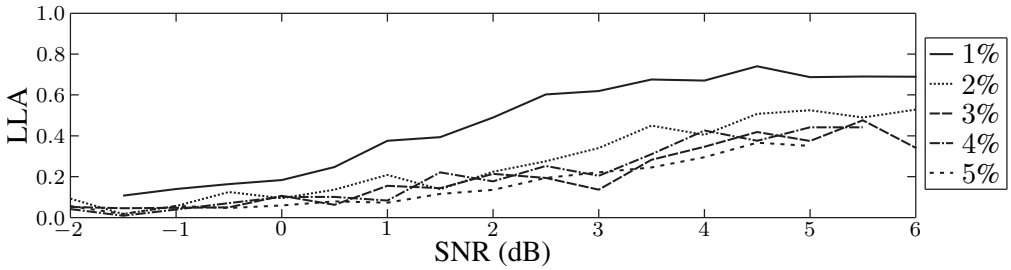
Figure 5.5: The mean line location accuracies of oblique track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. The results were obtained from ten repetitions of the experiment using the potential energy proposed in this thesis.

congruent with the theoretical analysis in Chapter 4.

The results of a comparison between the two algorithm variants when applied to the detection of oblique tracks is presented in Figure 5.5. These results confirm that the two variants of the algorithm achieve very similar LLAs. The original variant, however, demonstrates more resilience to reduced SNRs than the Perrin variant. Nevertheless, this could be partly due to the increase in false positive detections that results from this form of internal energy. In accordance with expectations, as the gradient of the track increases the LLA decreases. When detecting tracks with a gradient of 1 Hz/s the algorithm variants achieve mean LLAs of 0.8996 (Perrin) and 0.8728 (original) at SNRs of 5–7 dB and at 3–7 dB the means reduce to 0.8355 (Perrin) and 0.8715 (original). Therefore, in this case, although the Perrin variant produces higher results at the higher SNRs, the original variant is more consistent as the SNR decreases. When detecting tracks with greater gradients, the algorithm variant achieving the best performance reverses, for example, the mean values for the detection of 2 Hz/s gradient tracks in the same SNR range are: 0.4386 (Perrin) and 0.4658 (original). Tracks that have a gradient of 8 Hz/s and 16 Hz/s seem beyond the capabilities of both variants and the LLAs are close to zero. The original variant has a slightly higher LLA than the Perrin variant, however, the original variant also produces a greater number of false positive detections and therefore, as discussed, it is possible that the increase in LLA is attributed to the increase in the number of false positive detections. As the SNR of the tracks degrade, both algorithms experience a decline in performance, and this occurs at approximately the same point in the



(a) Original internal energy algorithm variant. The mean standard deviations for the each function is: $1\% = 0.0178$, $2\% = 0.0112$, $3\% = 0.0203$, $4\% = 0.0166$ and $5\% = 0.0205$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.4a.



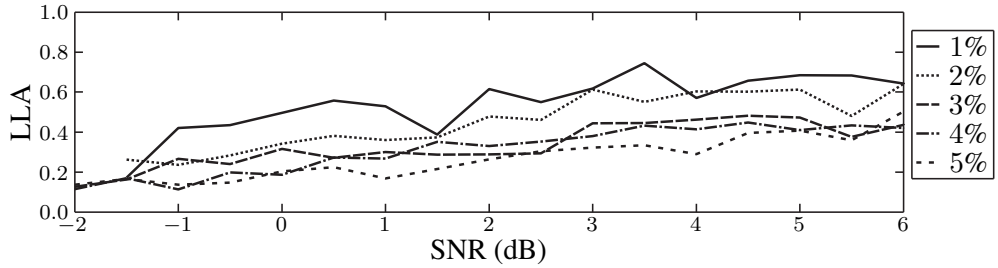
(b) Perrin internal energy algorithm variant. The mean standard deviations for the each function is: $1\% = 0.0044$, $2\% = 0.0056$, $3\% = 0.0083$, $4\% = 0.0166$ and $5\% = 0.0095$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.4b.

Figure 5.6: The mean line location accuracies of sinusoidal (ten second period) track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. The results were obtained from ten repetitions of the experiment using the potential energy proposed in this thesis.

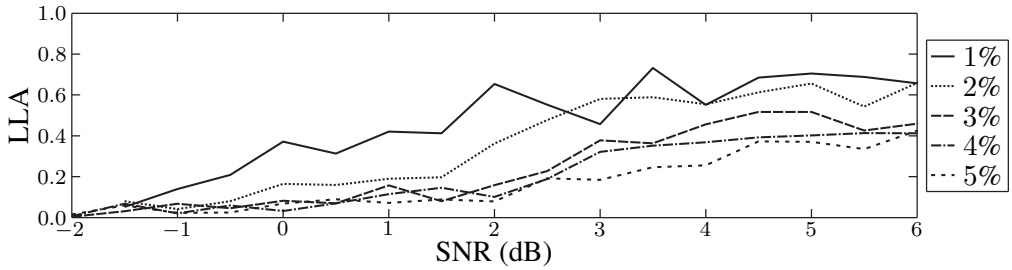
range of SNR considered. The original variant of the algorithm, however, has LLAs of 0.6504, 0.1816 and 0.0866 at an SNR of -1 dB in comparison to 0.1522, 0.0402 and 0.0066 resulting from the Perrin variant.

In Appendix A, Figure A.5 and Figure A.11, is presented the number of true positive and false positive detections attributed to these experiments. Once again, the false positive rate of the original algorithm variant is far greater than that of the Perrin variant; the Perrin variant has a maximum mean probability of false positive detections per row of less than 0.0026, however, the original variant results in a maximum false positive rate that is more than a factor of ten higher 0.0611.

With regards to the detection of sinusoidal tracks, (Figures 5.6, 5.7 and 5.8) an initial observation is that, as would be expected, as the amplitude (the centre frequency variation) of the sinusoid increases the detection rate decreases, which holds for both variants of the algorithm. A similar trend to that found in the oblique track experiments is present in these results, the original variant of the algorithm is more resilient to reduced SNRs than the Perrin variant. When applied to the detection of sinusoidal tracks with a period of ten and fifteen seconds (Figure 5.6 and Figure 5.7) and at SNRs greater than 4 dB both variants result in very similar LLAs; in some cases the original variant is marginally better than the Perrin and in other cases the opposite is true. When, however, the algorithms are applied to the detection of sinusoidal tracks with a period of twenty seconds



(a) Original internal energy algorithm variant. The mean standard deviations for the each function is: $1\% = 0.0163$, $2\% = 0.0099$, $3\% = 0.0148$, $4\% = 0.0137$ and $5\% = 0.0101$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.5a.



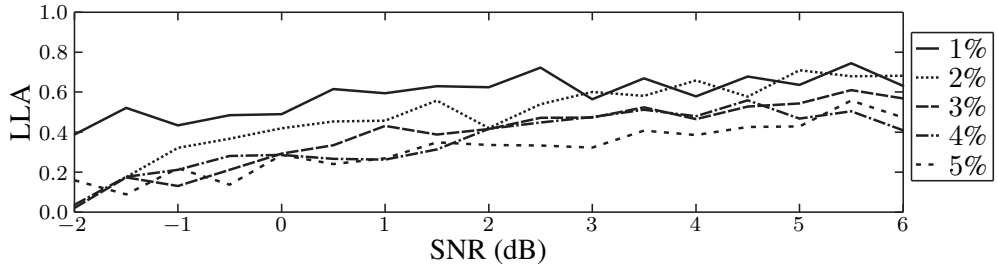
(b) Perrin internal energy algorithm variant. The mean standard deviations for the each function is: $1\% = 0.0051$, $2\% = 0.0055$, $3\% = 0.0049$, $4\% = 0.0050$ and $5\% = 0.0045$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.5b.

Figure 5.7: The mean line location accuracies of sinusoidal (fifteen second period) track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. The results were obtained from ten repetitions of the experiment using the potential energy proposed in this thesis.

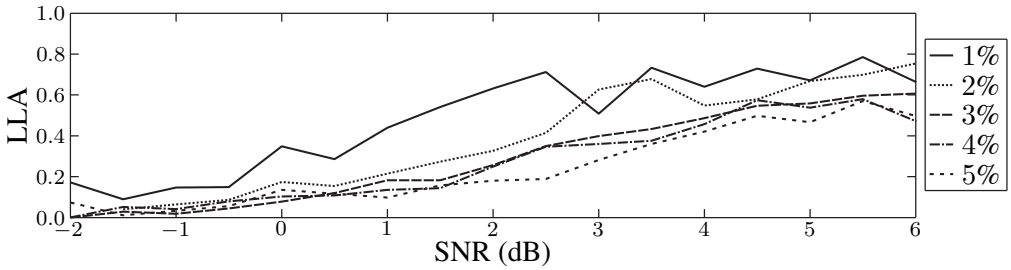
with low centre frequency variation (Figure 5.8) the Perrin variant has marginally better LLAs at SNRs above 4 dB.

Upon inspection of the number of true positive and false positive detections presented in Appendix A, Figures A.6–A.8 and Figures A.12–A.14, it is again obvious that the original variant produces many more false positive detections than the Perrin variant. The Perrin variant results in a maximum mean probability of false positives per row of 0.1155 (occurring when applied to the detection of sinusoidal tracks having a period of ten seconds with 4% centre frequency variation) whereas the original variant produces 0.4546, which is also its maximum mean probability of false positives per row. For the remaining cases the mean probability of false positives per row is largely between 0.02–0.03 for the Perrin variant and 0.2–0.3 for the original, the Perrin variant producing a factor of ten less than the original variant.

So as to not detract from the readability of the results, full lists of the standard deviations for the results presented here are included in Appendix A.2.6, Tables A.2–A.6, the means of which have been presented in each figure's caption. These standard deviations demonstrate that the results have a very low variation, typically exhibiting a standard deviation of less than 0.02 within five repetitions of each experiment. A number of examples of detections that are the result of both variants of the algorithm are presented in Appendix A.2.5, Figures A.27 and A.28.



(a) Original internal energy algorithm variant. The mean standard deviations for the each function is: $1\% = 0.0107$, $2\% = 0.0068$, $3\% = 0.0171$, $4\% = 0.0183$ and $5\% = 0.0121$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.6a.



(b) Perrin internal energy algorithm variant. The mean standard deviations for the each function is: $1\% = 0.0046$, $2\% = 0.0042$, $3\% = 0.0061$, $4\% = 0.0064$ and $5\% = 0.0057$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.6b.

Figure 5.8: The mean line location accuracies of sinusoidal (twenty second period) track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. The results were obtained from ten repetitions of the experiment using the potential energy proposed in this thesis.

5.3.3 Discussion

The overall trend of the results from these experiments imply that, at the higher signal-to-noise ratios, the difference between the two methods is negligible and the Perrin variant often outperforms the original variant. Moreover, when the probability of false positive detections per row is taken into account, the Perrin variant provides more favourable results across all the experiments. When the SNR decreases the original algorithm variant demonstrates more resilience to the reduction in the available information. Nevertheless, it is possible that this is due to the increased probability of false positive detections per row and not the detection ability of the algorithm. In this problem it is difficult to measure the actual true positive detection rate as it is possible that the energy balance, with the addition of the gradient potential, causes the contour to overrun the true location. This fact, in relation to the balloon force, is commented upon by Ji and Yan who state that "these [balloon and similar] forces all have to be included at the equilibrium of their snakes/segments and easily result in a slight overrun of the target contours" [89]. The true positive measures, which have been presented in Appendix A, therefore account for this overrun by taking a true positive detection to be any detection that is within five pixels distance of the true location. It is therefore difficult to separate a true but overrun detection from a false detection. This said, the Perrin energy formulation results in a mean probability of false positive detections per row of less than 0.022,

0.003, 0.115, 0.030 and 0.030 in the vertical, oblique, ten second sinusoidal, fifteen second sinusoidal and twenty second sinusoidal track experiments respectively. These figures translate into maximum mean false positive detections of approximately one in every 45, 333, 9 and 33 lines of the spectrogram. Therefore, the true positive detection figures are relatively reliable when compared with the original variant, which results in maximum mean false positive detections every 4, 16, 3, 4 and 4 lines. It has been seen that when the algorithms' sensitivity to parameter variations were evaluated, the strength of the gradient potential is directly correlated with the probability of false positive detections per row and a value that maximised the LLA was chosen. If, however, the false positive rate were to be minimised, a greater gradient potential should be chosen and it can be seen in Appendix A, Figure A.3b and Figure A.9b, that values of above 0.44 result in a very low false positive detection rate (less than 0.0025 and 0.0237 in the whole training data set for the Perrin and original variants respectively), however, it should be noted that the true positive rate is also directly correlated with the strength of the gradient potential.

It can be seen that, regardless of the internal energy representation used, the active contour algorithm is able to detect all variations of the track structure. There are limitations to this, however; tracks that have a gradient greater than 4 Hz/s are beyond the deformable capabilities of the contour or the generalisation capabilities of the potential energy. Sinusoidal tracks are detected with a high probability at SNRs above 3.5 dB (with reference to the true positive detections presented in Appendix A), and the same can be said for the detection of oblique tracks. Vertical tracks are reliably detected at very low SNRs of around -1 dB.

5.4 Original Potential Energy

A second aim of the comparisons is to determine whether the potential energy proposed in this thesis produces better detection performance than the original potential energy. Again, this assessment is initiated by evaluating the weighting parameter's sensitivity to different values, which allows an appropriate combination of values to be selected for use during the experimentation. As these variants of the algorithm are completely deterministic multiple repetitions yield the same result and therefore results derived from one repetition of the experiments are presented.

5.4.1 Parameter Sensitivity

The line location accuracies that result from the use of the original potential energy (see Figure 5.9) have much more complicated forms than those that resulted from the use of the novel potential energy term (see Figure 5.3). Nevertheless, in a broad sense, the trends between the two are similar. The LLA is proportional to the value of γ as is the case when using the novel potential energy and the LLA is indirectly proportional to the strength of the gradient potential. When, however, the behaviour of the LLA, as the value of the parameter γ increases, is analysed in more detail it becomes apparent that there is a slight reduction in the LLA as the value passes 0.88. Therefore a strong influence from the potential energy, which defines the position of the feature in the image, counterintuitively hinders performance, indicating that it is not suited to this problem. The internal energy parameter β has an interesting form in this variant, its value has

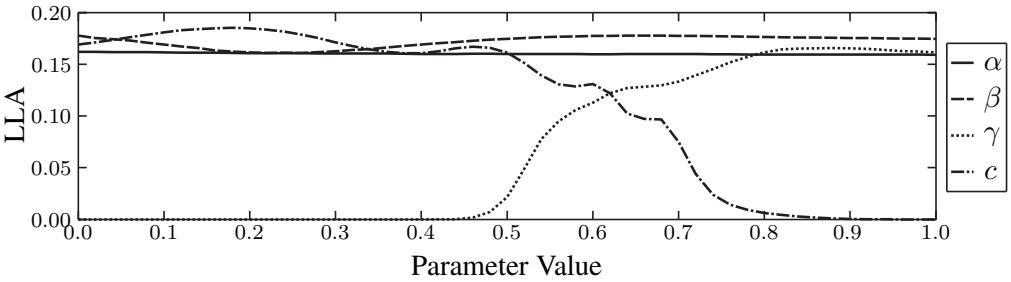


Figure 5.9: The mean line location accuracy of the training set detections as functions of each variant of the algorithm's parameter values. The results were obtained using the original potential energy (the pixel's intensity). Whilst varying each of the parameters the remainder took the following values: $\alpha = 0.1$, $\beta = 0.2$, $\gamma = 1$ and $c = 0.41$

only a very small effect on the LLA. To gain a deeper insight into these results the proportion of true positive detections and the mean probability of false positive detections per row are illustrated in Appendix A, Figure A.15. The probability of false positive detections per row increase as the value of β increases, and the number of true positive detections remains constant. The LLA must therefore be optimising the location of the detection, which is offset with the probability of false positive detections per row when combined to form the LLA measure. As has been mentioned, the LLA of the gradient potential behaves similarly to that of the variants that use the novel potential energy. The true positive detection rate is at its maximum in the range 0–0.5 and then declines after this. The false positive rate also has a similar behaviour; information related to the accuracy of the detections influence the LLA and the value 0.18 therefore results in the best track detection accuracy. The LLA function of the parameter γ also has a similar behaviour; the maximum of this function is reached at a value of 0.82, which is also the maximum of the true positive detections, however, the mean probability of false positive detections per row at this point is very large, at 1.6 per row. As indicated by both the LLA and the proportion of true detections, the value of the parameters controlling the internal energies, β and α, do not have any noticeable effect on the performance of the contour and therefore these parameters are set to the values $\beta = 0.66$ and $\alpha = 0.5$. The maximum LLA when varying parameter c is reached at a value of 0.18.

5.4.2 Performance

The LLAs resulting from the application of the original potential energy to this problem are presented in Figures 5.10–5.12. The LLAs are unequivocally lower than those obtained using the novel potential energy. As has been proved and discussed in Chapter 3, using the intensity values of single pixels as features, in this setting, does not provide enough class separability to be able to reliably detect the tracks. This is also evident in the probability of false positive detections per row; with reference to Figures A.16–A.20 presented in Appendix A.2.3, the mean probability of false positive detections per row of the spectrogram is, on average, sixteen times higher than that found with the novel potential energy.

Referring to the examples of the detections made when using the original potential energy presented in Appendix A.2.5, Figures A.27 and A.28, it becomes apparent that the algorithm is

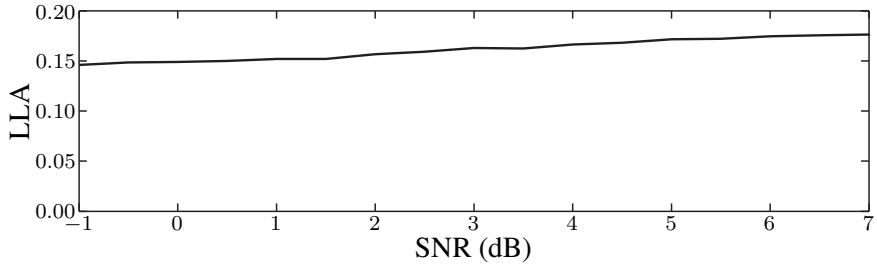


Figure 5.10: The mean line location accuracies of vertical track detections as functions of the spectrogram's SNR. The results were obtained using the original potential energy (the pixel's intensity) and the original internal energies (the active contour's weighted first and second derivatives).

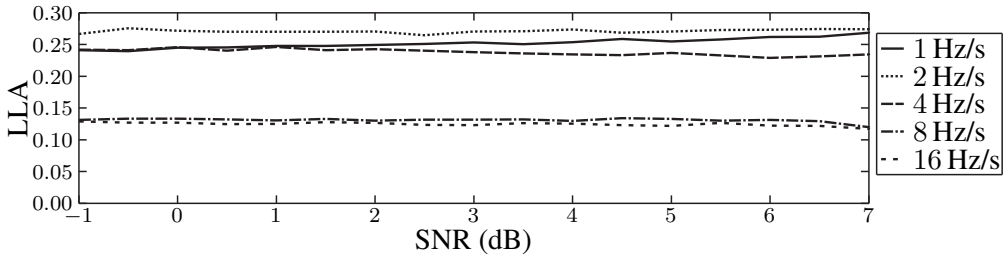


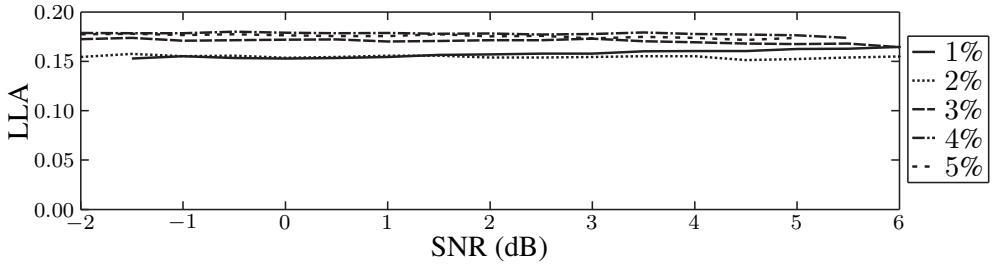
Figure 5.11: The mean line location accuracies of oblique track detections as functions of the spectrogram's SNR. The results were obtained using the original potential energy (the pixel's intensity) and the original internal energy (the active contour's weighted first and second derivatives).

not detecting anything meaningful in the spectrogram. The LLA is therefore giving weight to false positive detections that are near to the true track location. This explains the absence of any performance degradation as the complexity of the tracks increase and also as the SNR decreases. That said, the oblique track case presents some degradation in the LLA as the complexity of the track increases. Coincidently, it can also be seen that the probability of false positive detections (Figure A.17b) reduce in these spectrograms, which is the case for all variants of the algorithm, and therefore the LLA is apparently affected by a reduction in the number of false positives, rather than the complexity of the track leading to fewer detections.

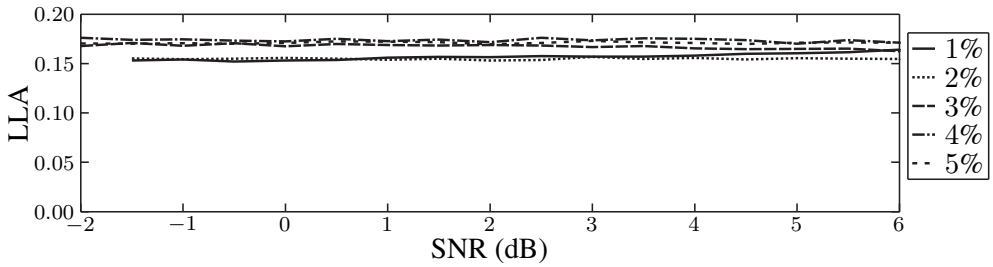
The inappropriate nature of this form of potential energy when applied to this problem is furthermore supported by the fact that, the optimal value for the parameter γ has been found to be less than its maximum value. Moreover, the probability of false positive detections is directly proportional to the potential energy's influence (see Figure A.15 in Appendix A.2.3).

5.5 Multiple Versus Individual Track Detection

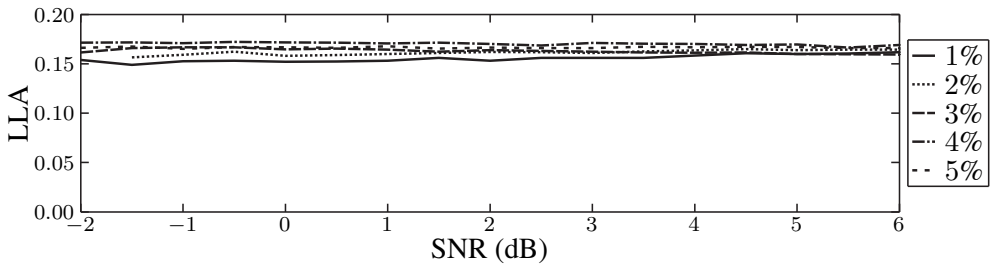
The final hypothesis that this chapter has set out to prove is; the detection of harmonic patterns of tracks, as opposed to individual tracks, increases detection performance. To ascertain the answer to this question, a variant of the algorithm that uses information derived from the fundamental track position, whilst adopting the novel potential energy and the original internal energy, is evaluated and the results are compared with those presented in Section 5.3. Those results having



(a) Ten second period.



(b) Fifteen second period.



(c) Twenty second period.

Figure 5.12: The mean line location accuracies of sinusoidal track detections as functions of the spectrogram's SNR. The results were obtained using the original potential energy (the pixel's intensity) and the original internal energy (the active contour's weighted first and second derivatives).

been obtained using information derived from the fundamental in addition to the harmonic series, are therefore compared with the results obtained using information derived from the fundamental position alone.

The effect of removing the averaging process upon the contour's search space is that false positive detections will be stronger. It is therefore necessary to determine the force of the gradient potential that will be used with the single contour as this controls the contour's ability to pass over false positive detections. An analysis of the LLA as the parameter's value changes is presented in Figure 5.13. As expected, a higher value for the force of the gradient potential than that used for multiple track detection is necessary to induce a high LLA. Therefore, the parameter c will have the value 0.72 throughout the experiments presented in this section. The remaining parameters are fixed at the values found previously, i.e. $\alpha = 0.96$, $\beta = 0.22$ and $\gamma = 1.00$.

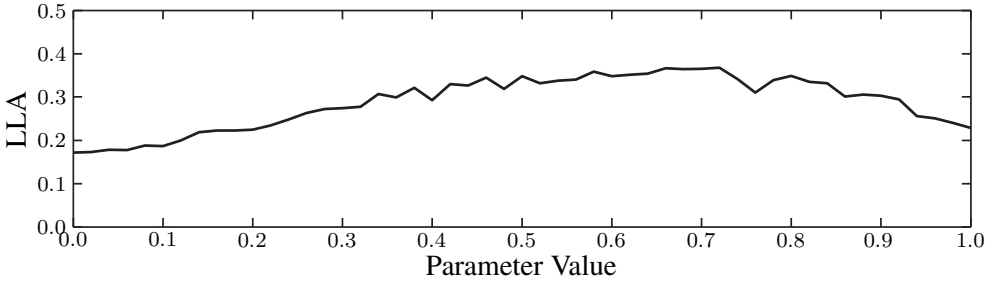


Figure 5.13: The mean line location accuracy of training set detection as functions of each variant of the algorithm's parameter values. The results were obtained from five repetitions of the experiment using the potential energy proposed in this thesis. The mean standard deviations of this function is 0.0204, a full list of the standard deviations is presented in Appendix A.2.6, Table A.7. Whilst varying each of the parameters the remainder took the following values: $\alpha = 0.10$, $\beta = 0.20$, $\gamma = 1.00$ and $c = 0.41$.

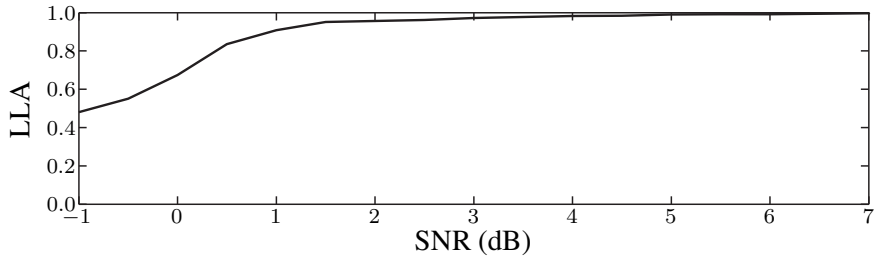


Figure 5.14: The mean line location accuracies of vertical track detections as a function of the spectrogram's SNR. The results were obtained using the potential energy proposed in this thesis and the original internal energy (the active contour's weighted first and second derivatives). The mean standard deviations for the function is 0.0014, a full list of the standard deviations is presented in Appendix A.2.6, Table A.8.

5.5.1 Performance

The results of applying a single contour to the detection of vertical fundamental tracks are presented in Figure 5.14. At SNRs higher than $0\,\text{dB}$ the single contour has a LLA of almost one, a mean increase of 0.2144 over integrating information from four harmonic locations (plus the fundamental's location). The true positive rate is, however, roughly equal at these SNRs (see Appendix A, Figures A.10a and A.22a). The difference in LLA is therefore reflecting the difference in false positive rates, which are presented in Figures A.10b and A.22b in Appendix A. Due to the increased gradient potential necessary to reach the single contour's maximum performance upon the training set, the single contour produces no false positive detections and this increases its LLA. Below a SNR of $0\,\text{dB}$ the single contour's true positive rate falls and consequently the LLA decreases to a mean of 0.1150 below that of the multiple contour detection. When detecting oblique tracks, integrating information from multiple locations increases the LLA in all cases except for $1\,\text{Hz/s}$ gradient tracks at SNRs greater than $5\,\text{dB}$ (in which case the LLA is, on average, 0.0445 higher when using a single contour), see Figure 5.15. Below this point the LLA of simultaneous multiple track detection is, on average, 0.4636 higher than single contour detection. The LLA

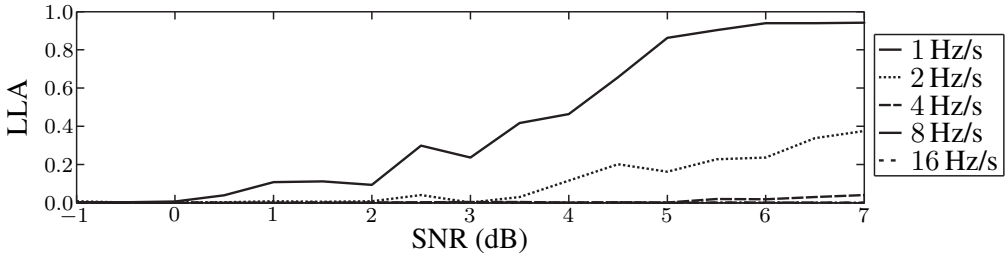


Figure 5.15: The mean line location accuracies of oblique track detections as functions of the spectrogram's SNR. The results were obtained using the potential energy proposed in this thesis and the original internal energy (the active contour's weighted first and second derivatives). The mean standard deviations for the each function is: 1 Hz $= 0.0100$, 2 Hz $= 0.0062$, 4 Hz $= 0.0014$, 8 Hz $= 0.0002$ and 16 Hz $= 0.0000$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.9.

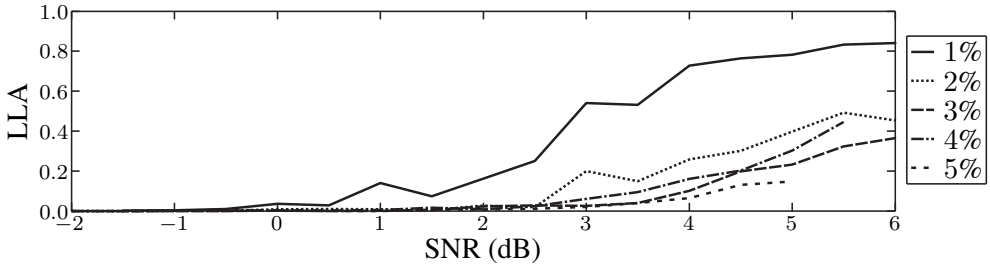


Figure 5.16: The mean line location accuracies of sinusoidal (ten second period) track detections as functions of the spectrogram's SNR. The results were obtained using the potential energy proposed in this thesis and the original internal energy (the active contour's weighted first and second derivatives). The mean standard deviations for the each function is: $1\% = 0.0073$, $2\% = 0.0054$, $3\% = 0.0042$, $4\% = 0.0057$ and $5\% = 0.0029$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.10.

achieved when integrating information from multiple locations has a mean LLA that is 0.2747, 0.1245, 0.0403 and 0.0247 higher for the cases of detecting oblique tracks that have gradients of 2 Hz/s, 4 Hz/s, 8 Hz/s, and 16 Hz/s respectively.

A similar trend is observed in the detection of sinusoidal tracks that have a period of ten seconds, the results of which are presented in Figure 5.16. At a centre frequency variation of 1% the detection of single tracks outperforms that of multiple tracks by a mean LLA of 0.1206 at SNRs above 3.5 dB. Below a SNR of 3.5 dB, however, the integration of information from multiple track positions drastically outperforms the single track detection by a LLA of 0.3355. When the track has greater centre frequency variation, the single track detection strategy falls behind the multiple track detection strategy, which results in mean LLAs that are 0.2497, 0.2456, 0.2318 and 0.2294 higher for centre frequency variations of 2%, 3%, 4% and 5% respectively.

As the period of the sinusoidal track increases, the detection of harmonic tracks, in addition to the fundamental track, follow a similar pattern; excluding tracks that have a centre frequency variation of 1% and that have a high SNR (greater than 4 dB when the period is 15 seconds and greater than 3 dB when the period is 20 seconds), integrating information from multiple locations

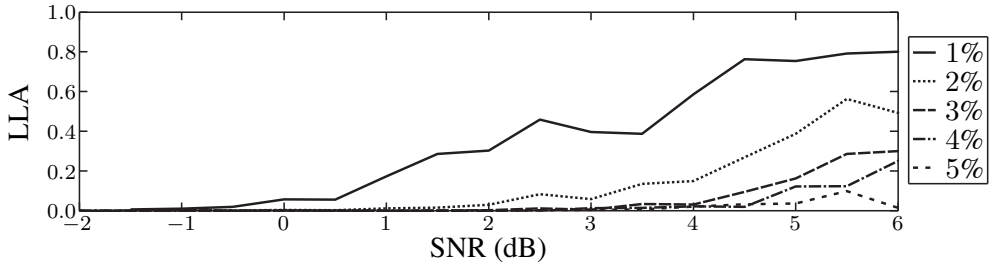


Figure 5.17: The mean line location accuracies of sinusoidal (fifteen second period) track detections as functions of the spectrogram's SNR. The results were obtained using the potential energy proposed in this thesis and the original internal energy (the active contour's weighted first and second derivatives). The mean standard deviations for the each function is: $1\% = 0.0069$, $2\% = 0.0060$, $3\% = 0.0039$, $4\% = 0.0031$ and $5\% = 0.0021$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.11.

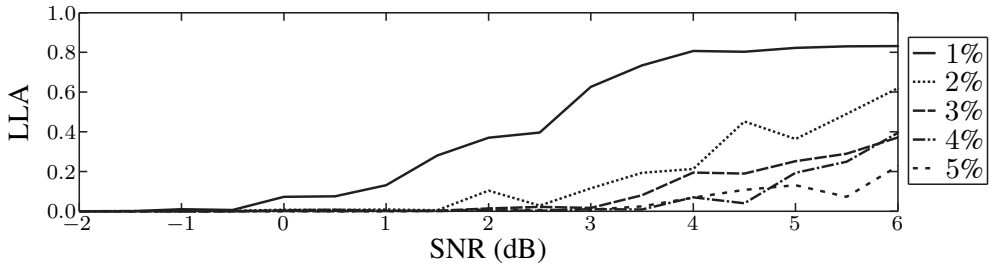


Figure 5.18: The mean line location accuracies of sinusoidal (twenty second period) track detections as functions of the spectrogram's SNR. The results were obtained using the potential energy proposed in this thesis and the original internal energy (the active contour's weighted first and second derivatives). The mean standard deviations for the each function is: $1\% = 0.0063$, $2\% = 0.0074$, $3\% = 0.0049$, $4\% = 0.0039$ and $5\% = 0.0028$, a full list of the standard deviations is presented in Appendix A.2.6, Table A.12.

outperforms the detection of single fundamental tracks in all the experiments, the results of which are presented in Figure 5.17 and Figure 5.18. Sinusoidal tracks having a period of fifteen seconds are detected with a mean LLA difference of: 0.2010, 0.3027, 0.2804, 0.2778 and 0.2567. Sinusoidal tracks having a period of twenty seconds are detected with a mean LLA difference of: 0.1866, 0.3274, 0.3009, 0.2993 and 0.2810. A number of examples of detections that are the result from this variant of the algorithm are presented in Appendix A.2.5, Figures A.27 and A.28.

Fewer false positive detections have occurred during this experiment than occurred when integrating information from harmonic locations. This is due to the necessity of increasing the gradient potential, which in turn reduces the number of true positive detections.

5.5.2 Discussion

Predominantly, integrating detections from harmonically related positions in the spectrogram enhances detection rates. Very low false positive rates have been produced during the single contour experimentation and in most cases no false positive detections were observed at all. This would

seem to contradict the proposition that integrating information from harmonically related positions reduces the potential energy's response to false positive detections, however, it is necessary to increase the gradient potential in the single contour case to realise the algorithm's maximum performance upon the test set. This consequently reduces the false positive rate, but has the disadvantage of reducing the contour's sensitivity to low SNR tracks. Unexpectedly, however, the enhancement of detection rates is not always realised; in some cases, particularly with near vertical tracks at high SNRs, the single contour detection capabilities, reflected in its LLA, outperform those of integrating information from multiple positions. It seems, however, that this is misleading. The proportion of true positive detections presented in Appendix A, demonstrate that integrating information from harmonically related positions increases the number of true positive detections. It is in fact the extremely low rates of false positive detections, realised by increasing the gradient potential of the single contour, which increase the LLAs.

It is obvious throughout the results that integrating information derived from harmonic locations offers a large amount of resilience to SNR degradation, independently of the track's structure.

5.6 Further Discussion

After analysing each experiment in detail it is now possible to discuss some general findings with respect to the proposed algorithm.

5.6.1 Active Contour Algorithm

It has been shown during optimisation that the value of the parameter α has very little influence on the detection performance of the algorithm. This was observed in two variants of the algorithm, each using one of the two potential energies discussed in this thesis. It is not possible to conclude the reason for this behaviour without additional investigation, however, it could potentially be due to an overlap of the salient information captured by the first and the second derivatives (the continuity and curvature) of the active contour. The parameter β behaves similarly in both algorithm variants, the key difference is that the maximum of the LLA is reached at a higher value in the original variant when compared with the Perrin variant, and this indicates two possibilities: the first that the Perrin energy captures more salient information and therefore its influence does not need to be as strong as the original internal energy; the second, the opposite, the Perrin energy does not capture the correct information and so it is beneficial to reduce its influence. The experimental evidence favours the former; the Perrin variant achieves similar or higher LLAs compared with the original variant at the higher SNRs and therefore, even whilst having a lower influence on the contour's energy, it produces comparable results. Generally, however, high weighting of the internal energies restricts the contour's ability to deform and evolve, resulting in a large degradation in the detection performance and an increase in the probability of false positive detections. The gradient potential, as designed, acts as a means of controlling false positive detections; a high gradient potential force reduces the probability of false positive detections and a low force increases them. It is therefore akin to a threshold and, as such, increasing its value has the concomitant effect

of reducing the number of true positive detections—a balance that should be a point of attention when selecting the parameter's value.

Moreover, the novel potential energy introduces the capability of detecting multiple tracks simultaneously. Integrating over harmonic positions in this way reduces the potential energy's response to false positive detections (as noise is not harmonic), which translates into a reduction in the strength of the gradient potential force, and creates a more finely balanced and sensitive detection strategy. Principally, the active contour algorithm, when utilising the novel potential energy, has proven to be a reliable method for extracting unknown shaped tracks in spectrograms. Moreover, the potential energy proposed in this thesis produces far fewer false positive detections when compared with the original potential energy, promoting its suitability to the problem. Increasing the influence of the original potential energy upon the contour's energy results in an increase in the probability of false positive detections, and the optimal value for the weighting of this energy is below its maximum. Contrarily, as the novel potential energy's influence upon the contour's energy is increased, there is a relatively stable probability of false positive detections.

5.6.2 Relation to Existing Methods

It is now possible to relate this research to existing techniques found in the literature. Chapter 2 presents a review of a number of methods that have been proposed and that are based upon some form of deformable model. Di Martino et al. introduce a number of perceptual track features derived from feature grouping theory [55] and these features are defined to be: frequential curvature regularity; temporal continuity; high average intensity; and high point density. Equivalences to these features are present in the active contour model proposed in this thesis. The frequential curvature, temporal continuity and point density are defined to be the continuity and curvature between pixels in the spectrogram and is synonymous with the internal energies of an active contour. High amplitude is taken to be the pixel's intensity value, which is equivalent to the original internal energy evaluated in this chapter. These features are also utilised in other work proposed by the same authors [54], the primary difference between the two being the method of searching for the instances of pixel groups that fulfil the criteria. The first proposal is to perform an exhaustive search between all pixels in the spectrogram, evaluating their cost function, and selecting those that result in high values. The second uses a self organising map and tests its convergence for the presence of a track using a cost function. Nevertheless, this chapter and the investigation into low-level feature detection presented in Chapter 3, have demonstrated the weakness of such methods when applied to low SNR spectrograms. The results presented have shown that relying upon individual pixel's amplitude results in poor detection performance at low SNRs. Furthermore, the active contour's energy minimisation has a far lower computational burden (whilst ensuring an even search throughout the spectrogram) as the multi-stage decision process conducts an exhaustive search between each and every pixel.

In addition to these methods, Di Martino and Tabbone [56] propose a similar cost function that also incorporates the track's amplitude and the slope between two pixels (an approximation to the first derivative), but from which the curvature (the second derivative) is removed. Initially, candidate locations of the spectrogram are identified for further processing within a stage that

applies Gaussian filtering. In an analogous fashion to the authors' previous work, the cost function subsequently groups pixels within the identified regions into track structures. This simplifies the algorithm's complexity by removing the need for exhaustive searches, however, it introduces the caveat that the subsequent pixel grouping is dependent upon the power of the initial detection process; any weak tracks that are overlooked in the initial stage are removed from the solution. This caveat is circumvented by the energy minimisation process drawn on by the active contour, which ensures an even search throughout the spectrogram. To impart dispersion of knowledge, this, and the author's previous work, could benefit from the insight into the description of curvature and continuity of a feature that is afforded by the numerous contributions to the active contour research area. Moreover, it is possible to augment the simplistic features of individual pixel values used to identify tracks by these methods by the potential energy proposed in this thesis.

Methods which make use of the hidden Markov model [190, 137] maximise the probability of a track based upon the observation (the current row of a spectrogram) and the model's state transition probabilities. Track structures that are unlikely, are therefore unaccounted for in this representation, can be mismodelled. To overcome this limitation, a solution using multiple track models, to be used in parallel, has been proposed [175]. The additional complexity resulting from this solution is avoided with the active contour algorithm as its internal energies afford great flexibility. Moreover, the transition matrix of the hidden Markov model can be loosely interpreted as a probabilistic method for learning the form of the internal energy that is applicable to the problem. As such, maximising the probability is therefore tantamount to minimising a cost function that defines the permitted model deformation. The distinct backgrounds of the hidden Markov model and the active contour prevent ready insight into the possibility of transference between the two algorithms. Nonetheless, the potential energy proposed by this thesis was first defined in Chapter 3 as a standalone low-level feature detector that outputs the probability of detection. As such, it should be possible to use this low-level feature detector in conjunction with the hidden Markov model to increase the reliability of the existing hidden Markov model solutions presented in Section 2.3.4.2, which derive probabilities from single pixel values.

Correlation methods such as that presented by Altes [8] are classed as 'optimal' detectors, as defined in Chapter 3. These methods test hypotheses by correlating a template, or reference spectrogram, with the spectrogram being analysed. As is shown in Section 3.2, correlation methods are computationally expensive as they form large search spaces in which the true detection needs to be located. The active contour model can be interpreted as a flexible correlation method, which, because of deformation and efficient energy minimisation (see Section 4.3) removes the computational burden associated with correlation based detectors.

Unfortunately, it is not possible to directly compare the results presented in this chapter to those presented by Di Martino and Tabbone [57], who first used the line location accuracy measure in this application, as the value of the LLA parameter λ used to derive their results is not known.

5.6.3 Line Location Accuracy

The line location accuracy has been used to optimise the parameter values and to measure the performance of the algorithms during the evaluation presented in this chapter. This performance

measure aggregates three pieces of information: the true positive rate; the false positive rate; and the location of the true positive detections. As such, optimisation using this measure forms a balance between the three detection criteria. One way in which this balance could manifest itself would be to improve true positive detections at the expense of false positive detections or detection location accuracy. Predominantly, this measure has lead to the selection of parameter values that provide acceptable performance. Nevertheless, fewer false positive detections could be obtained by raising the gradient potential force above the value that the LLA measure indicates is optimal (obviously this would have consequence upon the number of true positive detections). Moreover, depending upon the specific application, some of the aspects of the LLA measure may be more important than others. For example, in some applications it may be more desirable to identify the presence of a source in a particular time frame. To know that the algorithm has detected the source at a specific frequency may be secondary or unnecessary. Relaxing conditions such as this affords the tuning of parameters to maximise true-positive detections at the expense of location accuracy.

5.7 Summary

This chapter has presented a thorough evaluation of the spectrogram track detection algorithm presented in the previous chapter. In doing so, the benefits of the novel potential energy, the internal energy proposed by Perrin and Smith and the detection of a harmonic series of tracks, when applied to the problem of spectrogram track detection, were each evaluated. It has been determined that the Perrin internal energy achieves, and often surpasses, the LLA achieved by the original internal energy at the higher SNRs. As the SNR degrades further, however, the original internal energy produces higher LLA scores. Nevertheless, the probability of false positive detections that occur using the Perrin internal energy are over a factor of ten lower than those occurring with the original internal energy. It is therefore possible that the original internal energy produces better localisation results and that the Perrin internal energy, although producing greater specificity, overruns the target contour, which is a common condition when introducing forces such as the gradient potential.

The novel potential energy proposed in this thesis greatly improves the algorithm's detection capabilities. It has been shown that using the original potential energy formulation results in very low LLA. This, in conjunction with the high probability of false positive detections that also occur, indicate that the algorithm fails to detect anything meaningful in the data. The integration of information from harmonic locations also proves to increase the LLA.

Finally, a discussion has been presented that relates the findings of this chapter to existing work in the area. It has been shown that the multi-stage decision processes that optimise cost functions are similar to the energy minimisation used in the proposed algorithm. The energy minimisation technique used by the active contour, however, is far less computationally expensive whilst ensuring an even search throughout the spectrogram. It has also been proposed that the amplitude features that these cost functions depend upon could be substituted for the proposed potential energy to enhance the method's detection rates. It has also been discussed that the measures of continuity and curvature employed by the cost functions could benefit from research into

the forms of the active contour internal energies. The proposed algorithm has also been related to existing detection methods that utilise the hidden Markov model and it was proposed that the low-level feature detector that the potential energy is based upon could also enhance the detection rates of these algorithms.

Chapter 6

Conclusions

"On the mountains of truth you can never climb in vain:
either you will reach a point higher up today,
or you will be training your powers
so that you will be able to climb higher tomorrow."

— Friedrich Nietzsche, 1844–1900.

This thesis tackles the problem of detecting non-stationary quasi-periodic phenomenon in time-series data. This problem is expressed as the detection of tracks in spectrograms, which finds application in many remote sensing problems, and a formal definition of this problem is outlined in Section 1.2. The research presented is preceded by a taxonomy, review and survey of existing algorithms from the literature (see Chapter 2), which has led to the identification of shortfalls in current research, and has motivated solutions to these issues. Chapter 3 addresses one such shortfall by presenting a full investigation into low-level feature detection. Subsequently, a novel high-level detection algorithm based upon the active contour algorithm, which allows for flexible modelling of unknown track structures, is presented in Chapter 4. The active contour algorithm finds parallels with some aspects of the existing research applied to this problem and overcomes some of their limitations (see Section 5.6.2). This high-level algorithm integrates the findings of Chapter 3, namely the low-level feature detection methods and the harmonic integration, into an energy minimisation process. The strengths and weaknesses of the proposed algorithm are empirically evaluated in Chapter 5, where it is applied to the detection of tracks in a number of test scenarios. Additionally, Chapter 5 evaluates the benefits of each of the novel solutions that have been proposed in this thesis through an empirical comparison to the original algorithm [96].

This thesis' main conclusions are:

- It is concluded in Chapter 2 that many of the existing algorithms applied to the problem of spectrogram track detection rely upon simple low-level feature detection mechanisms;
- Chapter 3 shows that low-level feature detectors based upon single pixel values produce unreliable detection results;

- Chapter 3 also demonstrates that integrating spatial and structural information into the low-level detection process increases detection reliability;
- Utilising dimensionality reduction techniques during low-level feature detection has been shown in Chapter 3 to reduce computational burden but also to reduce detection rates;
- Integrating information from harmonic locations within the spectrogram reduces the variance of pixel values, and therefore, improves the reliability of low-level feature detection (see Section 3.4 and Section 5.5);
- The internal energy proposed by Perrin and Smith [141] closely models the track structure that can be observed in spectrograms (see Section 4.2.3) and results in slight improvements in the detection rates when compared with the original internal energy formulation (see Section 5.3);
- The active contour's original potential energy, which relies upon distinctions between single pixel values, is not suitable in this application (see Section 3.3.2) and fails to achieve reliable detection results when applied to detecting tracks in spectrograms (see Section 5.4);
- The potential energy proposed by this thesis (see Section 4.2.2), which is based upon low-level feature detection strategies (see Chapter 3), effectively facilitates the active contour's application to the problem of spectrogram track detection (see Sections 5.3, 5.4 and 5.5).

Following from these, the active contour algorithm proposed in Chapter 4 has fulfilled the thesis proposition (see Section 1.3) as an effective method, which incorporates and extends existing methodologies for detecting tracks that have a wide variety of structural configurations at low SNRs. Empirical and theoretical evidence for this claim is presented throughout this thesis. Therefore, in general, this is a new application of a developed idea, which leads to new results. In conducting this research several obstacles have been encountered, and a reflective discussion follows.

Existing research on spectrogram track detection has been found to lie in disparate areas of computer science and mathematics (see Table 2.2 in Section 2.2). Moreover, much of the research has been conducted in conjunction with governmental agencies, with limited data, and is subject to sensitivity clauses. As such, there existed no coherent and encompassing work that fully described the state of the art and the field itself. This offered the opportunity to survey research that is applied to the detection of tracks in spectrogram images and to advance an original and important contribution to the research community, this contribution is presented in Chapter 2.

The disparity of existing research has made it hard to determine equivalences in existing methods, to view it as a whole, and consequently, to ascertain a clear initial research direction. The locus of the initial investigation in Chapter 3 is therefore centred on the lowest level of abstraction afforded when posed as a computer vision and pattern recognition problem—low-level feature detection [71]. From this viewpoint, statistical machine learning techniques based upon the simplest decision boundaries [60] have been applied and limitations discussed (see Section 3.1.1). Increasing the amount of information available to feature detection mechanisms with increasing complexity has allowed their construction in a systematic and rigorous way, *ab initio* (see Sections 3.1.2 and 3.1.3). Mathematical concepts like dimensionality reduction [92], and machine

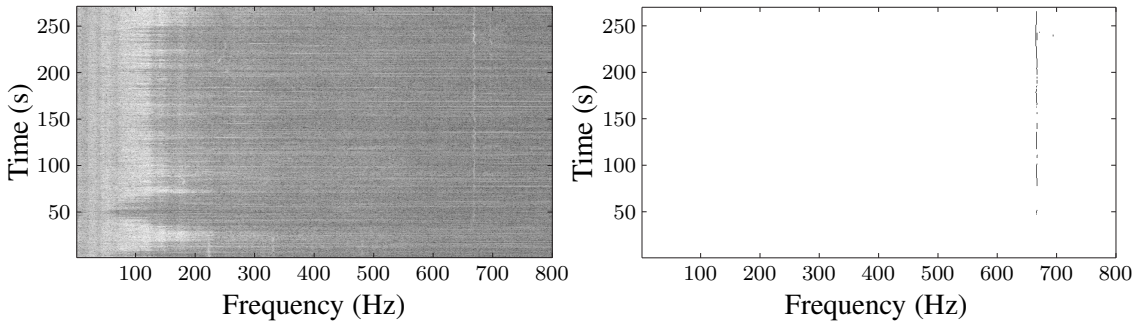
learning techniques [129] like neural networks [28] are applied to strike a balance between model specificity and complexity (see Section 3.2).

The process of this low-level investigation allows for the algorithms presented in the literature review to be thought of in terms of pattern recognition processes, and for their low-level feature detection capabilities to be related to the evaluated strategies (see Section 2.5). Under this light, a majority of the existing methods have been found to utilise very basic low-level feature detection, many of which utilise information derived from single pixels and ignore the spatial and structural information which exists in the spectrogram (see Chapter 2, in particular Section 2.5).

Subsequently, the active contour [96] is developed for use as a high-level [71] track detection algorithm (see Chapter 4). Commonalities with existing algorithms that have been applied to this problem are identified in Section 2.5 and Section 5.6.2, and limitations that it overcomes are identified in Sections 2.4.2 and 5.6.2. More specifically, the proposed algorithm offers: the ability to perform an even search of the whole spectrogram (see Section 4.2.1) at low computational burden (see Section 4.3); the ability to model rapidly varying and unknown structure (see Section 4.2.3); and allows for the integration of low-level feature detection methods (see Section 4.2.2). An even search of a spectrogram is ensured by the introduction of a gradient potential, however, this can also cause localisation errors to occur [89] as the contour can overrun the true position (see Section 5.3.3). The technique's ability to generalise to unobserved cases has been maximised in Section 4.2.2 by explicitly modelling the noise distribution and not the track class. Consequently, the proposed algorithm solution has been shown in Chapter 5 to allow for the detection of a great variety of track structure using one model. This dissection of ideas portrays the intricate nature of research in the this field: interdependencies are inherent and non-linear in nature, lending to complex solutions.

Due to the nature of the real-world data, and the absence of a publicly available data set, the development and evaluation of the algorithm (see Chapter 5) was achieved using synthetic data. The data set has been designed in collaboration with QinetiQ Ltd. to match real-world data as closely as possible, and opportunities to evaluate the approach using real-world data have proved the algorithm's viability. The data set described in Section 3.3.1 has been designed to contain low signal-to-noise ratio tracks to determine the limitations of the algorithm (see discussions in Chapter 5). Nevertheless, even the most carefully designed synthetic data set is no substitution for real-world examples and, therefore, aspects of an algorithm developed as such should be further developed to guarantee its suitability. To facilitate this, the development of the algorithm in Chapter 4 has been described in a modular approach, the necessary conditions for the substitution of components of the algorithm, such as the potential energy, have been outlined (see Section 4.2.2). Developments of this sort are further discussed in the 'Future Work' section below. The algorithm's applicability to real-world spectrograms in its current state is, however, demonstrated in Figure 6.1 by applying it to the detection of a track produced by a fishing vessel.

Nonetheless, there are advantages for the use of a synthetic data set: its use has allowed for the evaluation presented in Chapter 5 to be performed upon an extensive data set which spans the wide range of test cases described in Section 3.3.1, far grater than would be afforded with real-world data. In addition to this, accurate ground-truth data is readily available (again in Section 3.3.1),



(a) An example of a real spectrogram image where intensity represents $\log(\mathbf{S})$.

(b) The detection resulting from using the active contour algorithm, where the parameter values are $c = 0.17$, $\alpha = 0.10$, $\beta = 0.20$ and $\gamma = 1.00$.

Figure 6.1: An example of real-world track detection. The active contour model used to produce this result utilised the original internal energy and the novel potential energy proposed by this thesis.

which would also not be the case with real-world data. Evaluating the algorithm in this way has added weight to some of the theoretical arguments put forward by this thesis, such as the development of the low-level feature detector (see Chapter 3), gradient potential (see Section 4.2.1) and internal energy model (see Section 4.2.3). Quantitative results have been presented in Chapter 5 to encourage the comparison of results between research. Many of the papers reviewed in Chapter 2 present qualitative results (see Section 2.3), for example Di Martino et al. [55], Scharf and Elliot [162], Streit and Barrett [169], Shin and Kil [165], using data that is not available for comparison. To further encourage the dissemination of results, the data set and experimental code used throughout this investigation accompany this thesis (please refer to the included DVD for details).

A consequence of the lack of quantitative results is that no standard metric existed for the evaluation of such an algorithm. The few researchers who present quantitative results have adopted the Line Location Accuracy (LLA) measure [145] (see Section 5.1), which provides an objective aggregation of the number of true positive detections, false positive detections, and a measure of the detection's location accuracy. Issues relating to this aggregation are discussed in Section 5.6.3, predominantly, however, the measure produces desirable balances between the aggregated parts (see Section 5.3).

Empirical evidence using the line location accuracies derived from direct comparisons, in conjunction with theoretical analyses, has demonstrated the necessity and benefits of the novel solutions, as shown in Sections 5.3, 5.4 and 5.5. Each aspect of the proposed active contour algorithm has been evaluated in isolation to ensure a fair comparison, and to give credence to the interpretations (also presented in Sections 5.3, 5.4 and 5.5). Moreover, the potential energy employed has undergone extensive evaluation as a low-level feature detector (see Chapter 3). This evaluation has involved the calculation of Receiver Operating Characteristics (ROC) [65] using an extensive data set. The ROC has long been used in signal detection theory [63], and is increasingly being used in the machine learning community to characterise the trade-off between true positive and false positive detections [65]. This measure allows classifiers to be organised and selected

based upon their performance.

To re-iterate what has been stated in the introduction of this thesis, this research has focussed on the detection of tracks in spectrogram images, a problem into which any time series data that contains narrowband quasi-periodic phenomenon can be transformed. Although this research has focussed on the application of passive sonar for the detection of mechanical devices, many other directly related acoustic problems exist: marine mammal monitoring [130, 125], speech formant tracking [163], engine vibration monitoring and wolf population monitoring [61], to name but a few.

6.1 Future Work

Included in this section is a discussion of possible future research directions that have arisen as a result of the research presented in this thesis. Some of the suggestions specifically apply to the proposed algorithm, whilst others are general problems which exist in the research area.

6.1.1 Track Association

In can occur that multiple tracks exist in a spectrogram and that at some point in time they cross each other. High-level processing may require complete tracks to be extracted from the spectrogram and this occurrence complicates the matter. A number of algorithms are able to detect crossing tracks [3, 55, 98], including the one proposed in Chapter 4, however, it is still left to separate them into distinct tracks. This limits the ability of high-level algorithms to uniquely associate the state of each source during and after the crossing occurs and, therefore, over the whole length of the track. Mellema has recently proposed a technique to associate piece-wise sections of a track that has undergone temporal discontinuities, or, to associate multiple simultaneous tracks originating from a common harmonic series [124]. This would logically extend to the case of crossing tracks, however, this has not been explicitly investigated as it falls outside of the scope of this thesis. Integrating this type of track association into existing detection algorithms could improve detection rates and eliminate the need for post-processing of the detections. Existing solutions that currently achieve this are based upon the hidden Markov model and integrate information about the track's gradient into the state representation [137, 190]. The limitations relating to these algorithms have been fully discussed in Chapter 2.

6.1.2 Ambient Noise

A consequence of conducting the algorithm's development on synthetic data is that the potential energy may need to be developed further to account for more complex noise conditions that can be encountered in application. One such condition is the temporal noise variation that may be observed in the sea, a limitation that exists in a number of algorithms present in the literature (see Table 2.3 and the remainder of Chapter 2). Much of the variability in the sea's ambient noise is caused by changing dominant sources [174]. Dominant sources of ambient noise in the sea which affect the frequency ranges of concern (10 Hz–1 kHz) are identified by Urick [174] as

being shipping noise and wind noise. Distant shipping (greater than 1,000 miles away) and local shipping can produce broadband ambient noise in the range of 50–500 Hz [182] (it is also possible that distant storms act similarly [174]). Ambient noise has also been shown to be correlated with local wind speed [142], the direct mechanism causing the noise is still uncertain Urick [174], however, theoretical research has indicated that wind can produce noise in the frequency range of 0.1–1 kHz [174]. In the absence of ambient sources (such as distant shipping) wind speed becomes the dominant factor of the background noise levels [174]. Consequently, ambient noise sources remain relatively constant, noise levels in the sea remain relatively constant, and therefore, the proposed potential energy is sufficient. For a detailed discussion on these and further sources of ambient noise the reader is referred to "Principles of Underwater Sound" by Urick [174].

The effects of temporal variation of the ambient noise on the final spectrogram is still relatively unclear, and the design of the sensing apparatus can greatly influence this effect [174]. A hydrophone located in the sea receives pressure waves from a large radius and the contribution from local weather changes and ambient noise is directional [174]. Therefore, these contributions become averaged over a large volume. Consequently, it can be expected that local conditions will not effect the background noise dramatically, permitting the use of the existing potential energy, and normalisation of the spectrogram [72] can correct for any variations that do occur.

As discussed, the algorithm presented in this thesis does not explicitly model temporal fluctuations of noise. Solutions to this issue that are present in the literature are discussed in Chapter 2. A simple rectification, which is proposed in Section 4.2.2.4, is to update the mean of the distribution to that of the observed data (tracks in the spectrogram account for a very small fraction of the observed data and their effects on the mean could be ignored or accounted for through bias estimation), however, this has not been evaluated and would be an interesting future development.

6.1.3 Clutter

Distinct to the problem of temporal noise variability is the general problem of clutter, a problem which affects all remote sensing mechanisms: radar [78, 87]; passive sonar [174]; and active sonar [64, 15]. Clutter is caused by phenomenon which produce unwanted target-like features in the received sonar or radar data and can be caused, in the underwater environment, by: explosions; earthquakes and volcanoes; shipping; biological noise; and rain [174]. The difficulty attributed to this problem is that its sources, biological in particular, form an unpredictable part of the ambient background [174].

As such, there are three methods to deal with clutter: integrate *a priori* knowledge regarding the clutter into the detection algorithm itself [14, 110]; identify and remove the clutter using a pre-processing stage [118, 14]; or identify false detections resulting from clutter in post-processing [110]. There exists much research into the modelling of clutter in active sonar [70, 12, 132, 18], however, such research is lacking in the case of passive sonar.

The averaging process formulated in the algorithm proposed in Section 4.2.2 reduces the chance of false positive detections resulting from clutter assuming that it is not harmonic in nature. Nevertheless, false positive detections in passive sonar that result from clutter are a real problem for current algorithms and should be the focus of future investigation. One possible solution that

could be applied to detection algorithms, in particular that presented in this thesis, is to explicitly model the track class and augment the noise model with this information (a further discussion is presented in Section 4.2.2.4). This dependency upon a track model, however, would reduce the algorithm's ability to generalise to unknown track structure.

6.1.4 Automatic Determination of Harmonic Features

One of the contributions of this thesis is a track detection algorithm which aims to boost detection rates in low signal-to-noise ratio spectrograms by integrating information from locations defined by harmonic relationships (see Chapter 4). These relationships, the relative frequencies between tonal harmonics and the fundamental frequency, are a result of the mechanical components within a source which are typically identified as being the propulsion and auxiliary machinery (engine, motors, reduction gears, generators and pumps etc.) [174]. Algorithms of this sort can be tailored to function as detection mechanisms for a particular source in the case that harmonic relationships are not defined as integer multiples but as some arbitrary linear relationship. Currently, these harmonic relationships are manually determined, either through observation, or, through analysis of a source's mechanical structure. In remote sensing applications it may not be possible to have *a priori* knowledge regarding a source's mechanical components. Moreover, different operating conditions may excite or inhibit the mechanisms that produce particular harmonics, and therefore, the components that are observed. This complicates the manual identification of a source's identifying harmonics. Machine learning techniques can be applied to this problem, automatically learning the linear relationships of harmonic components that identify the source within varying conditions. One drawback of supervised machine learning is the requirement of manually labelled ground-truth data. If this is not available, there are two approaches to overcome this requirement: utilising unsupervised learning techniques removes the requirement for ground-truth data; or employing supervised learning techniques using noisy, automatically generated, ground-truth data. This noisy ground-truth data can be generated using a detection mechanism that has a high true positive, as well as a high false positive detection rate, which is a common trade-off when performing detection within noisy data. If a suitable supervised machine learning technique is applied, and enough training data is available, the relationships between true frequency components, which are common between multiple observations, are likely to be reliably discovered.

An additional complication in the automatic discrimination of sources based upon harmonic components is that subsets of these components belonging to distinct sources may overlap. The degree to which these overlap will directly influence a system's ability to distinguish between the sources that share common subsets. Multi-objective optimisation can be employed to minimise these effects by determining the optimal combination of components that uniquely identifies each source with respect to all other sources. Thus, optimising the system's ability to discriminate between sources. This type of optimisation problem is an ideal application of supervised machine learning techniques that are able to optimise complex hypotheses. Evolutionary computing methods, such as genetic algorithms, are one such technique [129]. These stochastic search algorithms search a large space of hypotheses, progressively refining multiple competing hypotheses until an optimal solution is found according to a predefined fitness function. As these algorithms

perform searches in large spaces the optimisation can take time. Nevertheless, once the system has been designed, the optimisation is a fully automatic process which is performed off-line and only needs to be repeated when a new set of sources are to be included.

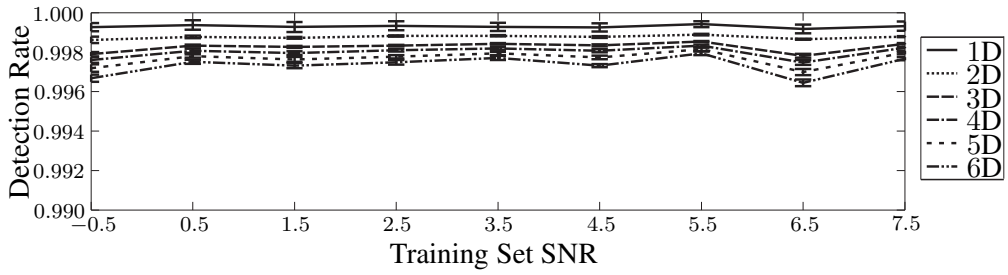
As such, two areas in which the application of machine learning techniques could improve existing systems have been identified as a result of the research presented in this thesis. Namely, the automatic identification of reliable time-invariant features for remote sources, and the optimisation of these features for source discrimination and detection, as discussed in this section.

Appendix A

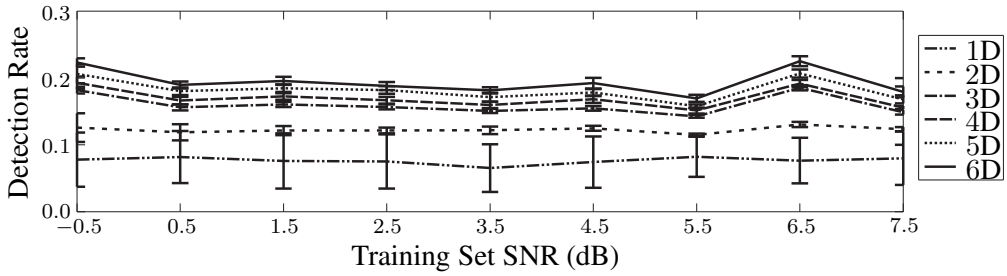
Additional Diagrams

A.1 Chapter 3

In this section of the appendix is presented the additional figures from Chapter 3.

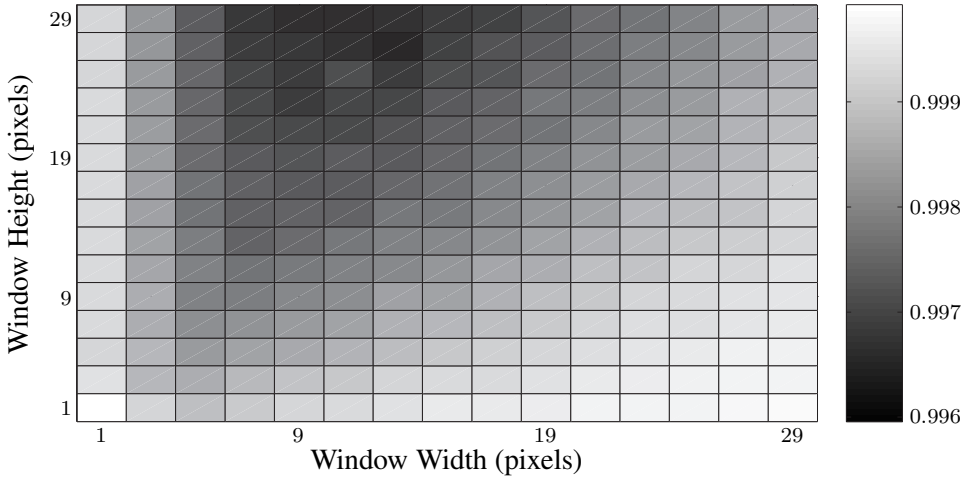


(a) Noise performance.

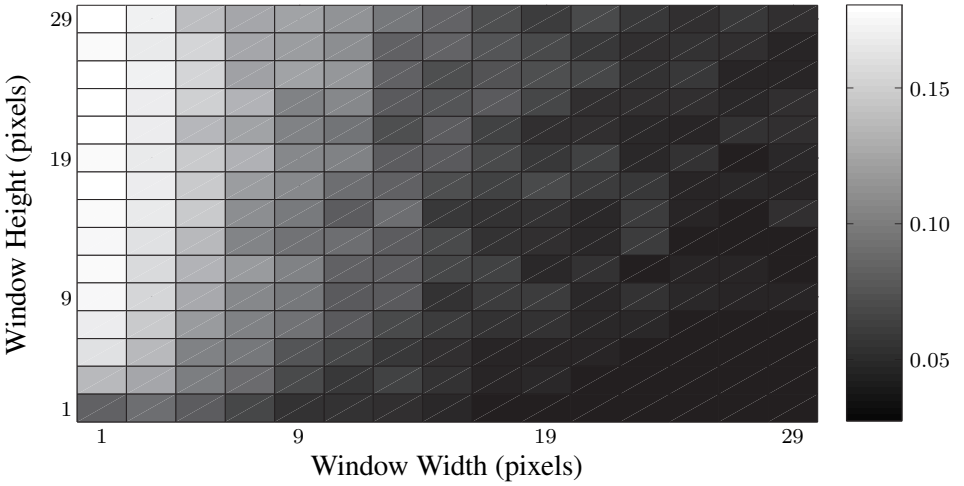


(b) Signal performance.

Figure A.1: PCA low-level feature detection performance as a function of the training set's SNR (SNRs have been rounded to the nearest 0.5 dB). The training sets consisted of 1,000 samples of each class.



(a) Noise performance.



(b) Signal performance.

Figure A.2: PCA low-level feature detection performance as a function of the window's height and width. The training set comprised of 1,000 samples of each class, the track class having a SNR of -0.5 dB.

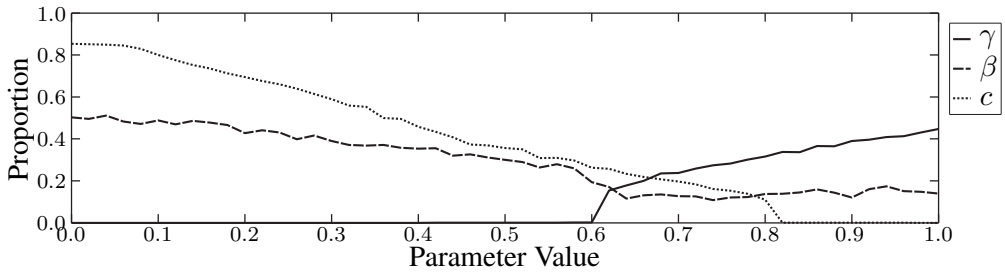
A.2 Chapter 5

In this section of the appendix is presented the additional figures from Chapter 5.

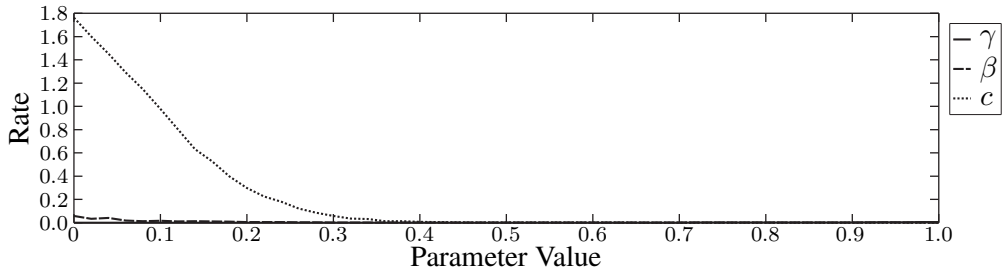
Each figure represents the mean of five repetitions of each experiment in addition to the standard deviation of these repetitions, each repetition utilises a different random training set. Two sets of results are presented for each experiment; the true positive performance, which is the proportion of correct detections, i.e. those that are within five pixels of the true detection, and the false positive performance, which represents the mean probability of additional detections per row of the spectrogram within, plus those outside of, this range. All SNRs have been rounded to the nearest 0.5 dB.

A.2.1 Perrin Internal Energy and the Proposed Potential Energy

This subsection presents the additional figures resulting from the experiments conducted upon the active contour algorithm using the Perrin internal energy and the novel potential energy proposed in this thesis (multiple track detection). These results are attributed to Section 5.3.

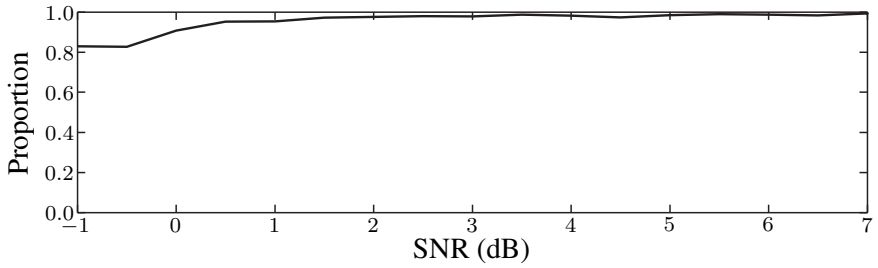


(a) Proportion of true positive detections.



(b) Mean probability of false positive detections per spectrogram row.

Figure A.3: The mean detection performance of the training set as functions of the algorithm's parameter values. Whilst varying each of the parameters the remaining took the following values: $\beta = 0.20$, $\gamma = 1.00$, $c = 0.41$, and the potential energy's window size was taken to be 3×21 pixels.

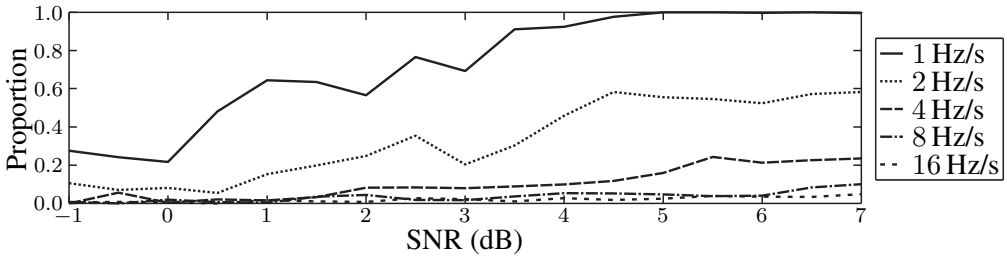


(a) Proportion of true positive detections.

−1.0 dB	−0.5 dB	0.0 dB	0.5 dB	1.0 dB	1.5 dB	2.0 dB	2.5 dB	3.0 dB
0.0221	0.0270	0.0176	0.0103	0.0402	0.0274	0.0244	0.0167	0.0209
3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
0.0183	0.0204	0.0436	0.0180	0.0137	0.0160	0.0216	0.0073	0.0215

(b) Mean probability of false positive detections per spectrogram row.

Figure A.4: The algorithm's detection performance of vertical tracks as a function of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.16$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

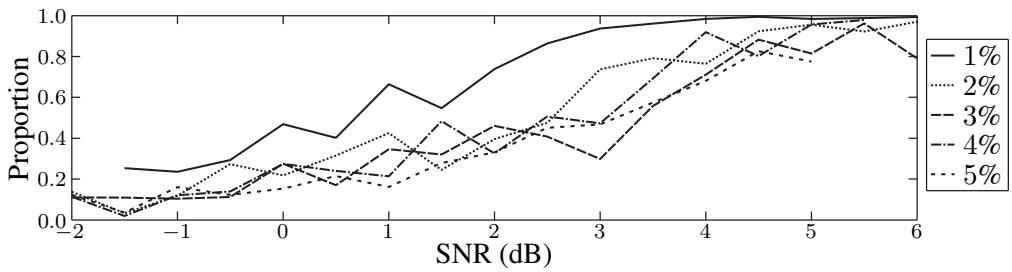


(a) Proportion of true positive detections.

	−1.0 dB	−0.5 dB	0.0 dB	0.5 dB	1.0 dB	1.5 dB	2.0 dB	2.5 dB	3.0 dB
1 Hz/s	0.0000	0.0056	0.0000	0.0119	0.0022	0.0000	0.0011	0.0022	0.0019
2 Hz/s	0.0022	0.0000	0.0061	0.0000	0.0010	0.0078	0.0005	0.0000	0.0000
4 Hz/s	0.0000	0.0039	0.0000	0.0000	0.0000	0.0000	0.0005	0.0000	0.0000
8 Hz/s	0.0067	0.0000	0.0000	0.0000	0.0000	0.0000	0.0062	0.0000	0.0000
16 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0020	0.0000	0.0026	0.0119	0.0000
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0039	0.0000	0.0000	0.0017
2 Hz/s	0.0000	0.0042	0.0162	0.0025	0.0015	0.0000	0.0019	0.0000	0.0026
4 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0061	0.0006
8 Hz/s	0.0000	0.0000	0.0049	0.0030	0.0000	0.0000	0.0047	0.0000	0.0015
16 Hz/s	0.0032	0.0011	0.0000	0.0010	0.0021	0.0010	0.0010	0.0000	0.0015

(b) Mean probability of false positive detections per spectrogram row.

Figure A.5: The algorithm's detection performance of oblique tracks as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.16$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

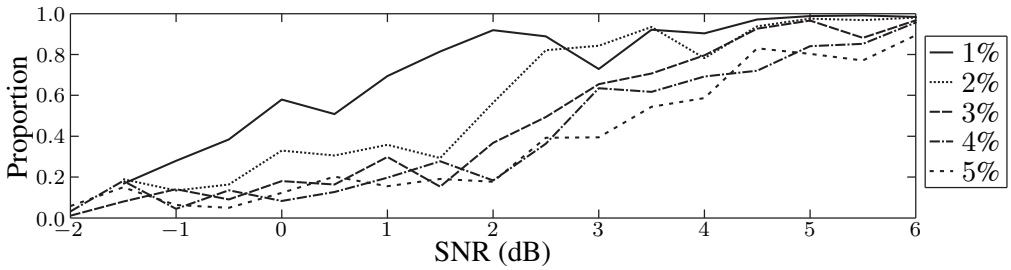


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	0.0272	0.0183	0.0205	0.0537	0.0377	0.0294	0.0046	0.0085
2%	0.0000	0.0071	0.0022	0.0349	0.0606	0.0344	0.0189	0.0020	0.0237
3%	0.0378	0.0121	0.0068	0.0076	0.0416	0.0153	0.0478	0.0340	0.0367
4%	0.0134	0.0006	0.0711	0.0084	0.0581	0.0186	0.0485	0.0785	0.0151
5%	0.0260	0.0037	0.0428	0.0597	0.0443	0.0278	0.0259	0.0471	0.0795
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0127	0.0262	0.0139	0.0125	0.0031	0.0140	0.0137	0.0075	0.0179
2%	0.0077	0.0712	0.0049	0.0846	0.0370	0.0268	0.0295	0.0221	0.0275
3%	0.0302	0.0626	0.0128	0.0899	0.1639	0.2755	0.2578	0.3445	0.0869
4%	0.0270	0.1367	0.1566	0.2234	0.2282	0.3602	0.4030	–	0.1155
5%	0.0655	0.0526	0.1250	0.1493	0.1882	0.2305	–	–	0.0779

(b) Mean probability of false positive detections per spectrogram row.

Figure A.6: The algorithm's detection performance of sinusoidal tracks having a period of ten seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.16$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

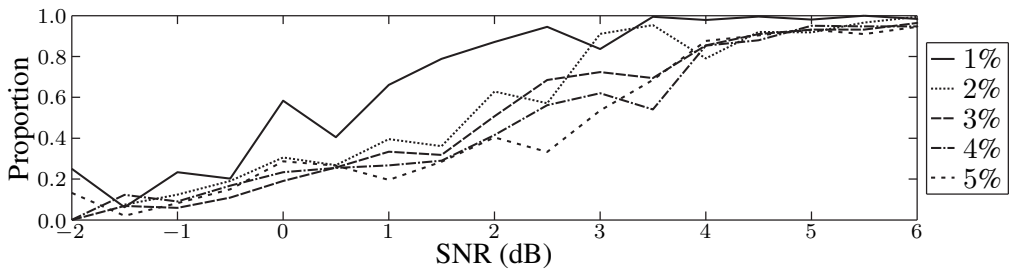


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0009	0.0604	0.0216	0.0127	0.0187	0.0110	0.0294	0.0662	0.0061
2%	–	0.0294	0.0718	0.0319	0.0382	0.0163	0.0509	0.0188	0.0115
3%	0.0084	0.0042	0.0444	0.0200	0.0176	0.0249	0.0276	0.0051	0.0312
4%	0.0067	0.0533	0.0027	0.0306	0.0024	0.0073	0.0048	0.0020	0.0056
5%	0.0173	0.0251	0.0097	0.0028	0.0090	0.0260	0.0210	0.0070	0.0027
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0345	0.0284	0.0000	0.0187	0.0544	0.0150	0.0131	0.0183	0.0241
2%	0.0389	0.0014	0.0099	0.0012	0.0159	0.0067	0.0513	0.0049	0.0249
3%	0.0279	0.0134	0.0146	0.0022	0.0328	0.0171	0.0569	0.1490	0.0292
4%	0.0077	0.0349	0.0072	0.0331	0.0063	0.0558	0.0649	0.0889	0.0244
5%	0.0234	0.0091	0.0241	0.0384	0.0560	0.0190	0.0763	0.0000	0.0216

(b) Mean probability of false positive detections per spectrogram row.

Figure A.7: The algorithm's detection performance of sinusoidal tracks having a period of fifteen seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.16$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.



(a) Proportion of true positive detections.

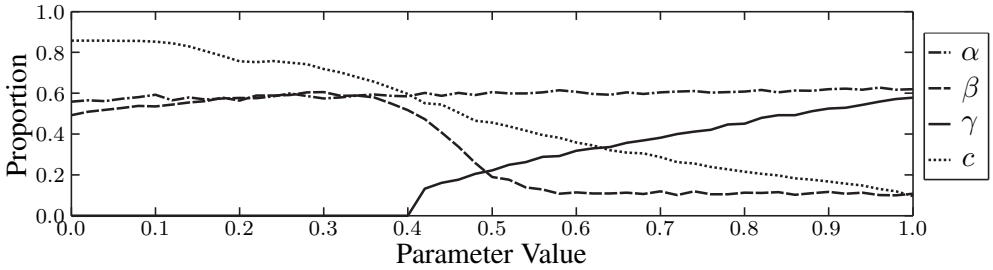
	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0065	0.0000	0.0037	0.0005	0.0327	0.0180	0.0300	0.0106	0.0148
2%	–	0.0032	0.0033	0.0165	0.0018	0.0075	0.0214	0.0003	0.0732
3%	0.0000	0.0132	0.0151	0.0104	0.0405	0.0367	0.0214	0.0211	0.0369
4%	0.0000	0.0536	0.0116	0.0265	0.0235	0.0046	0.0306	0.0116	0.0310
5%	0.0397	0.0031	0.0145	0.0173	0.0389	0.0268	0.0022	0.0109	0.0542
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0048	0.0566	0.0094	0.0293	0.0062	0.0251	0.0002	0.0214	0.0159
2%	0.0092	0.0075	0.0111	0.0217	0.0643	0.0001	0.0371	0.0060	0.0178
3%	0.0468	0.0198	0.0252	0.0498	0.0198	0.0201	0.0081	0.0626	0.0263
4%	0.0091	0.0181	0.0000	0.0891	0.0045	0.0286	0.0367	0.1277	0.0298
5%	0.0393	0.0248	0.0344	0.0750	0.0417	0.0099	0.0030	0.0317	0.0275

(b) Mean probability of false positive detections per spectrogram row.

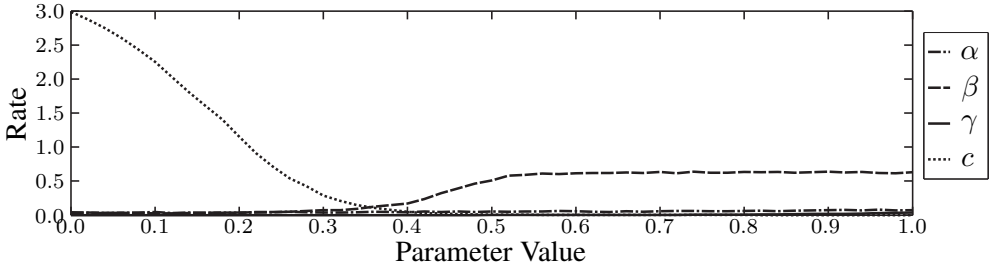
Figure A.8: The algorithm's detection performance of sinusoidal tracks having a period of twenty seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.16$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

A.2.2 Original Internal Energy and the Proposed Potential Energy

This subsection presents the additional figures resulting from the experiments conducted upon the active contour algorithm using the Original internal energies (the weighted first and second derivatives of the active contour) and the novel potential energy proposed in this thesis. These results are attributed to Section 5.3.

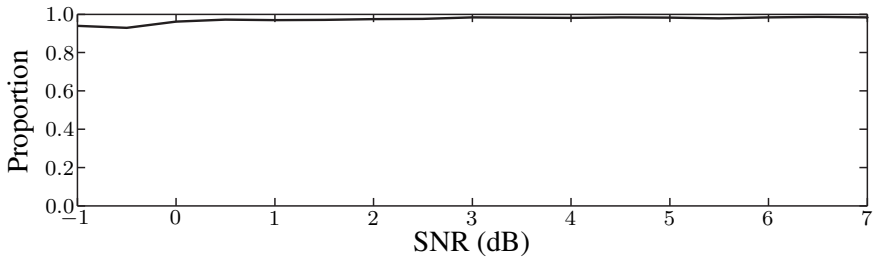


(a) Proportion of true positive detections.



(b) Mean probability of false positive detections per spectrogram row.

Figure A.9: The mean detection performance of the training set as functions of the algorithm's parameter values. Whilst varying each of the parameters the remaining took the following values: $\beta = 0.20$, $\alpha = 0.10$, $\gamma = 1.00$, $c = 0.41$, and the potential energy's window size was taken to be 3×21 pixels.

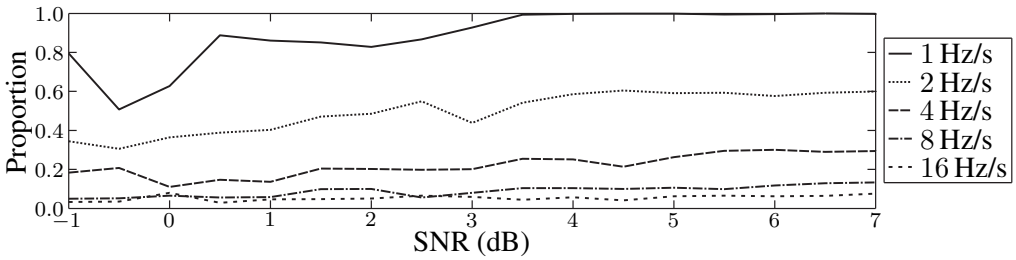


(a) Proportion of true positive detections.

−1.0 dB	−0.5 dB	0.0 dB	0.5 dB	1.0 dB	1.5 dB	2.0 dB	2.5 dB	3.0 dB
0.2855	0.2416	0.2817	0.1794	0.2685	0.2900	0.2531	0.2598	0.1846
3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
0.1796	0.2214	0.2331	0.2151	0.2027	0.1757	0.2139	0.1367	0.2248

(b) Mean probability of false positive detections per spectrogram row.

Figure A.10: The algorithm's detection performance of vertical tracks as a function of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

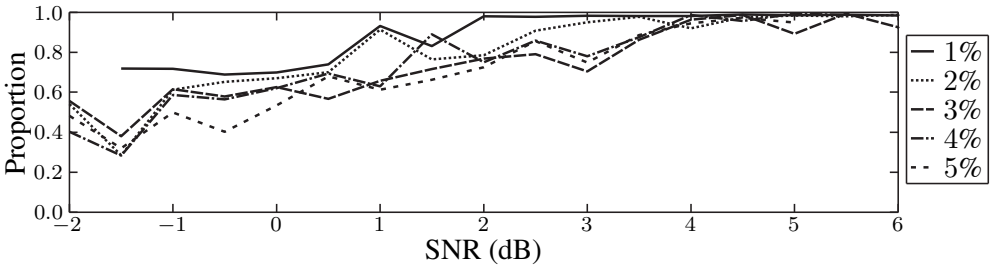


(a) Proportion of true positive detections.

	−1.0 dB	−0.5 dB	0.0 dB	0.5 dB	1.0 dB	1.5 dB	2.0 dB	2.5 dB	3.0 dB
1 Hz/s	0.0472	0.0556	0.0398	0.0786	0.1252	0.1042	0.0517	0.1000	0.0278
2 Hz/s	0.0933	0.0244	0.0661	0.0144	0.0232	0.0511	0.0421	0.0325	0.0012
4 Hz/s	0.0593	0.0361	0.0123	0.0197	0.0111	0.0051	0.0338	0.0183	0.0354
8 Hz/s	0.0106	0.0167	0.0160	0.0144	0.0043	0.0269	0.0179	0.0062	0.0069
16 Hz/s	0.0165	0.0061	0.0238	0.0074	0.0177	0.0124	0.0121	0.0345	0.0038
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	0.0414	0.0315	0.0404	0.0667	0.0472	0.1139	0.0289	0.0383	0.0611
2 Hz/s	0.0259	0.0451	0.1009	0.0549	0.0293	0.0067	0.0426	0.0267	0.0400
4 Hz/s	0.0259	0.0162	0.0051	0.0228	0.0222	0.0111	0.0079	0.0283	0.0218
8 Hz/s	0.0160	0.0141	0.0250	0.0111	0.0111	0.0093	0.0167	0.0222	0.0144
16 Hz/s	0.0087	0.0042	0.0095	0.0181	0.0286	0.0181	0.0219	0.0000	0.0143

(b) Mean probability of false positive detections per spectrogram row.

Figure A.11: The algorithm's detection performance of oblique tracks as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$, $c = 0.36$,, and the potential energy's window size was taken to be 3×21 pixels.

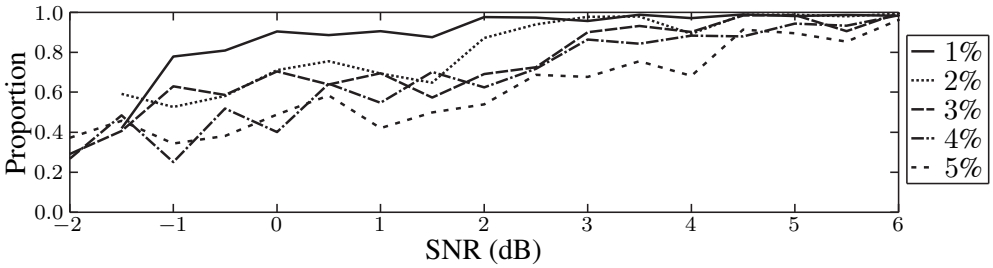


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	0.4192	0.2862	0.2288	0.4326	0.2037	0.3324	0.0965	0.1156
2%	0.1657	0.1052	0.2538	0.3778	0.3428	0.2792	0.2692	0.1064	0.2114
3%	0.3397	0.1674	0.2967	0.2275	0.4025	0.2290	0.3793	0.3191	0.4227
4%	0.2296	0.0898	0.3326	0.2886	0.4055	0.2840	0.3103	0.5899	0.2702
5%	0.3699	0.1280	0.3682	0.2927	0.3641	0.4116	0.2807	0.3914	0.4059
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.1595	0.2831	0.2187	0.2091	0.1032	0.1468	0.2298	0.2027	0.2293
2%	0.1531	0.4138	0.1508	0.3555	0.2838	0.2621	0.3338	0.2476	0.2537
3%	0.3463	0.2643	0.2852	0.4432	0.6242	0.7142	0.6945	0.8301	0.4109
4%	0.3576	0.5103	0.6783	0.7663	0.6416	0.8318	0.6880	–	0.4546
5%	0.4897	0.4345	0.5449	0.6365	0.6422	0.6153	–	–	0.4250

(b) Mean probability of false positive detections per spectrogram row.

Figure A.12: The algorithm's detection performance of sinusoidal tracks having a period of ten seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

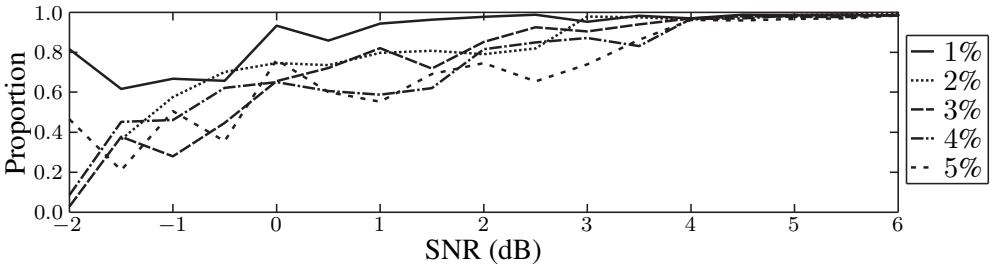


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.1936	0.3451	0.3227	0.3152	0.3455	0.2136	0.2752	0.4633	0.2027
2%	–	0.3485	0.3408	0.2660	0.3267	0.2585	0.2577	0.1652	0.2338
3%	0.1546	0.1719	0.3383	0.2618	0.2636	0.2467	0.2913	0.1130	0.2737
4%	0.1903	0.4007	0.1082	0.3094	0.1276	0.2320	0.1621	0.1918	0.1177
5%	0.2702	0.3173	0.2160	0.1798	0.1952	0.3247	0.1734	0.1630	0.1132
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.3101	0.2161	0.0624	0.2795	0.2241	0.1327	0.1347	0.1890	0.2486
2%	0.3380	0.1195	0.1965	0.0684	0.1983	0.1364	0.3960	0.1233	0.2358
3%	0.3027	0.1801	0.2503	0.1119	0.2188	0.2074	0.3349	0.4755	0.2469
4%	0.1938	0.3285	0.2004	0.3158	0.1693	0.4013	0.3330	0.4901	0.2513
5%	0.2123	0.1306	0.2269	0.2821	0.3083	0.2267	0.3395	0.0836	0.2213

(b) Mean probability of false positive detections per spectrogram row.

Figure A.13: The algorithm's detection performance of sinusoidal tracks having a period of fifteen seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.



(a) Proportion of true positive detections.

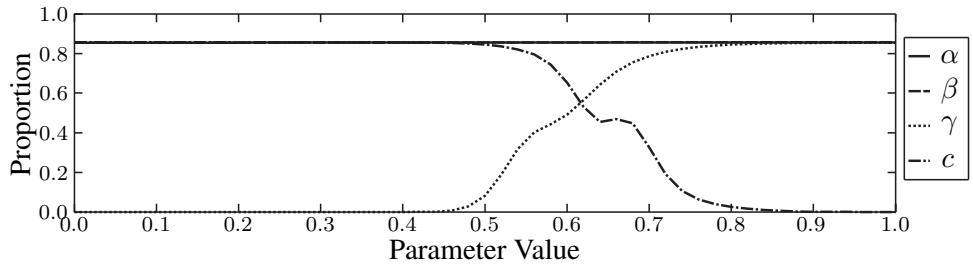
	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.5000	0.0641	0.1493	0.0834	0.3907	0.1518	0.2444	0.1850	0.2088
2%	–	0.1602	0.1859	0.2717	0.2287	0.1850	0.2318	0.1011	0.3380
3%	0.0014	0.1574	0.1345	0.1522	0.2866	0.2917	0.2336	0.1826	0.3155
4%	0.0627	0.3508	0.1861	0.2998	0.3109	0.2880	0.2834	0.2021	0.2988
5%	0.3655	0.0753	0.1989	0.2242	0.4552	0.3412	0.1513	0.2003	0.3598
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0966	0.3267	0.1631	0.2975	0.1473	0.2254	0.0847	0.2247	0.2084
2%	0.1185	0.1988	0.2289	0.1578	0.3262	0.1136	0.1743	0.1031	0.1952
3%	0.3202	0.2612	0.2123	0.3767	0.2640	0.2240	0.1309	0.2288	0.2220
4%	0.2332	0.2265	0.1331	0.4177	0.1707	0.3739	0.2851	0.5549	0.2752
5%	0.1825	0.2927	0.3178	0.4870	0.3485	0.3004	0.0972	0.2825	0.2753

(b) Mean probability of false positive detections per spectrogram row.

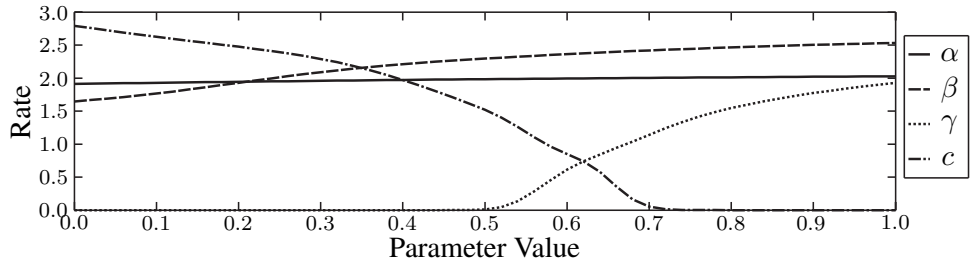
Figure A.14: The algorithm's detection performance of sinusoidal tracks having a period of twenty seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$, $c = 0.36$, and the potential energy's window size was taken to be 3×21 pixels.

A.2.3 Original Internal Energy and the Original Potential Energy

This subsection presents the additional figures resulting from the experiments conducted upon the active contour algorithm using the original internal energies (the weighted first and second derivatives of the active contour) and the original potential energy (the pixel's intensity). These results are attributed to Section 5.4.

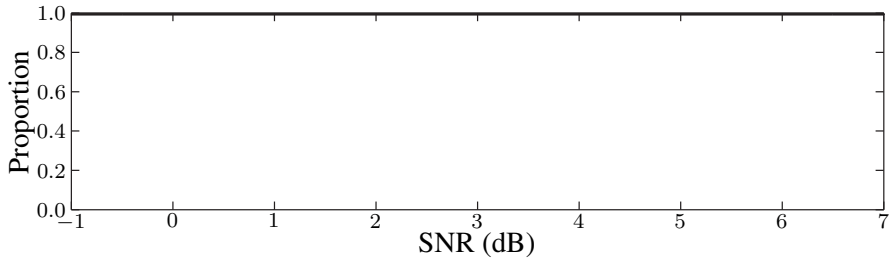


(a) Proportion of true positive detections.



(b) Mean probability of false positive detections per spectrogram row.

Figure A.15: The mean detection performance of the training set as functions of the algorithm's parameter values. Whilst varying each of the parameters the remaining took the following values: $\beta = 0.20$, $\alpha = 0.10$, $\gamma = 1.00$ and $c = 0.41$.

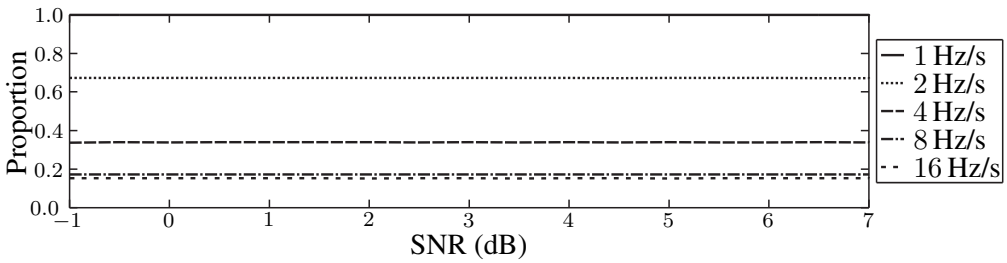


(a) Proportion of true positive detections.

−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
3.6558	3.6418	3.6472	3.6540	3.6577	3.6766	3.6723	3.6596	3.6737
3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
3.7024	3.6973	3.7025	3.7061	3.7134	3.7071	3.7152	3.7203	3.6825

(b) Mean probability of false positive detections per spectrogram row.

Figure A.16: The algorithm's detection performance of vertical tracks as a function of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.66$, $\alpha = 0.50$, $\gamma = 0.82$ and $c = 0.18$.

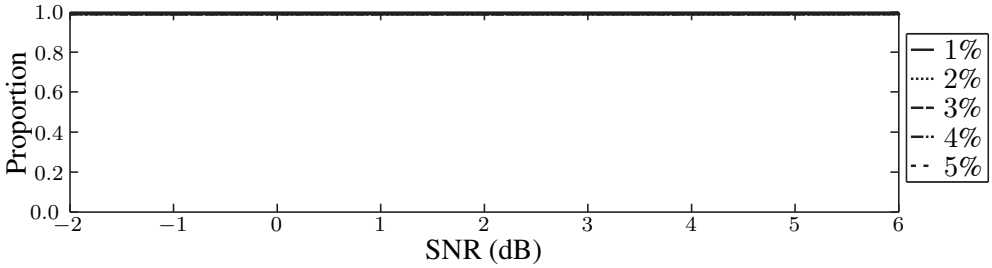


(a) Proportion of true positive detections.

	−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
1 Hz/s	2.1750	2.1361	2.1315	2.1349	2.1311	2.1410	2.1250	2.1100	2.1167
2 Hz/s	1.1267	1.1256	1.1294	1.1222	1.1283	1.1228	1.1106	1.1206	1.1000
4 Hz/s	0.5722	0.5728	0.5753	0.5697	0.5764	0.5773	0.5702	0.5714	0.5667
8 Hz/s	0.2856	0.2917	0.2944	0.2880	0.2827	0.2884	0.2877	0.2840	0.2824
16 Hz/s	0.2528	0.2494	0.2488	0.2476	0.2449	0.2486	0.2545	0.2440	0.2495
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	2.1217	2.0969	2.0980	2.1300	2.1094	2.0994	2.1100	2.0956	2.1213
2 Hz/s	1.1037	1.1007	1.1111	1.1019	1.0965	1.0867	1.0815	1.0800	1.1087
4 Hz/s	0.5722	0.5736	0.5677	0.5756	0.5631	0.5651	0.5698	0.5589	0.5705
8 Hz/s	0.2854	0.2828	0.2792	0.2813	0.2778	0.2801	0.2722	0.2667	0.2830
16 Hz/s	0.2540	0.2519	0.2495	0.2429	0.2476	0.2457	0.2429	0.2476	0.2484

(b) Mean probability of false positive detections per spectrogram row.

Figure A.17: The algorithm's detection performance of oblique tracks as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.66$, $\alpha = 0.50$, $\gamma = 0.82$ and $c = 0.18$.

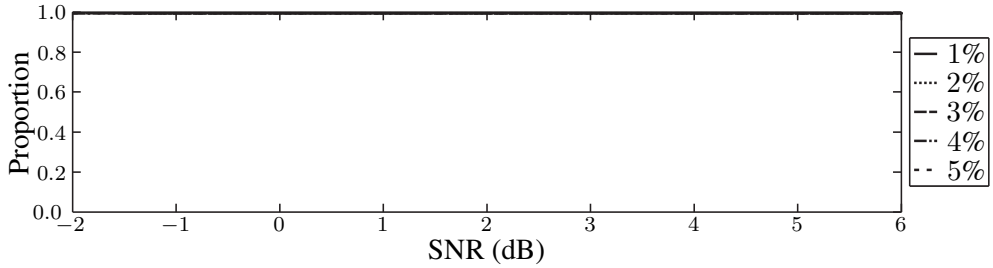


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	3.6481	3.6384	3.6542	3.6496	3.6419	3.6416	3.6308	3.6240
2%	3.5961	3.6357	3.6320	3.6141	3.6320	3.6334	3.6047	3.6192	3.6351
3%	3.5868	3.5984	3.6068	3.5949	3.6013	3.5825	3.5881	3.5884	3.5864
4%	3.5769	3.5788	3.5860	3.5682	3.5543	3.5798	3.5594	3.5644	3.5491
5%	3.6240	3.6233	3.5933	3.6034	3.6160	3.6156	3.6040	3.5985	3.6012
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	3.6285	3.6189	3.6035	3.6372	3.6331	3.6323	3.6348	3.6060	3.6327
2%	3.6047	3.6114	3.6068	3.5875	3.6091	3.5979	3.5958	3.6031	3.6129
3%	3.5735	3.5627	3.5532	3.5692	3.5635	3.5398	3.5474	3.5482	3.5759
4%	3.5495	3.5391	3.5318	3.5364	3.5361	3.5144	3.5487	–	3.5546
5%	3.5944	3.5818	3.5866	3.5745	3.5922	3.5585	–	–	3.5978

(b) Mean probability of false positive detections per spectrogram row.

Figure A.18: The algorithm's detection performance of sinusoidal tracks having a period of ten seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.66$, $\alpha = 0.50$, $\gamma = 0.82$ and $c = 0.18$.

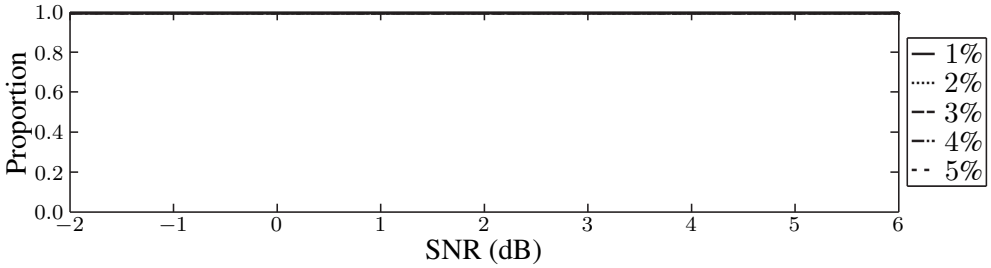


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	3.7047	3.6657	3.6415	3.6444	3.6548	3.6400	3.6357	3.6375	3.6490
2%	–	3.6292	3.6207	3.6426	3.6305	3.6262	3.6285	3.6212	3.6323
3%	3.6026	3.5966	3.6026	3.6149	3.5976	3.5781	3.5933	3.5838	3.6000
4%	3.5929	3.5801	3.5819	3.5738	3.5696	3.5710	3.5750	3.5705	3.5869
5%	3.6123	3.6114	3.6045	3.6196	3.6060	3.6075	3.5942	3.5891	3.6064
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	3.6277	3.6240	3.6360	3.6391	3.6230	3.6235	3.6128	3.6105	3.6394
2%	3.6207	3.6037	3.5955	3.5933	3.5969	3.5951	3.5872	3.5841	3.6130
3%	3.5918	3.5900	3.5880	3.5634	3.5701	3.5875	3.5665	3.5738	3.5883
4%	3.5814	3.5738	3.5584	3.5565	3.5503	3.5587	3.5560	3.5554	3.5701
5%	3.5943	3.5940	3.5802	3.5682	3.5713	3.5741	3.5636	3.5153	3.5889

(b) Mean probability of false positive detections per spectrogram row.

Figure A.19: The algorithm's detection performance of sinusoidal tracks having a period of fifteen seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.66$, $\alpha = 0.50$, $\gamma = 0.82$ and $c = 0.18$.



(a) Proportion of true positive detections.

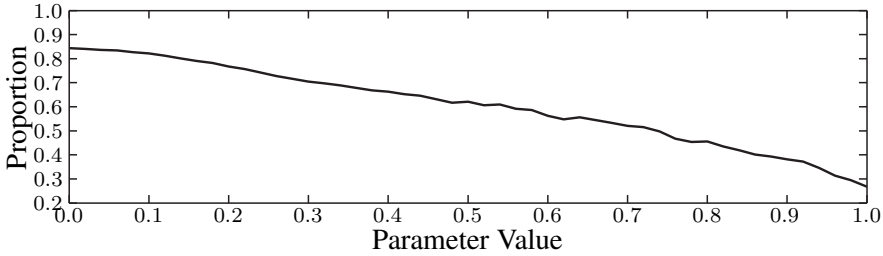
	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	3.6295	3.6602	3.6607	3.6581	3.6536	3.6583	3.6290	3.6290	3.6496
2%	–	3.6233	3.6014	3.6194	3.6113	3.5936	3.5903	3.5983	3.5965
3%	3.6267	3.5571	3.6004	3.6038	3.5914	3.5971	3.5916	3.5958	3.5882
4%	3.5627	3.5891	3.6025	3.5736	3.5877	3.5751	3.5833	3.5832	3.5807
5%	3.6357	3.6195	3.6214	3.6182	3.6037	3.6286	3.6106	3.6061	3.6130
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	3.6400	3.6371	3.6171	3.6202	3.6147	3.6565	3.6158	3.6243	3.6384
2%	3.5956	3.5865	3.5818	3.5680	3.5710	3.5838	3.5682	3.5745	3.5915
3%	3.5917	3.5909	3.5801	3.5967	3.5875	3.5771	3.5741	3.5526	3.5884
4%	3.5788	3.5718	3.5744	3.5816	3.5670	3.5657	3.5616	3.5543	3.5761
5%	3.6125	3.6113	3.6062	3.6070	3.6017	3.5874	3.5938	3.5897	3.6098

(b) Mean probability of false positive detections per spectrogram row.

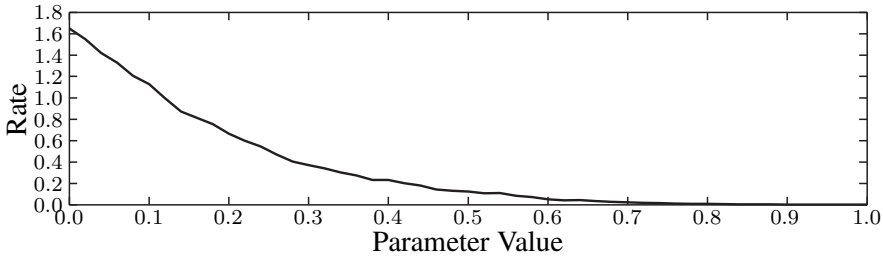
Figure A.20: The algorithm's detection performance of sinusoidal tracks having a period of twenty seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.66$, $\alpha = 0.50$, $\gamma = 0.82$ and $c = 0.18$.

A.2.4 Single Track Detection

This subsection presents the additional figures resulting from the experiments conducted upon the active contour algorithm using the original internal energies (the weighted first and second derivatives of the active contour) and the proposed potential energy when applied to single track detection. These results are attributed to Section 5.5.

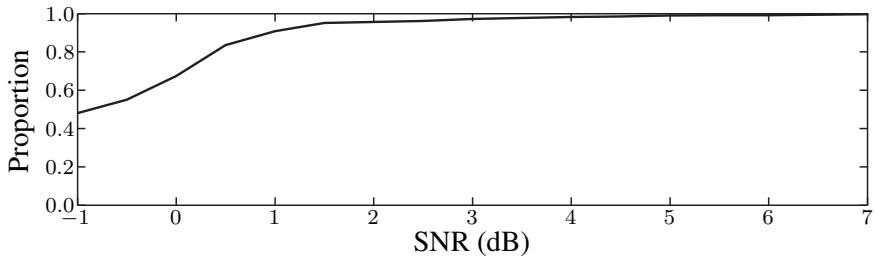


(a) Proportion of true positive detections.



(b) Mean probability of false positive detections per spectrogram row.

Figure A.21: The mean detection performance of the training set as functions of the algorithm's parameter values. Whilst varying the parameter's value, the remaining took the following values: $\beta = 0.20$, $\alpha = 0.10$ and $\gamma = 1.00$.

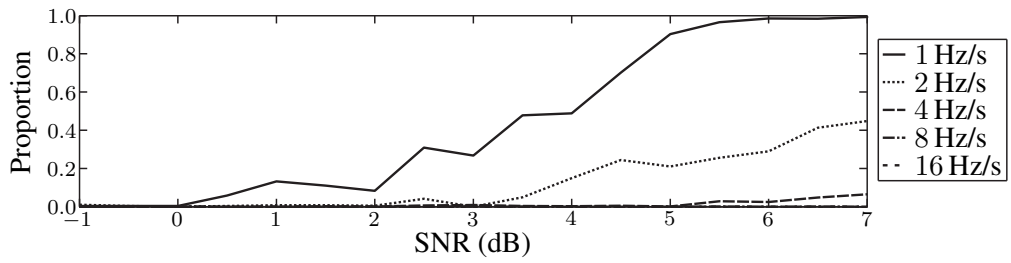


(a) Proportion of true positive detections.

−1.0 dB	−0.5 dB	0.0 dB	0.5 dB	1.0 dB	1.5 dB	2.0 dB	2.5 dB	3.0 dB
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

(b) Mean probability of false positive detections per spectrogram row.

Figure A.22: The algorithm's detection performance of vertical tracks as a function of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$ and $c = 0.74$.

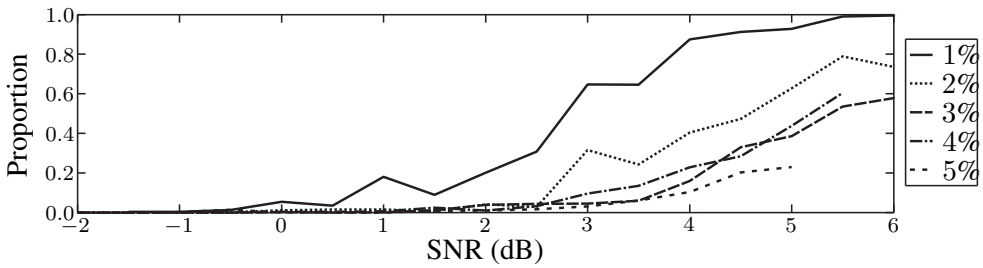


(a) Proportion of true positive detections.

	−1.0 dB	−0.5 dB	0.0 dB	0.5 dB	1.0 dB	1.5 dB	2.0 dB	2.5 dB	3.0 dB
1 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
16 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
16 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

(b) Mean probability of false positive detections per spectrogram row.

Figure A.23: The algorithm's detection performance of oblique tracks as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$ and $c = 0.74$.

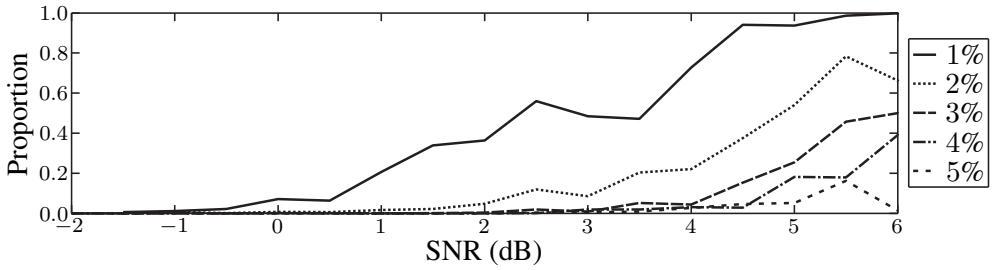


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2%	0.0000	0.0000	0.0000	0.0000	0.0004	0.0000	0.0000	0.0000	0.0000
3%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000
5%	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3%	0.0008	0.0002	0.0001	0.0092	0.0250	0.0471	0.0496	0.1055	0.0140
4%	0.0001	0.0021	0.0039	0.0053	0.0288	0.0586	0.0171	–	0.0073
5%	0.0001	0.0001	0.0017	0.0044	0.0096	0.0193	–	–	0.0024

(b) Mean probability of false positive detections per spectrogram row.

Figure A.24: The algorithm's detection performance of sinusoidal tracks having a period of ten seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$ and $c = 0.74$.

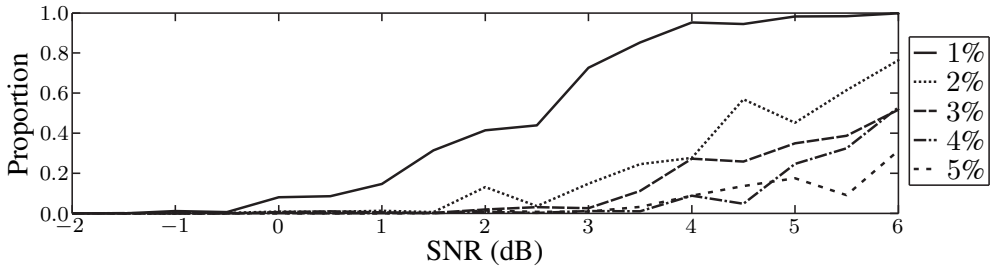


(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2%	–	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4%	0.0000	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3%	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0019	0.0001
4%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0006	0.0001
5%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0000	0.0000

(b) Mean probability of false positive detections per spectrogram row.

Figure A.25: The algorithm's detection performance of sinusoidal tracks having a period of fifteen seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$ and $c = 0.74$.



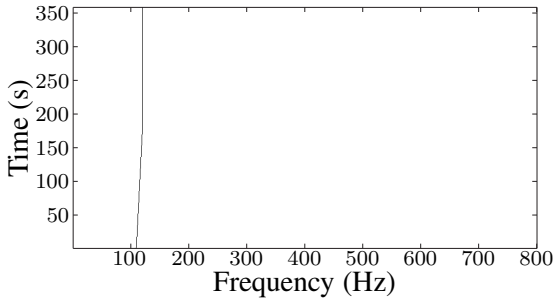
(a) Proportion of true positive detections.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2%	–	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002	0.0000	0.0008	0.0001
5%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

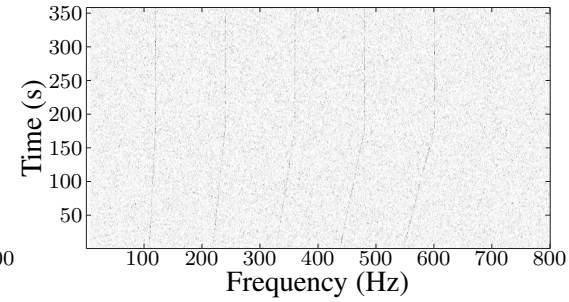
(b) Mean probability of false positive detections per spectrogram row.

Figure A.26: The algorithm's detection performance of sinusoidal tracks having a period of twenty seconds as functions of the spectrogram's SNR in addition to the mean probability of false positives per spectrogram row measured during the experiment. The parameter values used were: $\beta = 0.22$, $\alpha = 0.96$, $\gamma = 1.00$ and $c = 0.74$.

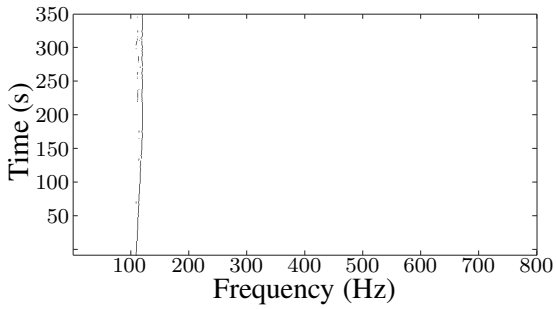
A.2.5 Example Detections



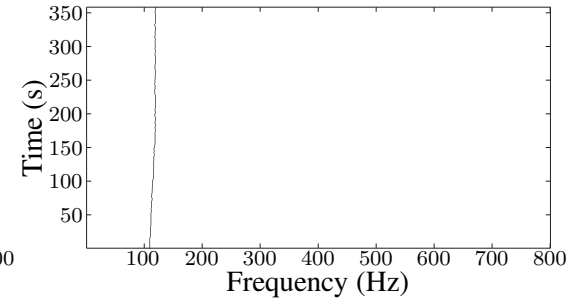
(a) Ground Truth.



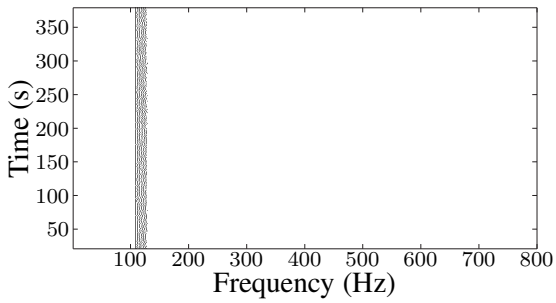
(b) Spectrogram.



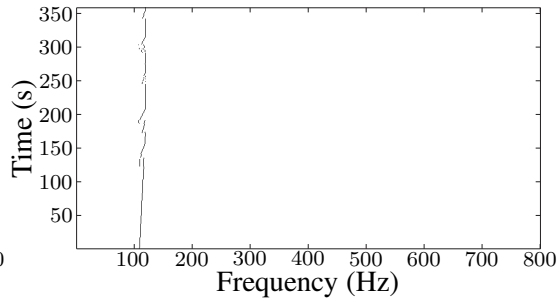
(c) Original internal and novel potential energy. The LLA of the fundamental track detection is 0.7640.



(d) Perrin internal and novel potential energy. The LLA of the fundamental track detection is 0.8025.

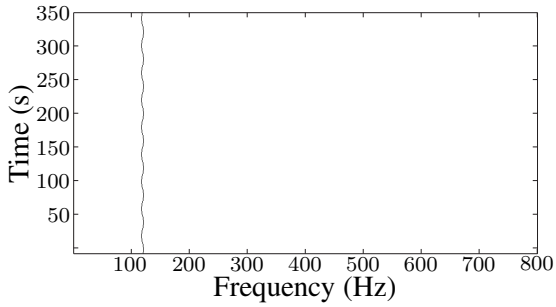


(e) Original internal and original potential energy. The LLA of the fundamental track detection is 0.2100.

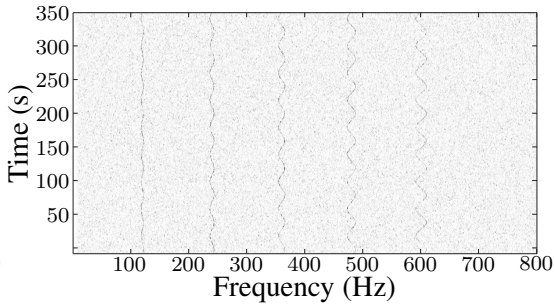


(f) Original internal and novel potential energy, single contour. The LLA of the fundamental track detection is 0.6791.

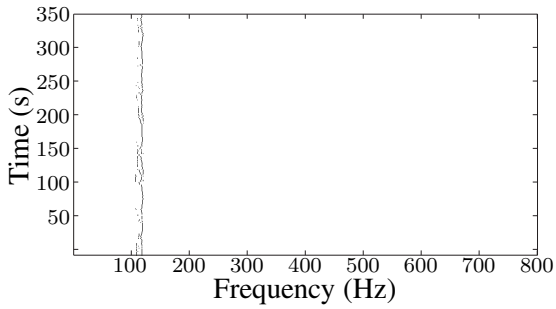
Figure A.27: A set of example detections. The SNR of the spectrogram is 4.5 dB and contains vertical and oblique tracks that have a gradient of 1 Hz/s.



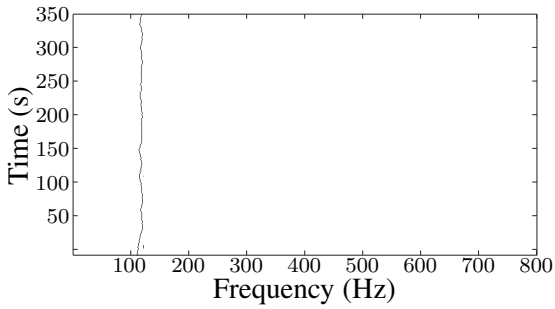
(a) Ground Truth.



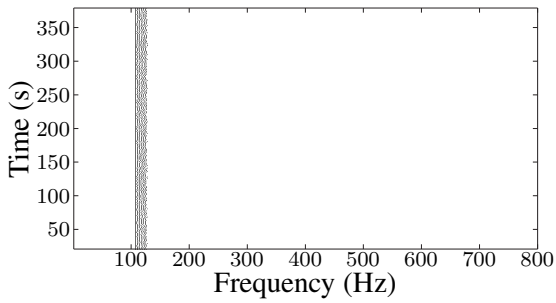
(b) Spectrogram.



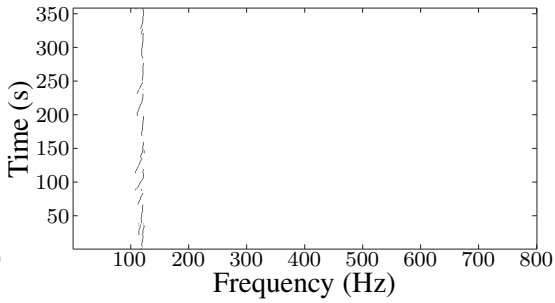
(c) Original internal and novel potential energy. The LLA of the fundamental track detection is 0.3967.



(d) Perrin internal and novel potential energy. The LLA of the fundamental track detection is 0.3658.



(e) Original internal and original potential energy. The LLA of the fundamental track detection is 0.1134.



(f) Original internal and original potential energy, single contour. The LLA of the fundamental track detection is 0.4568.

Figure A.28: A set of example detections. The SNR of the spectrogram is 6.5 dB and contains sinusoidal tracks that have a centre frequency variation of 3%.

A.2.6 Standard Deviations

This subsection presents the standard deviations attributed to the results presented in Sections 5.3, 5.4 and 5.5.

	0.00	**0.02**	**0.04**	**0.06**	**0.08**	**0.10**	**0.12**	**0.14**	**0.16**	**0.18**	**0.20**	**0.22**	**0.24**
α	0.0060	0.0034	0.0057	0.0054	0.0031	0.0042	0.0058	0.0046	0.0050	0.0047	0.0043	0.0057	0.0046
β	0.0039	0.0041	0.0045	0.0045	0.0028	0.0034	0.0060	0.0047	0.0072	0.0019	0.0072	0.0050	0.0072
γ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
c	0.0006	0.0002	0.0004	0.0002	0.0007	0.0005	0.0013	0.0003	0.0017	0.0021	0.0033	0.0071	0.0036
	0.26	**0.28**	**0.30**	**0.32**	**0.34**	**0.36**	**0.38**	**0.40**	**0.42**	**0.44**	**0.46**	**0.48**	**0.50**
α	0.0032	0.0046	0.0088	0.0046	0.0038	0.0039	0.0048	0.0058	0.0023	0.0083	0.0063	0.0049	0.0025
β	0.0068	0.0024	0.0050	0.0025	0.0037	0.0045	0.0116	0.0107	0.0146	0.0093	0.0142	0.0107	0.0032
γ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0015	0.0072	0.0035	0.0062	0.0055
c	0.0094	0.0140	0.0122	0.0121	0.0036	0.0012	0.0029	0.0039	0.0036	0.0068	0.0080	0.0043	0.0117
	0.52	**0.54**	**0.56**	**0.58**	**0.60**	**0.62**	**0.64**	**0.66**	**0.68**	**0.70**	**0.72**	**0.74**	**0.76**
α	0.0027	0.0044	0.0042	0.0016	0.0047	0.0037	0.0078	0.0032	0.0045	0.0007	0.0034	0.0027	0.0047
β	0.0031	0.0025	0.0063	0.0037	0.0051	0.0039	0.0041	0.0031	0.0063	0.0073	0.0050	0.0026	0.0032
γ	0.0067	0.0079	0.0073	0.0169	0.0083	0.0063	0.0086	0.0053	0.0113	0.0104	0.0134	0.0048	0.0050
c	0.0036	0.0045	0.0075	0.0083	0.0146	0.0112	0.0085	0.0050	0.0163	0.0170	0.0113	0.0056	0.0039
	0.78	**0.80**	**0.82**	**0.84**	**0.86**	**0.88**	**0.90**	**0.92**	**0.94**	**0.96**	**0.98**	**1.00**	**Mean**
α	0.0050	0.0031	0.0044	0.0024	0.0037	0.0032	0.0042	0.0051	0.0019	0.0013	0.0017	0.0035	0.0042
β	0.0037	0.0027	0.0023	0.0041	0.0029	0.0063	0.0050	0.0058	0.0038	0.0029	0.0038	0.0028	0.0052
γ	0.0060	0.0061	0.0066	0.0080	0.0099	0.0116	0.0125	0.0057	0.0055	0.0077	0.0059	0.0031	0.0044
c	0.0068	0.0042	0.0039	0.0055	0.0046	0.0063	0.0052	0.0032	0.0078	0.0048	0.0029	0.0015	0.0057

(a) Original internal energy algorithm variant. Whilst varying each of the parameters the remainder took the following values: $\alpha = 0.10$, $\beta = 0.20$, $\gamma = 1.00$ and $c = 0.41$.

	0.00	**0.02**	**0.04**	**0.06**	**0.08**	**0.10**	**0.12**	**0.14**	**0.16**	**0.18**	**0.20**	**0.22**	**0.24**
β	0.0105	0.0043	0.0052	0.0027	0.0033	0.0012	0.0037	0.0057	0.0029	0.0049	0.0074	0.0039	0.0080
γ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
c	0.0008	0.0005	0.0007	0.0011	0.0021	0.0019	0.0026	0.0040	0.0048	0.0044	0.0113	0.0090	0.0165
	0.26	**0.28**	**0.30**	**0.32**	**0.34**	**0.36**	**0.38**	**0.40**	**0.42**	**0.44**	**0.46**	**0.48**	**0.50**
β	0.0038	0.0070	0.0040	0.0096	0.0081	0.0058	0.0022	0.0023	0.0111	0.0075	0.0041	0.0083	0.0088
γ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001
c	0.0076	0.0097	0.0087	0.0056	0.0026	0.0037	0.0035	0.0031	0.0060	0.0065	0.0034	0.0063	0.0108
	0.52	**0.54**	**0.56**	**0.58**	**0.60**	**0.62**	**0.64**	**0.66**	**0.68**	**0.70**	**0.72**	**0.74**	**0.76**
β	0.0049	0.0049	0.0073	0.0093	0.0079	0.0071	0.0083	0.0097	0.0053	0.0110	0.0052	0.0088	0.0102
γ	0.0001	0.0001	0.0001	0.0002	0.0002	0.0048	0.0039	0.0087	0.0080	0.0032	0.0081	0.0063	0.0101
c	0.0057	0.0046	0.0062	0.0101	0.0104	0.0070	0.0066	0.0063	0.0074	0.0079	0.0127	0.0047	0.0036
	0.78	**0.80**	**0.82**	**0.84**	**0.86**	**0.88**	**0.90**	**0.92**	**0.94**	**0.96**	**0.98**	**1.00**	**Mean**
β	0.0051	0.0050	0.0034	0.0084	0.0082	0.0073	0.0039	0.0034	0.0041	0.0047	0.0053	0.0086	0.0062
γ	0.0105	0.0130	0.0103	0.0093	0.0060	0.0127	0.0039	0.0072	0.0034	0.0045	0.0054	0.0028	0.0028
c	0.0060	0.0024	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0047

(b) Perrin internal energy algorithm variant. Whilst varying each of the parameters the remainder took the following values: $\beta = 0.2$, $\gamma = 1.00$ and $c = 0.41$.

Table A.1: The standard deviation of five repetitions of the training set detections as functions of each variant of the algorithm's parameter values. The results were obtained using the potential energy proposed in this thesis. These standard deviations are attributed to Figure 5.3a and Figure 5.3b in Chapter 5.

	−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
Original	0.0085	0.0046	0.0098	0.0066	0.0049	0.0061	0.0043	0.0050	0.0064
Perrin	0.0088	0.0051	0.0102	0.0069	0.0081	0.0096	0.0059	0.0054	0.0064
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
Original	0.0061	0.0043	0.0042	0.0064	0.0063	0.0040	0.0077	0.0077	0.0060
Perrin	0.0044	0.0038	0.0054	0.0045	0.0055	0.0059	0.0055	0.0069	0.0064

Table A.2: The standard deviations of ten repetitions of the line location accuracies of vertical track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. These standard deviations are attributed to Figure 5.4 in Chapter 5.

	−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
1 Hz/s	0.0328	0.0118	0.0127	0.0158	0.0084	0.0094	0.0180	0.0147	0.0113
2 Hz/s	0.0151	0.0041	0.0081	0.0137	0.0091	0.0041	0.0080	0.0099	0.0096
4 Hz/s	0.0061	0.0040	0.0078	0.0034	0.0053	0.0029	0.0060	0.0071	0.0063
8 Hz/s	0.0037	0.0025	0.0029	0.0031	0.0027	0.0033	0.0032	0.0016	0.0022
16 Hz/s	0.0033	0.0020	0.0031	0.0013	0.0032	0.0028	0.0022	0.0043	0.0019
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	0.0054	0.0141	0.0112	0.0100	0.0156	0.0092	0.0087	0.0083	0.0128
2 Hz/s	0.0048	0.0105	0.0055	0.0058	0.0062	0.0041	0.0070	0.0024	0.0075
4 Hz/s	0.0062	0.0069	0.0048	0.0058	0.0024	0.0052	0.0042	0.0037	0.0052
8 Hz/s	0.0040	0.0025	0.0029	0.0028	0.0033	0.0038	0.0033	0.0057	0.0031
16 Hz/s	0.0020	0.0025	0.0016	0.0042	0.0025	0.0020	0.0020	0.0120	0.0031

(a) Original internal energy algorithm variant.

	−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
1 Hz/s	0.0295	0.0058	0.0182	0.0188	0.0119	0.0198	0.0136	0.0111	0.0140
2 Hz/s	0.0079	0.0113	0.0058	0.0079	0.0062	0.0059	0.0172	0.0127	0.0210
4 Hz/s	0.0102	0.0049	0.0034	0.0027	0.0019	0.0064	0.0029	0.0056	0.0048
8 Hz/s	0.0012	0.0003	0.0020	0.0015	0.0021	0.0024	0.0031	0.0041	0.0017
16 Hz/s	0.0018	0.0013	0.0023	0.0002	0.0017	0.0023	0.0012	0.0020	0.0012
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	0.0107	0.0132	0.0137	0.0062	0.0058	0.0110	0.0054	0.0059	0.0126
2 Hz/s	0.0108	0.0119	0.0090	0.0093	0.0090	0.0063	0.0068	0.0072	0.0098
4 Hz/s	0.0074	0.0067	0.0040	0.0114	0.0041	0.0050	0.0040	0.0049	0.0053
8 Hz/s	0.0028	0.0036	0.0027	0.0026	0.0027	0.0065	0.0024	0.0076	0.0029
16 Hz/s	0.0011	0.0011	0.0009	0.0014	0.0022	0.0011	0.0016	0.0020	0.0015

(b) Perrin internal energy algorithm variant.

Table A.3: The standard deviations of the line location accuracies of oblique track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. These standard deviations are attributed to Figure 5.5b and Figure 5.5a in Chapter 5.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	0.0407	0.0254	0.0073	0.0697	0.0217	0.0407	0.0054	0.0034
2%	0.0138	0.0035	0.0048	0.0039	0.0007	0.0055	0.0165	0.0183	0.0350
4%	0.0533	0.0144	0.0131	0.0156	0.0020	0.0004	0.0274	0.0100	0.0410
8%	0.0177	0.0010	0.0184	0.0042	0.0170	0.0053	0.0005	0.0507	0.0214
16%	0.0106	0.0234	0.0235	0.0211	0.0416	0.0138	0.0077	0.0138	0.0100
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0086	0.0033	0.0105	0.0252	0.0070	0.0051	0.0177	0.0105	0.0178
2%	0.0035	0.0175	0.0203	0.0064	0.0144	0.0050	0.0118	0.0091	0.0112
4%	0.0131	0.0158	0.0064	0.0552	0.0003	0.0150	0.0175	0.0449	0.0203
8%	0.0387	0.0093	0.0041	0.0403	0.0149	0.0063	0.0158	–	0.0166
16%	0.0429	0.0369	0.0218	0.0172	0.0085	0.0150	–	–	0.0205

(a) Original internal energy algorithm variant.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	0.0061	0.0039	0.0038	0.0037	0.0034	0.0117	0.0009	0.0040
2%	0.0000	0.0020	0.0027	0.0079	0.0115	0.0041	0.0038	0.0020	0.0106
4%	0.0067	0.0058	0.0026	0.0018	0.0063	0.0072	0.0210	0.0117	0.0020
8%	0.0026	0.0006	0.0027	0.0043	0.0224	0.0067	0.0143	0.0058	0.0046
16%	0.0057	0.0011	0.0120	0.0096	0.0153	0.0081	0.0058	0.0007	0.0112
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0012	0.0067	0.0058	0.0024	0.0020	0.0069	0.0051	0.0026	0.0044
2%	0.0019	0.0021	0.0015	0.0219	0.0067	0.0008	0.0097	0.0065	0.0056
4%	0.0078	0.0162	0.0017	0.0048	0.0070	0.0071	0.0253	0.0060	0.0083
8%	0.0059	0.0160	0.0099	0.0171	0.0154	0.0270	0.1100	–	0.0166
16%	0.0075	0.0153	0.0072	0.0172	0.0163	0.0089	–	–	0.0095

(b) Perrin internal energy algorithm variant.

Table A.4: The standard deviations of the line location accuracies of sinusoidal (ten second period) track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. These standard deviations are attributed to Figure 5.6a and Figure 5.6b in Chapter 5.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0650	0.0054	0.0112	0.0313	0.0059	0.0136	0.0211	0.0032	0.0014
2%	–	0.0209	0.0000	0.0142	0.0247	0.0079	0.0077	0.0098	0.0076
4%	0.0098	0.0093	0.0133	0.0246	0.0097	0.0315	0.0037	0.0033	0.0179
8%	0.0289	0.0077	0.0085	0.0102	0.0024	0.0113	0.0038	0.0073	0.0061
16%	0.0032	0.0071	0.0148	0.0118	0.0036	0.0073	0.0171	0.0066	0.0068
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0204	0.0148	0.0094	0.0127	0.0302	0.0041	0.0136	0.0141	0.0163
2%	0.0098	0.0104	0.0179	0.0006	0.0032	0.0038	0.0132	0.0065	0.0099
4%	0.0374	0.0061	0.0187	0.0107	0.0197	0.0116	0.0192	0.0047	0.0148
8%	0.0134	0.0063	0.0145	0.0291	0.0124	0.0208	0.0116	0.0392	0.0137
16%	0.0013	0.0015	0.0037	0.0247	0.0167	0.0177	0.0241	0.0039	0.0101

(a) Original internal energy algorithm variant.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0016	0.0166	0.0078	0.0027	0.0014	0.0012	0.0072	0.0122	0.0018
2%	–	0.0063	0.0135	0.0044	0.0051	0.0081	0.0037	0.0030	0.0111
4%	0.0058	0.0047	0.0062	0.0070	0.0093	0.0033	0.0056	0.0010	0.0058
8%	0.0007	0.0034	0.0032	0.0012	0.0013	0.0066	0.0015	0.0014	0.0024
16%	0.0128	0.0072	0.0062	0.0012	0.0015	0.0022	0.0014	0.0056	0.0008
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0033	0.0037	0.0000	0.0077	0.0044	0.0065	0.0032	0.0047	0.0051
2%	0.0042	0.0017	0.0003	0.0021	0.0093	0.0006	0.0140	0.0006	0.0055
4%	0.0079	0.0057	0.0059	0.0017	0.0046	0.0028	0.0025	0.0042	0.0049
8%	0.0041	0.0042	0.0021	0.0091	0.0028	0.0024	0.0191	0.0193	0.0050
16%	0.0023	0.0028	0.0057	0.0047	0.0070	0.0057	0.0093	0.0000	0.0045

(b) Perrin internal energy algorithm variant.

Table A.5: The standard deviations of the line location accuracies of sinusoidal (fifteen second period) track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. These standard deviations are attributed to Figure 5.7a and Figure 5.7b in Chapter 5.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0058	0.0000	0.0011	0.0009	0.0110	0.0056	0.0030	0.0035	0.0053
2%	–	0.0008	0.0014	0.0074	0.0015	0.0025	0.0073	0.0005	0.0138
4%	0.0000	0.0127	0.0067	0.0034	0.0072	0.0064	0.0065	0.0078	0.0012
8%	0.0000	0.0121	0.0018	0.0072	0.0025	0.0028	0.0066	0.0030	0.0035
16%	0.0163	0.0013	0.0038	0.0087	0.0103	0.0122	0.0014	0.0051	0.0031
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0021	0.0188	0.0028	0.0062	0.0033	0.0008	0.0003	0.0076	0.0046
2%	0.0020	0.0043	0.0014	0.0036	0.0152	0.0002	0.0041	0.0011	0.0042
4%	0.0065	0.0044	0.0036	0.0105	0.0066	0.0110	0.0033	0.0057	0.0061
8%	0.0015	0.0050	0.0000	0.0053	0.0034	0.0087	0.0172	0.0280	0.0064
16%	0.0012	0.0089	0.0034	0.0098	0.0064	0.0024	0.0009	0.0015	0.0057

(a) Original internal energy algorithm variant.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0373	0.0178	0.0116	0.0119	0.0107	0.0185	0.0133	0.0168	0.0200
2%	–	0.0094	0.0115	0.0080	0.0129	0.0139	0.0174	0.0144	0.0121
4%	0.0000	0.0064	0.0020	0.0035	0.0035	0.0061	0.0088	0.0158	0.0112
8%	0.0001	0.0064	0.0066	0.0087	0.0077	0.0074	0.0129	0.0127	0.0167
16%	0.0129	0.0024	0.0067	0.0073	0.0117	0.0088	0.0085	0.0055	0.0090
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0072	0.0195	0.0102	0.0135	0.0087	0.0087	0.0047	0.0189	0.0147
2%	0.0119	0.0116	0.0113	0.0143	0.0112	0.0179	0.0084	0.0093	0.0115
4%	0.0164	0.0124	0.0114	0.0112	0.0083	0.0048	0.0097	0.0066	0.0081
8%	0.0074	0.0104	0.0125	0.0129	0.0085	0.0141	0.0129	0.0086	0.0098
16%	0.0181	0.0109	0.0153	0.0134	0.0128	0.0093	0.0119	0.0128	0.0104

(b) Perrin internal energy algorithm variant.

Table A.6: The standard deviations of the line location accuracies of sinusoidal (twenty second period) track detections as functions of the spectrogram's SNR—a comparison between the original and Perrin internal energies. These standard deviations are attributed to Figure 5.8a and Figure 5.8b in Chapter 5.

0.00	**0.02**	**0.04**	**0.06**	**0.08**	**0.10**	**0.12**	**0.14**	**0.16**	**0.18**	**0.20**	**0.22**	**0.24**
0.0075	0.0101	0.0109	0.0093	0.0126	0.0159	0.0103	0.0101	0.0074	0.0152	0.0185	0.0109	0.0203
0.26	**0.28**	**0.30**	**0.32**	**0.34**	**0.36**	**0.38**	**0.40**	**0.42**	**0.44**	**0.46**	**0.48**	**0.50**
0.0133	0.0239	0.0160	0.0203	0.0069	0.0322	0.0116	0.0199	0.0111	0.0198	0.0073	0.0367	0.0225
0.52	**0.54**	**0.56**	**0.58**	**0.60**	**0.62**	**0.64**	**0.66**	**0.68**	**0.70**	**0.72**	**0.74**	**0.76**
0.0374	0.0345	0.0311	0.0299	0.0252	0.0240	0.0368	0.0274	0.0256	0.0222	0.0280	0.0350	0.0199
0.78	**0.80**	**0.82**	**0.84**	**0.86**	**0.88**	**0.90**	**0.92**	**0.94**	**0.96**	**0.98**	**1.00**	**Mean**
0.0277	0.0294	0.0284	0.0169	0.0345	0.0137	0.0143	0.0220	0.0315	0.0216	0.0132	0.0084	0.0204

Table A.7: The standard deviation of five repetitions of the training set detections as a function of the gradient potential's parameter values. The results were obtained using the potential energy proposed in this thesis and using a single active contour. These standard deviations are attributed to Figure 5.13 in Chapter 5.

−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
0.0049	0.0028	0.0025	0.0042	0.0013	0.0014	0.0008	0.0021	0.0003
3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
0.0005	0.0007	0.0005	0.0002	0.0002	0.0002	0.0001	0.0003	0.0014

Table A.8: The standard deviations of the line location accuracies of vertical track detections as functions of the spectrogram's SNR—single track detection. These standard deviations are attributed to Figure 5.14 in Chapter 5.

	−1.0 dB	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**	**2.5 dB**	**3.0 dB**
1 Hz/s	0.0015	0.0022	0.0037	0.0095	0.0091	0.0135	0.0143	0.0223	0.0218
2 Hz/s	0.0008	0.0010	0.0001	0.0012	0.0010	0.0019	0.0045	0.0053	0.0008
4 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0004	0.0000	0.0027	0.0022
8 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0002
16 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	3.5 dB	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**6.5 dB**	**7.0 dB**	**Mean**
1 Hz/s	0.0195	0.0168	0.0081	0.0066	0.0082	0.0023	0.0060	0.0043	0.0100
2 Hz/s	0.0069	0.0166	0.0124	0.0093	0.0120	0.0118	0.0112	0.0081	0.0062
4 Hz/s	0.0020	0.0006	0.0012	0.0005	0.0037	0.0027	0.0036	0.0042	0.0014
8 Hz/s	0.0005	0.0000	0.0017	0.0000	0.0005	0.0002	0.0005	0.0000	0.0002
16 Hz/s	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table A.9: The standard deviations of the line location accuracies of oblique track detections as functions of the spectrogram's SNR—single track detection. These standard deviations are attributed to Figure 5.15 in Chapter 5.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	–	0.0008	0.0011	0.0011	0.0123	0.0080	0.0073	0.0106	0.0120
2%	0.0000	0.0000	0.0003	0.0004	0.0009	0.0020	0.0022	0.0006	0.0041
4%	0.0001	0.0000	0.0000	0.0000	0.0016	0.0004	0.0006	0.0014	0.0053
8%	0.0000	0.0000	0.0010	0.0005	0.0016	0.0006	0.0011	0.0048	0.0011
16%	0.0000	0.0000	0.0003	0.0025	0.0003	0.0006	0.0003	0.0012	0.0010
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0040	0.0066	0.0186	0.0109	0.0108	0.0067	0.0046	0.0020	0.0073
2%	0.0070	0.0058	0.0132	0.0088	0.0228	0.0145	0.0046	0.0043	0.0054
4%	0.0047	0.0061	0.0054	0.0075	0.0141	0.0071	0.0108	0.0059	0.0042
8%	0.0045	0.0047	0.0050	0.0136	0.0037	0.0139	0.0352	–	0.0057
16%	0.0036	0.0042	0.0052	0.0058	0.0071	0.0110	–	–	0.0029

Table A.10: The standard deviations of the line location accuracies of sinusoidal (ten second period) track detections as functions of the spectrogram's SNR—single track detection. These standard deviations are attributed to Figure 5.16 in Chapter 5.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0000	0.0019	0.0014	0.0029	0.0043	0.0079	0.0153	0.0200	0.0076
2%	–	0.0006	0.0011	0.0009	0.0012	0.0016	0.0024	0.0050	0.0044
4%	0.0000	0.0000	0.0009	0.0000	0.0002	0.0003	0.0008	0.0005	0.0017
8%	0.0000	0.0000	0.0000	0.0004	0.0001	0.0000	0.0005	0.0000	0.0003
16%	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001	0.0000	0.0001
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0113	0.0142	0.0116	0.0037	0.0101	0.0017	0.0022	0.0007	0.0069
2%	0.0094	0.0072	0.0090	0.0102	0.0146	0.0075	0.0101	0.0111	0.0060
4%	0.0030	0.0016	0.0108	0.0041	0.0082	0.0127	0.0115	0.0098	0.0039
8%	0.0002	0.0016	0.0026	0.0101	0.0041	0.0101	0.0100	0.0124	0.0031
16%	0.0013	0.0010	0.0014	0.0044	0.0060	0.0030	0.0085	0.0103	0.0021

Table A.11: The standard deviations of the line location accuracies of sinusoidal (fifteen second period) track detections as functions of the spectrogram's SNR—single track detection. These standard deviations are attributed to Figure 5.17 in Chapter 5.

	−2.0 dB	**−1.5 dB**	**−1.0 dB**	**−0.5 dB**	**0.0 dB**	**0.5 dB**	**1.0 dB**	**1.5 dB**	**2.0 dB**
1%	0.0000	0.0000	0.0016	0.0007	0.0059	0.0055	0.0122	0.0174	0.0171
2%	–	0.0000	0.0000	0.0006	0.0034	0.0012	0.0038	0.0016	0.0177
4%	0.0000	0.0000	0.0001	0.0000	0.0008	0.0019	0.0005	0.0018	0.0029
8%	0.0000	0.0002	0.0000	0.0000	0.0002	0.0000	0.0002	0.0020	0.0020
16%	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0001	0.0004	0.0014
	2.5 dB	**3.0 dB**	**3.5 dB**	**4.0 dB**	**4.5 dB**	**5.0 dB**	**5.5 dB**	**6.0 dB**	**Mean**
1%	0.0164	0.0058	0.0058	0.0050	0.0050	0.0026	0.0041	0.0015	0.0063
2%	0.0048	0.0108	0.0084	0.0100	0.0169	0.0150	0.0095	0.0150	0.0074
4%	0.0046	0.0034	0.0080	0.0173	0.0138	0.0127	0.0092	0.0071	0.0049
8%	0.0015	0.0034	0.0028	0.0114	0.0070	0.0121	0.0088	0.0142	0.0039
16%	0.0022	0.0023	0.0040	0.0108	0.0078	0.0068	0.0052	0.0062	0.0028

Table A.12: The standard deviations of the line location accuracies of sinusoidal (twenty second period) track detections as functions of the spectrogram's SNR—single track detection. These standard deviations are attributed to Figure 5.18 in Chapter 5.

List of References

[1] Abe, T., Matsuzawa, Y., June 2000. A region extraction method using multiple active contour models. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. Vol. 1. pp. 64–69.

[2] Abel, J. S., 2008. Personal Communication.

[3] Abel, J. S., Lee, H. J., Lowell, A. P., March 1992. An image processing approach to frequency tracking. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. Vol. 2. pp. 561–564.

[4] Adams, G. J., Evans, R. J., April 1994. Neural networks for frequency line tracking. IEEE Transactions on Signal Processing 42 (4), 936–941.

[5] Aho, A. V., Hopcroft, J. E., Ullman, J. D., 1976. The Design and Analysis of Computer Algorithms. Addison-Wesley.

[6] Allen, J., 1982. Application of the short-time Fourier transform to speech processing and spectral analysis. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 1012–1015.

[7] Allen, J. B., June 1977. Short term spectral analysis, synthesis, and modification by discrete Fourier transform. IEEE Transactions on Acoustics, Speech, Signal Processing ASSP-25, 235–238.

[8] Altes, R. A., April 1980. Detection, estimation, and classification with spectrograms. Journal of the Acoustical Society of America 67 (4), 1232–1246.

[9] Amini, A. A., Tehrani, S., Weymouth, T. E., December 1988. Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints. In: Proceedings of the Second IEEE International Conference on Computer Vision. pp. 95–99.

[10] Amini, A. A., Weymouth, T. E., Jain, R. C., 1990. Using dynamic programming for solving variational problems in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (9), 855–867.

[11] Anderson, B. D. O., James, B., Williamson, R. C., February 1992. Frequency line tracking, extended Kalman filters and some HMM problems. In: Proceedings of the Workshop on Hidden Markov Models for Tracking. pp. 1–8.

[12] Aridgides, T., Fernandez, M., Dobeck, G., August 1999. Adaptive clutter suppression, sea mine detection/classification, and fusion processing string for sonar imagery. In: Detection and Remediation Technologies for Mines and Minelike Targets IV. Vol. 3710. pp. 626–637.

[13] Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing 50 (2), 174–188.

[14] Aughenbaugh, J., La Cour, B., July 2009. Use of prior information in active sonar tracking. In: Proceedings of the International Conference on Information Fusion. pp. 1584–1591.

[15] Aughenbaugh, J., Yocom, B., La Cour, B., Yudichak, T., October 2009. Active clutter reduction through fusion with passive sonar data. Journal of the Acoustical Society of America 126 (4), 2184–2184.

[16] Aujol, J. F., Aubert, G., Blanc-Feraud, L., 2003. Wavelet-based level set evolution for classification of textured images. IEEE Transactions on Image Processing 12 (12), 1634–1641.

[17] Ballerini, L., May 1999. Genetic snakes for medical images segmentation. In: Proceedings of the First European Workshop on Evolutionary Image Analysis, Signal Processing and Telecommunications. Vol. 1596 of Lecture Notes in Computer Science. Springer-Verlag, London, UK, pp. 59–73.

[18] Banerji, A., Goutsias, J., May 1996. Detection of minelike targets in heavily cluttered environments using the MNF transform and grayscale morphological image reconstruction. In: Detection and Remediation Technologies for Mines and Minelike Targets. Vol. 2765. pp. 58–68.

[19] Barrett, R. F., Holdsworth, D. A., 1993. Frequency tracking using hidden Markov models with amplitude and phase information. IEEE Transactions on Signal Processing 41 (10), 2965–2976.

[20] Barrett, R. F., McMahon, D. R. A., August 1987. ML estimation of the fundamental frequency of a harmonic series. In: Proceedings of International Conference on Information Sciences, Signal Processing and their Applications. Brisbane, Australia, pp. 333–336.

[21] Barron, A., Rissanen, J., Yu, B., October 1998. The minimum description length principle in coding and modeling. IEEE Transactions on Information Theory 44 (6), 2743–2760.

[22] Baum, L. E., Petrie, T., Soules, G., Weiss, N., 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. The Annals of Mathematical Statistics 41 (1), 164–171.

[23] Belhumeur, P. N., Hespanha, J. P., Kriegman, D. J., August 1997. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7), 711–720.

[24] Belkin, M., Niyogi, P., 2003. Laplacian eigenmaps and spectral techniques for embedding and clustering. Neural Computations 15 (6), 1373–1396.

[25] Bellman, R., 1957. Dynamic Programming. Princeton University Press, Princeton.

[26] Bengio, Y., Paiement, J.-F., Vincent, P., Delalleau, O., Le Roux, N., Ouimet, M., December 2004. Out-of-sample extensions for LLE, ISOMAP, MDS, eigenmaps and spectral clustering. In: Advances in Neural Information Processing Systems. Vol. 16. MIT Press, pp. 177–184.

[27] Berger, M., 1991. Towards dynamic adaptation of snake contours. In: Proceedings of the Sixth International conference on image analysis and processing. pp. 47–54.

[28] Bishop, C. M., 1995. Neural Networks for Pattern Recognition. Oxford University Press Inc.

[29] Blake, A., Isard, M., 1998. Active Contours. Springer.

[30] Brahosky, V. A., June 1992. A combinatorial approach to automated lofargram analysis. Master's thesis, Naval Postgraduate School, Monterey, U.S.A.

[31] Brent, R. P., 1976. Fast multiple-precision evaluation of elementary functions. Journal of the ACM 23, 242–251.

[32] Brillinger, D. R., 1975. Time Series: data analysis and theory. International Series in Decision Processes. Holt, Reinhart and Winston Inc., New York.

[33] Canny, J. F., November 1986. A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (6), 679–698.

[34] Caselles, V., Kimmel, R., Sapiro, G., February 1997. Geodesic active contours. International Journal of Computer Vision 22 (1), 61–79.

[35] Chalana, V., Linker, D. T., Haynor, D. R., Kim, Y., June 1996. A multiple active contour model for cardiac boundary detection on echocardiographic sequences. IEEE Transactions on Medical Imaging 15 (3), 290–298.

[36] Chan, T. F., Sandberg, B. Y., Vese, L. A., June 2000. Active contours without edges for vector-valued images. Journal of Visual Communication and Image Representation 11 (2), 130–141.

[37] Chan, T. F., Vese, L. A., February 2001. Active contours without edges. IEEE Transactions on Image Processing 10 (2), 266–277.

[38] Chen, C.-H., Lee, J.-D., Lin, M.-C., 2000. Classification of underwater signals using neural networks. Tamkang Journal of Science and Engineering 3 (1), 31–48.

[39] Chen, C.-T., Millero, F. J., November 1977. Speed of sound in seawater at high pressures. Journal of the Acoustical Society of America 62 (5), 1129–1135.

[40] Chen, T.-S., December 1990. Simulated annealing in sonar track detection. Master's thesis, Naval Postgraduate School, Monterey, U.S.A.

[41] Chesnaud, C., Réfreégier, P., Boulet, V., November 1999. Statistical region snake-based segmentation adapted to different physical noise models. IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (11), 1145–1157.

[42] Cohen, L., Cohen, I., November 1993. Finite-element methods for active contour models and balloons for 2-D and 3-D images. IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (11), 1131–1147.

[43] Cohen, L. D., March 1991. On active contour models and balloons. Computer Vision, Graphics, and Image Processing: Image Understanding 53 (2), 211–218.

[44] Cohen, L. D., Kimmel, R., August 1997. Global minimum for active contour models: a minimal path approach. International Journal of Computer Vision 24 (1), 57–78.

[45] Cootes, T. F., Cooper, D., Taylor, C. J., Graham, J., January 1995. Active shape models — their training and application. Computer Vision and Image Understanding 61 (1), 38–59.

[46] Coppersmith, D., Winograd, S., March 1990. Matrix multiplication via arithmetic progressions. Journal of Symbolic Computation 9 (3), 251–280.

[47] Córdoba, A., April 1989. Dirac combs. Letters in Mathematical Physics 17 (3), 191–196.

[48] Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., September 2001. Introduction to Algorithms, 2nd Edition. MIT Press.

[49] da Costa Andrade, E. N., January 1959. Doppler and the Doppler effect. Endeavour XVIII (69).

[50] Davatzikos, C., Prince, J. L., January 1999. Convexity analysis of active contour problems. Image and Vision Computing 17 (1), 27–36.

[51] Davatzikos, C. A., Prince, J. L., March 1995. An active contour model for mapping the cortex. IEEE Transactions on Medical Imaging 14 (1), 65–80.

[52] Davis, D. N., Natarajan, K., Claridge, E., July 1995. Multiple energy function active contours applied to CT and MR images. In: Proceedings of the Fifth IEEE International Conference on Image Processing and its Applications. pp. 114–118.

[53] Del Grosso, V. A., October 1974. New equation for the speed of sound in natural waters (with comparisons to other equations). Journal of the Acoustical Society of America 56 (4), 1084–1091.

[54] Di Martino, J.-C., Colnet, B., Di Martino, M., April 1994. The use of non supervised neural networks to detect lines in lofargram. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. Vol. 2. IEEE, pp. 293–296.

[55] Di Martino, J.-C., Haton, J. P., Laporte, A., April 1993. Lofargram line tracking by multistage decision process. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. Vol. 1. IEEE, pp. 317–320.

[56] Di Martino, J.-C., Tabbone, S., September 1995. Detection of lofar lines. In: Braccini, C., De Floriani, L., Vernazza, G. (Eds.), Proceedings of the 8th International Conference on Image Analysis and Processing. Vol. 974 of Lecture Notes in Computer Science. Springer, Berlin, pp. 709–714.

[57] Di Martino, J.-C., Tabbone, S., January 1996. An approach to detect lofar lines. Pattern Recognition Letters 17 (1), 37–46.

[58] Doucet, A., De Freitas, N., Gordon, N. J., 2001. Sequential Monte Carlo Methods in Practice. Information Science and Statistics. Springer-Verlag.

[59] Duda, R. O., Hart, P. E., January 1972. Use of Hough transform to detect lines and curves in pictures. Communications of the ACM 15 (1), 11–15.

[60] Duda, R. O., Hart, P. E., Stork, D. G., 2000. Pattern Classification. Wiley-Interscience Publication.

[61] Dugnola, B., Fernándeza, C., Galiano, G., March 2007. Wolf population counting by spectrogram image processing. Applied Mathematics and Computation 186 (1), 820–830.

[62] Dushaw, B. D., Worcester, P. F., Cornuelle, B. D., Howe, B. M., January 1993. On equations for the speed of sound in seawater. Journal of the Acoustical Society of America 93 (1), 255–275.

[63] Egan, J. P., 1975. Signal detection theory and ROC analysis. Series in Cognition and Perception. Academic Press, New York.

[64] Ellis, D., March 2007. Measurements and analysis of reverberation, target echo and clutter. Tech. Rep. N00014-06-1-0830 and N00014-03-1-0420, Defence Research and Development Canada — Atlantic.

[65] Fawcett, T., June 2006. An introduction to ROC analysis. Pattern Recognition Letters 27 (8), 861–874.

[66] Ferguson, B. G., October 1996. Time-frequency signal analysis of hydrophone data. IEEE Journal of Oceanic Engineering 21 (4), 537–544.

[67] Fukunaga, K., 1990. Introduction to Statistical Pattern Recognition. Elsevier.

[68] Ghosh, J., Turner, K., Beck, S., Deuser, L., June 1996. Integration of neural classifiers for passive sonar signals. Control and Dynamic Systems — Advances in Theory and Applications 77, 301–338.

[69] Gillespie, D., 2004. Detection and classification of right whale calls using an 'edge' detector operating on a smoothed spectrogram. Canadian Acoustics 32 (2), 39–47.

[70] Goldman, A., Cohen, I., July 2004. Anomaly detection based on an iterative local statistics approach. Signal Processing 84 (7), 1225–1229.

[71] Gonzalez, R. C., Woods, R. E., 2006. Digital Image Processing, 3rd Edition. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[72] Grigorakis, A., August 1997. Application of detection theory to the measurement of the minimum detectable signal for a sinusoid in Gaussian noise displayed on a lofargram. Tech. Rep. DSTO-TR-0568, Maritime Operations Division, Aeronautical and Maritime Research Laboratory, Defence Science and Technology Organisation, Canberra.

[73] Grzeszczuk, R. P., Levin, D. N., October 1997. Brownian strings: segmenting images with stochastically deformable contours. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (10), 1100–1114.

[74] Gunes, T., Erdöl, N., May 2006. HMM based spectral frequency line tracking: improvements and new results. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. Vol. 2. pp. 673–676.

[75] Gunn, S. R., Nixon, M. S., January 1997. A robust snake implementation; a dual active contour. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (1), 63–68.

[76] Harris, F. J., January 1978. On the use of windows for harmonic analysis with the discrete Fourier transform. Proceedings of the IEEE 66 (1), 51–83.

[77] Haykin, S., 1999. Neural Networks : A Comprehensive Foundation, 2nd Edition. Prentice Hall, Upper Saddle River, N.J.

[78] Haykin, S., Thomson, D., November 1998. Signal detection in a nonstationary environment reformulated as an adaptive pattern classification problem. Proceedings of the IEEE Special Issue on Intelligent Signal Processing 86 (11), 2325–2344.

[79] Hebb, D. O., 1949. The Organization of behaviour. John Wiley, New York.

[80] Herault, L., Horaud, R., September 1993. Figure ground discrimination: a combinatorial optimisation approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (9), 899–914.

[81] Hernández, C., Schmitt, F., October 2003. A snake approach for high quality image-based 3D object modeling. In: Proceedings of the Second IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision. pp. 241–248.

[82] Hinton, G., Roweis, S. T., December 2003. Stochastic neighbor embedding. In: Advances in Neural Information Processing Systems. Vol. 15. MIT Press, pp. 857–864.

[83] Hinton, G., Salakhutdinov, R. R., July 2006. Reducing the dimensionality of data with neural networks. Science 313 (5786), 504–507.

[84] Hinton, G. E., 1987. Learning translation invariant recognition in massively parallel networks. In: Proceedings of the PARLE Conference on Parallel Architectures and Languages Europe. Vol. 258 of Lecture Notes in Computer Science. pp. 1–13.

[85] Hopfield, J. J., April 1982. Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences of the USA 79 (8), 2554–2558.

[86] Howell, B. P., Wood, S., Koksal, S., September 2003. Passive sonar recognition and analysis using hybrid neural networks. In: Proceedings of OCEANS '03. Vol. 4. pp. 1917–1924.

[87] Hubbert, J., Dixon, M., Ellis, S., Meymaris, G., 2009. Weather radar ground clutter, part i: Identification, modeling and simulation. Journal of Atmospheric and Oceanic Technology, In Press.

[88] Jauffret, C., Bouchet, D., November 1996. Frequency line tracking on a lofargram: an efficient wedding between probabilistic data association modelling and dynamic programming technique. In: Conference Record of the Thirtieth Asilomar Conference on Signals, Systems and Computers. Vol. 1. IEEE, pp. 486–490.

[89] Ji, L., Yan, H., April 2002. Attractable snakes based on the greedy algorithm for contour extraction. Pattern Recognition 35 (4), 791–806.

[90] Jia, P., Yin, J., Huang, X., Hu, D., December 2009. Incremental laplacian eigenmaps by preserving adjacent information between data points. Pattern Recognition Letters 30 (16), 1457–1463.

[91] Jifeng, N., Chengke, W., Shigang, L., Shuqin, Y., January 2007. NGVF: an improved external force field for active contour model. Pattern Recognition Letters 28 (1), 58–63.

[92] Jolliffe, I., 2002. Principal Component Analysis, 2nd Edition. Springer.

[93] Kalman, R., March 1960. A new approach to linear filtering and prediction problems. Transactions of the ASME — Journal of Basic Engineering 82 (Series D), 35–45.

[94] Kanevsky, D., Sainath, T. N., Ramabhadran, B., Nahamoo, D., September 22–26 2008. Generalization of extended Baum-Welch parameter estimation for discriminative training and decoding. In: Proceedings of the 9th Annual Conference of the International Speech Communication Association. pp. 277–280.

[95] Karhunen, J., Joutsensalo, J., 1995. Generalizations of principal component analysis, optimization problems, and neural networks. Neural Networks 8 (4), 549–562.

[96] Kass, M., Witkin, A., Terzopoulos, D., January 1988. Snakes: Active contour models. International Journal of Computer Vision 1 (4), 321–331.

[97] Kendall, G. D., Hall, T. J., May 1993. Improving generalisation with Ockham's networks: minimum description length networks. In: Proceedings of the 3rd International Conference on Artificial Neural Networks. pp. 81–85.

[98] Kendall, G. D., Hall, T. J., Newton, T. J., June 1993. An investigation of the generalisation performance of neural networks applied to lofargram classification. Neural Computing and Applications 1 (2), 147–159.

[99] Khotanzad, A., Lu, J. H., Srinath, M. D., June 1989. Target detection using a neural network based passive sonar system. In: Proceedings of the International Joint Conference on Neural Networks. Vol. 1. pp. 335–440.

[100] Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., May 1983. Optimization by simulated annealing. Science 220 (4598), 671–680.

[101] Koenig, W., Dunn, H. K., Lacy, L. Y., July 1946. The sound spectrograph. Journal of the Acoustical Society America 18 (1), 244–244.

[102] Koffka, K., 1935. Principles of gestalt psychology. Harcourt Brace, New York.

[103] Kohonen, T., January 1982. Self-organized formation of topologically correct feature maps. Biological Cybernetics 43 (1), 59–69.

[104] Kohonen, T., 2001. Self-Organizing Maps, 3rd Edition. Vol. 30 of Springer Series in Information Sciences. Springer, Heidelberg.

[105] Kootsookos, P. J., 1993. A review of the frequency estimation and tracking problems. Tech. rep., Systems Engineering Department, Australian National University.

[106] Kouropteva, O., Okun, O., Pietikäinen, M., October 2005. Incremental locally linear embedding. Pattern Recognition 38 (10), 1764–1767.

[107] Lam, K.-M., Yan, H., January 1994. Fast greedy algorithm for active contours. Electronics Letters 30 (1), 21–23.

[108] Larson, R. E., 1968. State Increment Dynamic Programming. Elsevier, New York.

[109] Law, M. H. C., Jain, A. K., March 2006. Incremental nonlinear dimensionality reduction by manifold learning. IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (3), 377–391.

[110] Le Chevalier, F., 2002. Principles of radar and sonar signal processing. Artech House Publishers.

[111] Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., Jackel, L. D., 1990. Backpropagation applied to handwritten zip code recognition. Neural Computation 1 (4), 541–551.

[112] Lee, C.-H., April 1992. Simulated annealing applied to acoustic signal tracking. In: Dougherty, E. R., Astola, J. T., Boncelet, C. G. (Eds.), Proceedings of the SPIE, Nonlinear Image Processing III. Vol. 1658 of Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference. pp. 344–355.

[113] Lee, J. A., Verleysen, M., August 2005. Nonlinear dimensionality reduction of data manifolds with essential loops. Neurocomputing 67, 29–53.

[114] Leeming, N., March 1993. Artificial neural nets to detect lines in noise. In: Proceedings of the International Conference on Acoustic Sensing and Imaging. pp. 147–152.

[115] Li, C., Xu, W., Tayjasanant, T., July 2003. Interharmonics: basic concepts and techniques for their detection and measurement. Electric Power Systems Research 66 (1), 39–48.

[116] Liu, X., Yin, J., Feng, Z., Dong, J., 2006. Incremental manifold learning via tangent space alignment. In: Proceedings of the Second IAPR International Workshop on Artificial Neural Networks in Pattern Recognition. Vol. 4087 of Lecture Notes in Computer Science. pp. 107–121.

[117] Lu, M., Li, M., Mao, W., August 2007. The detection and tracking of weak frequency line based on double-detection algorithm. In: Proceedings of the IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications. pp. 1195–1198.

[118] MacInnes, C., April 2004. Source localization using subspace estimation and spatial filtering. IEEE Journal of Oceanic Engineering 29 (2), 488–497.

[119] Mackenzie, K. V., September 1981. Nine-term equation for sound speed in the oceans. Journal of the Acoustical Society of America 70 (3), 807–812.

[120] Matthews, M. L., Bos, J., Webb, R. D. G., October 2003. A prototype task network model to simulate the analysis of narrow band sonar data and the effects of automation on critical operator tasks. Tech. Rep. DRDC-TORONTO-CR-2003-131, Defence Research and Development Canada, Toronto.

[121] McCulloch, W. S., Pitts, W. H., December 1943. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 5 (4), 115–133.

[122] McMahon, D. R. A., Barrett, R. F., September 1986. An efficient method for the estimation of the frequency of a single tone in noise from the phases of discrete Fourier transform. Signal Processing 11 (2), 169–177.

[123] McMahon, D. R. A., Barrett, R. F., 1987. Generalization of the method for the estimation of the frequencies of tones in noise from the phases of discrete Fourier transforms. Signal Processing 12 (4), 371–383.

[124] Mellema, G., June 2007. An structured approach to passive sonar track segment association. In: Proceedings of OCEANS 2007. pp. 1–6.

[125] Mellinger, D. K., Nieukirk, S. L., Matsumoto, H., Heimlich, S. L., Dziak, R. P., Haxel, J., Fowler, M., Meinig, C., Miller, H. V., October 2007. Seasonal occurrence of North Atlantic Right Whale (*Eubalaena glacialis*) vocalizations at two sites on the Scotian Shelf. Marine Mammal Science 23 (4), 856–867.

[126] Mero, L., Vassey, Z., September 1975. A simplified and fast version of the Heuckel operator for finding optimal edges in pictures. In: Proceedings of the International conference on Artificial Intelligence. pp. 650–655.

[127] Mirhosseini, A. R., Yan, H., June 1997. Optimally fast greedy algorithm for active contours. In: Proceedings of the IEEE International Symposium on Circuits and Systems. pp. 1189–1192.

[128] Mitchell, M., 1996. An Introduction to Genetic Algorithms. MIT Press, Cambridge, U.S.A.

[129] Mitchell, T. M., October 1997. Machine Learning. McGraw-Hill, New York.

[130] Morrissey, R. P., Ward, J., DiMarzio, N., Jarvis, S., Moretti, D. J., November–December 2006. Passive acoustic detection and localisation of sperm whales (*Physeter Macrocephalus*) in the tongue of the ocean. Applied Acoustics 67 (11–12), 1091–1105.

[131] Nayar, S., Baker, S., Murase, H., March 1998. Parametric feature detection. International Journal of Computer Vision 27 (1), 471–477.

[132] Nelson, S., Tuovila, S., May 1996. Automated recognition of acoustic-image clutter. In: Detection and Remediation Technologies for Mines and Minelike Targets. Vol. 2765. pp. 122–129.

[133] Neuenschwander, W. M., Fua, P., Székely, G., Kübler, O., October 1994. Making snakes converge from minimal initialisation. In: Proceedings of IEEE International Conference on Pattern Recognition. pp. 613–615.

[134] Nowlan, S. J., Hinton, G. E., July 1992. Simplifying neural networks by soft weight-sharing. Neural Computation 4 (4), 473–493.

[135] Paragios, N., Deriche, R., June 1999. Geodesic active contours for supervised texture segmentation. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. pp. 2422–2427.

[136] Pardàs, M., Sayrol, E., November 2001. Motion estimation based tracking of active contours. Pattern Recognition Letters 22 (13), 1447–1456.

[137] Paris, S., Jauffret, C., March 2001. A new tracker for multiple frequency line. In: Proceedings of the IEEE Conference on Aerospace. Vol. 4. IEEE, pp. 1771–1782.

[138] Paris, S., Jauffret, C., April 2003. Frequency line tracking using HMM-based schemes. IEEE Transactions on Aerospace and Electronic Systems 39 (2), 439–450.

[139] Payne, C. M., January 2007. Principles of Naval Weapon Systems. Naval Institute Press.

[140] Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. Philosophical Magazine 2 (6), 559–572.

[141] Perrin, D. P., Smith, C. E., 2001. Rethinking classical internal forces for active contour models. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. pp. 615–620.

[142] Perrone, A., 1969. Deep-ocean ambient-noise spectra in the northwest Atlantic. Journal of the Acoustical Society of America 46, 762–770.

[143] Poole, D., 2005. Linear Algebra: A Modern Introduction, 2nd Edition. Brooks/Cole.

[144] Potter, J. R., Mellinger, D. K., Clark, C. W., September 1994. Marine mammal call discrimination using artificial neural networks. Journal of the Acoustical Society of America 96 (3), 1255–1262.

[145] Pratt, W. K., 1991. Digital Image Processing, 2nd Edition. Wiley-Interscience Publication.

[146] Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P., October 1992. Numerical recipes in C: the art of scientific computing, 2nd Edition. Cambridge University Press, New York, NY, USA.

[147] Prewitt, J. M. S., 1970. Picture Processing and Psychopictorics. Academic Press Inc., New York, NY, USA, Ch. Object Enhancement and Extraction, pp. 75–149.

[148] Quinn, B. G., May 1994. Estimating frequency by interpolation using Fourier coefficients. IEEE Transactions on Signal Processing 42 (5), 1264–1268.

[149] Quinn, B. G., Hannan, E. J., 2001. The Estimation and Tracking of Frequency. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

[150] Rabiner, L. R., February 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE 77 (2), 257–286.

[151] Rabiner, L. R., Levinson, S. E., Sondhi, M. M., April 1983. An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition. Bell System Technical Journal 62 (4), 1035–1074.

[152] Rife, D. C., Boorstyn, R. R., September 1974. Single-tone parameter estimation from discrete-time observations. IEEE Transactions on Information Theory 20 (5), 591–598.

[153] Rosenblatt, F., 1962. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington, D. C.

[154] Roweis, S. R., Saul, L. K., December 2000. Nonlinear dimensionality reduction by locally linear embedding. Science 290 (5500), 2323–2326.

[155] Rumelhart, D. E., Hinton, G. E., Williams, R. J., 1986. Learning internal representations by error propagation. In: Parallel Distributed processing: explorations in the microstructure of cognition. Computational Models of Cognition and Perception, Volume 1: Foundations. MIT Press, pp. 318–362.

[156] Sagiv, C., Sochen, N. A., Zeevi, Y., June 2006. Integrated active contour for texture segmentation. IEEE Transactions on Image Processing 15 (6), 1633–1646.

[157] Samadani, R., March 1989. Changes in connectivity in active contour models. In: Proceedings of the Workshop on Visual Motion. pp. 337–343.

[158] Sandberg, B., Chan, T., Vese, L., 2002. A level-set and Gabor-based active contour algorithm for segmenting textured images. Tech. rep., UCLA Department of Mathematics CAM report.

[159] Savelonas, M. A., Iakovidis, D. K., Maroulis, D. E., Karkanis, S. A., September 2006. An active contour model guided by LBP distributions. In: Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems. Vol. 4179 of Lecture Notes in Computer Science. Antwerp, Belgium, pp. 197–207.

[160] Schafer, R., Rabiner, L., June 1973. Design and simulation of a speech analysis-synthesis system based on short-time Fourier analysis. IEEE Transactions on Audio and Electroacoustics 21 (3), 165–174.

[161] Scharf, L. L., Cox, D. D., Masreliez, C. J., September 1980. Modulo-2π phase sequence estimation. IEEE Transactions on Information Theory 26 (5), 615–620.

[162] Scharf, L. L., Elliot, H., October 1981. Aspects of dynamic programming in signal and image processing. IEEE Transactions on Automatic Control 26 (5), 1018–1029.

[163] Shi, Y., Chang, E., April 2003. Spectrogram-based formant tracking via particle filters. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing. Vol. 1. pp. I–168–I–171.

[164] Shih, F. Y., Zhang, K., February 2007. Locating object contours in complex background using improved snakes. Computer Vision and Image Understanding 105 (2), 93–98.

[165] Shin, F. B., Kil, D. H., April 1996. Full-spectrum signal processing using a classify-before-detect paradigm. Journal of the Acoustical Society of America 99 (4), 2188–2197.

[166] Slabaugh, G., Unal, G., Fang, T., Wels, M., June 2006. Ultrasound-specific segmentation via decorrelation and statistical region-based active contours. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. Vol. 1. pp. 45–53.

[167] Srinark, T., Kambhamettu, C., September 2006. A framework for multiple snakes and its applications. Pattern Recognition 39 (9), 1555–1565.

[168] Strassen, V., 1969. Gaussian elimination is not optimal. Numer. Math. 13, 354–356.

[169] Streit, R. L., Barrett, R. F., April 1990. Frequency line tracking using hidden Markov models. IEEE Transactions on Acoustics, Speech and Signal Processing 38 (4), 586–598.

[170] Tenenbaum, J. B., de Silva, V., Langford, J. C., December 2000. A global geometric framework for nonlinear dimensionality reduction. Science 290 (5500), 2319–2323.

[171] Tiel, R., April 1976. The passive sonar equation — effects of additive interference. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing. Vol. 1. pp. 675–678.

[172] Urazghildiiev, I. R., Clark, C. W., August 2007. Acoustic detection of North Atlantic Right Whale contact calls using spectrogram-based statistics. Journal of the Acoustical Society of America 122 (2), 769–776.

[173] Urick, R. J., May 1962. Generalized form of the sonar equations. The Journal of the Acoustical Society of America 34 (5), 547–550.

[174] Urick, R. J., 1983. Principles of Underwater Sound, 3rd Edition. McGraw-Hill, New York.

[175] Van Cappel, D., Alinat, P., October 1998. Frequency line extractor using multiple hidden Markov models. In: OCEANS '98 Conference Proceedings. Vol. 3. pp. 1481–1485.

[176] Van der Maaten, L., Hinton, G., November 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9, 2579–2605.

[177] Černý, V., January 1985. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm export. Journal of Optimization Theory and Applications 45 (1), 41–51.

[178] Waite, A. D., March 2002. Sonar for Practising Engineers, 3rd Edition. John Wiley & Sons.

[179] Walker, J., 1996. Fast Fourier Transform, 2nd Edition. CRC Press.

[180] Webb, A. R., Lowe, D., 1990. The optimised internal representation of multilayer classifier networks performs nonlinear discriminant analysis. Neural Networks 3 (4), 367–375.

[181] Weinberger, K. Q., Sha, F., Saul, L. K., 2004. Learning a kernel matrix for nonlinear dimensionality reduction. In: Proceedings of the twenty-first international conference on Machine learning. Vol. 69 of ACM International Conference Proceeding Series. pp. 839–846.

[182] Wenz, G., 1962. Acoustic ambient noise in the ocean: spectra and sources. Journal of the Acoustical Society of America 34, 1936–1956.

[183] Widrow, B., Hoff, M. E., August 1960. Adaptive switching circuits. In: IRE WESCON Convention Record. Vol. 4. pp. 96–104.

[184] Williams, D. J., Shah, M., January 1992. A fast algorithm for active contours and curvature estimation. CVGIP: Image Understanding 55 (1), 14–26.

[185] Wold, H. O. A., 1963. Forecasting by the chain principle. Time Series Analysis, 471–497.

[186] Wong, G. S. K., Zhu, S., March 1995. Speed of sound in seawater as a function of salinity, temperature and pressure. Journal of the Acoustical Society of America 97 (3), 1732–1736.

[187] Wong, Y. Y., Yuen, P. C., Tong, C. S., November 1998. Segmented snake for contour detection. Pattern Recognition 31 (11), 1669–1679.

[188] Wu, H.-H., Liu, J.-C., Chui, C., November 2000. A wavelet-frame based image force model for active contouring algorithms. IEEE Transactions on Image Processing 9 (11), 1983–1988.

[189] Xerri, B., Cavassilas, J.-F., Borloz, B., August 2002. Passive tracking in underwater acoustic. Signal Processing 82 (8), 1067–1085.

[190] Xie, X., Evans, R. J., December 1991. Multiple target tracking and multiple frequency line tracking using hidden Markov models. IEEE Transactions on Signal Processing 39 (12), 2659–2676.

[191] Xie, X., Evans, R. J., January 1993. Multiple frequency line tracking with hidden Markov models — further results. IEEE Transactions on Signal Processing 41 (1), 334–343.

[192] Xu, C., Prince, J. L., June 1997. Gradient vector flow: a new external force for snakes. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. San Juan, Puerto Rico, pp. 66–71.

[193] Xu, C., Prince, J. L., March 1998. Snakes, shapes, and gradient vector flow. IEEE Transactions on Image Processing 7 (3), 359–369.

[194] Xu, G., Segawa, E., Tsuji, S., July 1994. Robust active contours with insensitive parameters. Pattern Recognition 27 (7), 879–884.

[195] Yan, S., Xu, D., Zhang, B., Zhang, H.-J., Yang, Q., Lin, S., January 2007. Graph embedding and extensions: A general framework for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1), 40–51.

[196] Yang, S., Li, Z., Wang, X., July 2002. Ship recognition via its radiated sound: the fractal based approaches. Journal of the Acoustical Society of America 11 (1), 172–177.

[197] Yang, Y. H., June 1990. Relaxation method applied to lofargram. Master's thesis, Naval Postgraduate School, Monterey, U.S.A.

[198] Zhang, Z. Y., Zha, H. Y., January 2004. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. SIAM Journal of Scientific Computing 26 (1), 131–338.

[199] Zhu, S. C., Yuille, A., September 1996. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (9), 884–900.

Author Index

Subject Index