Article:
Lampert, Thomas and O'Keefe, Simon orcid.org/0000-0001-5957-2474 (2011) A Detailed Investigation into Low-Level Feature Detection in Spectrogram Images. Pattern recognition. pp. 2076-2092. ISSN 0031-3203

https://doi.org/10.1016/j.patcog.2011.02.014

A Detailed Investigation into Low-Level Feature Detection in Spectrogram Images

Thomas A. Lampert*, Simon E. M. O'Keefe*

Department of Computer Science, University of York, Deramore Lane, York, YO10 5GH, UK.

Abstract

Being the first stage of analysis within an image, low-level feature detection is a crucial step in the image analysis process and, as such, deserves suitable attention. This paper presents a systematic investigation into low-level feature detection in spectrogram images. The result of which is the identification of frequency tracks. Analysis of the literature identifies different strategies for accomplishing low-level feature detection. Nevertheless, the advantages and disadvantages of each are not explicitly investigated. Three model-based detection strategies are outlined, each extracting an increasing amount of information from the spectrogram, and, through ROC analysis, it is shown that at increasing levels of extraction the detection rates increase. Nevertheless, further investigation suggests that model-based detection has a limitation—it is not computationally feasible to fully evaluate the model of even a simple sinusoidal track. Therefore, alternative approaches, such as dimensionality reduction, are investigated to reduce the complex search space. It is shown that, if carefully selected, these techniques can approach the detection rates of model-based strategies that perform the same level of information extraction. The implementations used to derive the results presented within this paper are available online from `http://stdetect.googlecode.com`.

Keywords: Spectrogram, Low-Level Feature Detection, Periodic Time Series, Remote Sensing, Line Detection

1. Introduction

The problem of detecting tracks in a spectrogram (also known as a LOFARgram, periodogram, sonogram, or spectral waterfall), particularly in underwater environments, has been investigated since the spectrogram's introduction in the mid 1940s by Koenig et al. [26]. 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 [41], image processing [1, 10] and expert systems [35]. 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" [44]. 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

*Corresponding author. Tel.: +44 (0)1904 325500; fax: +44 (0)1904 567687.

Email addresses: `tomal@cs.york.ac.uk` (Thomas A. Lampert), `sok@cs.york.ac.uk` (Simon E. M. O'Keefe)

which is malfunctioning. Applications within these areas are wide and include identifying and tracking marine mammals via their calls [39, 36], identifying ships, torpedoes or submarines via the noise radiated by their mechanical movements such as propeller blades and machinery [52, 7], distinguishing underwater events such as ice cracking [16] and earth quakes [20] from different types of source, meteor detection, speech formant tracking [47] and so on. Recent advances in torpedo technology 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 [36, 39] has resulted in increased data collection which requires automated algorithms to detect calls and determine local specie population and numbers. The research presented in this paper is applicable to any area of science in which it is necessary to detect frequency components within time-series data.

A spectrogram is a visual representation of the distribution of acoustic energy across frequencies and over time, and is formally defined in [29]. The vertical axis of a spectrogram typically represents time, the horizontal axis represents the discrete frequency steps, and the amount of power detected is represented as the intensity at each time-frequency point. For a complete review of spectrogram track detection methods the reader is referred to a recently published survey of spectrogram track detection algorithms [29].

The methods presented can be reduced to, and can therefore be characterised by, their low-level feature detection mechanisms. Low-level feature detection is the first stage in the detection of any object within an image and it is therefore key to any higher level processing. 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 [18]. It is found that a number of mechanisms are in use, however, there exists no systematic investigation into the advantages and disadvantages of each. Abel et al. [1], Di Martino et al. [9], Scharf and Elliot [46] and Paris and Jauffret [41], 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. Methods such as those presented by Gillespie [17], Kendall et al. [25] and Leeming [34] use windows in a spectrogram to train neural network classifiers—the benefits of this, however, were not investigated and the research was probably motivated for the ability to use neural networks. In addition to intensity information, their approach allowed for information regarding the track structure to be 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 paper 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 'unconstrained' detectors as they:

- perform an exhaustive search of the feature space;
- retain all of the information provided to them by the feature model;
- utilise the original, unprocessed, data.

The exhaustive search performed by these methods, however, means that they are computationally expensive and, as such, a number of 'constrained' detectors are examined. These 'constrained' detectors are characterised by one or more of the following:

- machine-learning techniques are utilised for class modelling;
- the data is transformed through dimensionality reduction;
- the data is transformed through preprocessing,

and therefore these detection techniques simplify the search space. All of the 'constrained' feature detectors evaluated derive feature vectors from within a window and they therefore act upon intensity and structural information. The 'constrained' 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 a model-based 'unconstrained' feature detector is compared against a model-based 'constrained' feature detector to ascertain the degree of performance divergence between the two approaches.

Furthermore, this paper 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 paper through a comparison with the detection performance achieved by the low-level feature detectors when applied to the original spectrogram.

The remainder of this paper is organised as follows: Section 2 presents the low-level detection mechanisms; these are evaluated in Section 3 and a discussion of findings is presented; and finally the conclusions of the investigation are drawn in Section 4.

2. Method

In this section several low-level feature detection mechanisms are described and investigated. By definition, the detection of lines and edges forms two distinct problems and is commonly approached differently [18]; an edge is defined by a step function, and a line by a ridge function. Edge detectors such as the Canny operator, along with more recent methods [32], are specifically defined to detect step features and are therefore not evaluated here. The Laplacian detector is, however, an edge detector which can be applied to line detection [18] and therefore it is evaluated in Section 3 of this paper.

2.1. 'Unconstrained' Feature Detectors

Detection methods that utilise dimensionality reduction techniques such as principal component analysis [22] to reduce the model or data complexity, lose information regarding the feature model in the process [6]. Preprocessing of the data also introduces information loss. This information loss detracts from a detector's 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 unprocessed data by varying its parameters. In this section are defined three detection methods with the properties of an 'unconstrained' feature detector, i.e. no model reduction or approximation is performed during the search for the feature, and no pre-processing of the data that may destroy information is carried out (for example filtering or calculating gradient information). 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" [8] and, therefore, "can be expressed as a function of the time" [8], 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.

2.1.1. 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 [12]. 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 [12], 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 [45] and Barrett and McMahon [2] 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.

Intensity Distribution Models. 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 [6]. 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 255 in the training set). These were then histogrammed into 1000 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 [12] as this has been shown to reach the Bayesian estimation under such conditions [6] and are simpler to evaluate [12] (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 Figure 1. As such, the class-conditional probabilities of a pixel value s_{ij} in the spectrogram $\boldsymbol{S} = [s_{ij}]_{N\times M}$, given the hypotheses of noise h_1 and of signal h_2, are determined such that

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

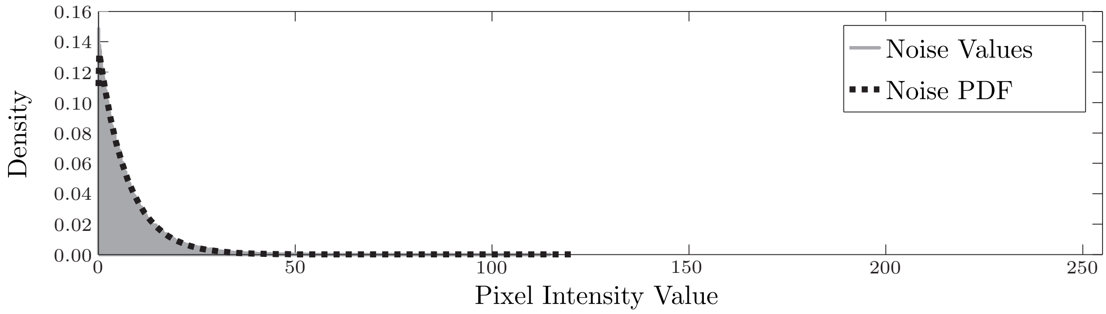
where $x > 0$, the term Γ represents the gamma distribution and the distribution parameters were found to be $\alpha = 1.1439$, $\beta = 20.3073$ and $\lambda = 7.2764$ (with standard errors of 0.0029, 0.0576 and 0.0144 respectively).

The histograms presented in Figure 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.

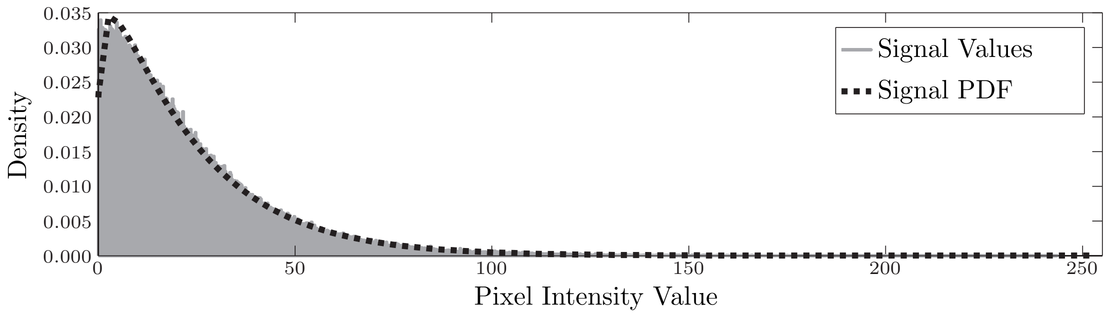
Decision Rules. The simplest form of Bayesian inference, referred to as maximum likelihood (ML) [38], is to calculate the class for which the pixel intensity value has the maximum membership. By defining a set of candidate hypotheses $H = \{h_1, h_2\}$, where h_1 and h_2 are the hypotheses that an observation is a member of the noise or signal class, respectively, and the probability density functions corresponding to these hypotheses, given the data $s_{ij}, \forall i \in N \wedge j \in M$, the likelihood that the data is a result of each hypothesis is determined, such that

$$h_{ML} = \operatorname*{arg\,max}_{h\in H} P(s_{ij}|h). \tag{2}$$

When all the hypotheses in H have equal likelihood of being true any convenient tie breaking rule can be taken [12], in this case a random classification is made.



(a) Noise modelled using an exponential PDF.



(b) Track modelled using a gamma PDF.

Figure 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 1000 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.

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 [12]. The a posteriori probability $P(h|s_{ij})$, which combines the class-conditional and prior, can be computed with Bayes formula,

$$P(h|s_{ij}) = \frac{P(s_{ij}|h)P(h)}{P(s_{ij})}. \tag{3}$$

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

$$h_{MAP} = \underset{h \in H}{\arg\max} \frac{P(s_{ij}|h)P(h)}{P(s_{ij})}. \tag{4}$$

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 $P(s_{ij})$ is a normalisation term, which is independent of h, and therefore, does not influence the decision. It can therefore be dropped [12] and Equation (4) reduces to

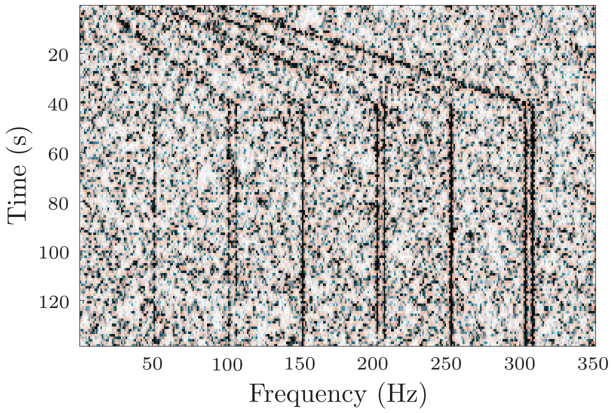
$$h_{MAP} = \underset{h \in H}{\arg\max}\ P(s_{ij}|h)P(h). \tag{5}$$

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 [6], 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.

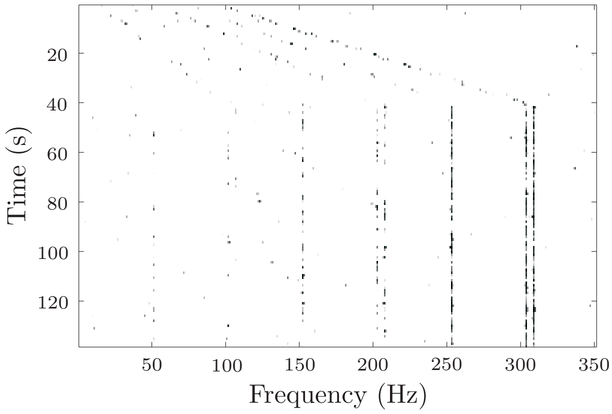
An example of a spectrogram's membership of the noise and track class is presented in Figure 2, Figure 2a presents the noise membership values of each pixel. It can be seen that the majority of noise pixels have a large likelihood of belonging to the noise class. Nevertheless, the high noise values are 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 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 Equation (2), a classification of the spectrogram is obtained, Figure 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.

2.1.2. Bayesian Inference using Spatial Information

Classification based upon single-pixel values is limited to forming a decision using only intensity information. Assuming that a track is defined as a narrowband component of energy that 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. [43], Sildam [48] and Di Martino et al. [8] 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.



(a) Noise class membership.



(b) Track class membership.

Figure 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.

Window Function. The spectrogram $\boldsymbol{S}$, can be broken down into I overlapping windows $\boldsymbol{W}$ of predefined size, such that

$$\boldsymbol{W}_{ij} = \begin{bmatrix} s_{i-\rho,j-\gamma} & \cdots & s_{i-\rho,j-1} & s_{i-\rho,j} & s_{i-\rho,j+1} & \cdots & s_{i-\rho,j+\gamma} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \iddots & \vdots \\ s_{i-1,j-\gamma} & \cdots & s_{i-1,j-1} & s_{i-1,j} & s_{i-1,j+1} & \cdots & s_{i-1,j+\gamma} \\ s_{i,j-\gamma} & \cdots & s_{i,j-1} & s_{ij} & s_{i,j+1} & \cdots & s_{i,j+\gamma} \\ s_{i+1,j-\gamma} & \cdots & s_{i+1,j-1} & s_{i+1,j} & s_{i+1,j+1} & \cdots & s_{i+1,j+\gamma} \\ \vdots & \iddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{i+\rho,j-\gamma} & \cdots & s_{i+\rho,j-1} & s_{i+\rho,j} & s_{i+\rho,j+1} & \cdots & s_{i+\rho,j+\gamma} \end{bmatrix} \tag{6}$$

$$\gamma = \left\lfloor \frac{M'}{2} \right\rfloor, \qquad \rho = \left\lfloor \frac{N'}{2} \right\rfloor$$

where $M' \in \mathbb{N}$ and $N' \in \mathbb{N}$ are odd numbers defining the size of the window (width and height respectively) such that $\gamma < j < M - \gamma$ and $\rho < i < N - \rho$, and therefore $I = (N - 2\rho)(M - 2\gamma)$. A row vector, $\boldsymbol{V}^{ij}$ of size $d = M'N'$, can be constructed from the values contained within window $\boldsymbol{W}_{ij}$ in a column-wise fashion where $\boldsymbol{C}_r^{ij}$ contains values from the rth column of $\boldsymbol{W}_{ij}$, such that

$$\boldsymbol{C}_r^{ij} = [s_{i-\rho,j-\gamma+r} \; s_{i-\rho+1,j-\gamma+r} \; \ldots \; s_{i+\rho,j-\gamma+r}] \tag{7}$$

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

$$\boldsymbol{V}^{ij} = [\boldsymbol{C}_0^{ij} \; \boldsymbol{C}_1^{ij} \; \ldots \; \boldsymbol{C}_{M'-1}^{ij}]. \tag{8}$$

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 [12], such that

$$h_{coML} = \operatorname*{arg\,max}_{h \in H} \prod_{k=1}^{d} P(\boldsymbol{V}_k^{ij} | h). \tag{9}$$

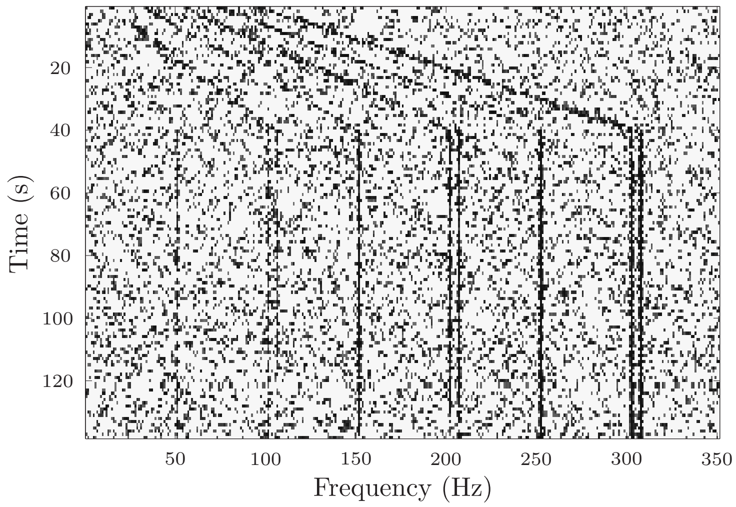


Figure 3: An example of maximum likelihood spectrogram pixel classification, in this image likelihood has been encoded as the inverse of intensity and scaled to have a maximum value of 255. The tracks in this spectrogram have SNRs of, from left to right; first three 3 dB, middle three 6 dB and the last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

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

$$h_{coMAP} = \arg\max_{h \in H} \prod_{k=1}^{d} P(h|\boldsymbol{V}_k^{ij}) \tag{10}$$

$$= \arg\max_{h \in H} \prod_{k=1}^{d} P(\boldsymbol{V}_k^{ij}|h)P(h) \tag{11}$$

where $d = |\boldsymbol{V}^{ij}| \triangleq M'N'$ and $\boldsymbol{V}_k^{ij}$ are the cardinality and kth element of the feature vector $\boldsymbol{V}^{ij}$ respectively.

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 [12].

2.1.3. 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 subsection 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 [30]. In accordance with the detectors in this section, this detector can also be viewed as 'unconstrained' because it detects all variations of the parameters defining the arrangement of pixels belonging to a track within a window in an exhaustive fashion and it also performs this analysis upon the original unprocessed data.

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

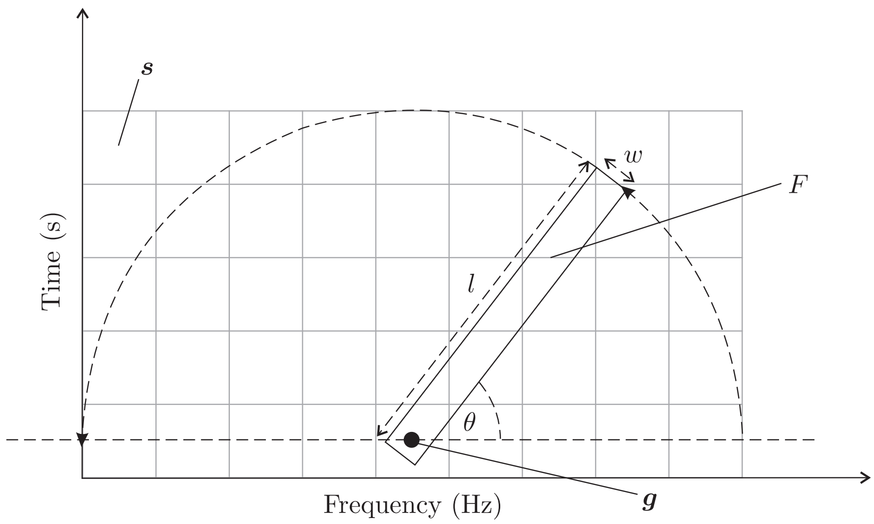


Figure 4: The bar operator, having the properties: width w, length l and angle θ.

$\boldsymbol{g} = [x_g, y_g]$, where $x_g \in \{0, 1, \ldots, M-1\}$ and $y_g \in \{0, 1, \ldots, N-1\}$, in a spectrogram $\boldsymbol{S}$, such that $\boldsymbol{g} \in \boldsymbol{S}$ where $\boldsymbol{s} = [x_s, y_s]$, $x_s \in \{0, 1, \ldots, M-1\}$ and $y_s \in \{0, 1, \ldots, N-1\}$, and extends in the direction of the l previous observations, see Figure 4. The values of the pixels that are 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

$$\begin{aligned} P_l(\boldsymbol{s}, \theta, l) &\iff \quad 0 \leq [cos(\theta), sin(\theta)][\boldsymbol{s} - \boldsymbol{g}]^T < l \\ P_w(\boldsymbol{s}, \theta, w) &\iff \quad \left|[-sin(\theta), cos(\theta)][\boldsymbol{s} - \boldsymbol{g}]^T\right| < \tfrac{w}{2}, \end{aligned} \tag{12}$$

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

$$B(\theta, l, w) = \frac{1}{|F|} \sum_{\boldsymbol{f} \in F} \boldsymbol{f}. \tag{13}$$

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 $\bar{B}(\theta, l, w)$ [40], which is also normalised with respect to the background noise, such that

$$\bar{B}(\theta, l, w) = \frac{1}{\sigma(B)} \left[B(\theta, l, w) - \mu(B)\right] \tag{14}$$

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

$$\theta_l = \arg\max_{\theta} \bar{B}(\theta, l, w). \tag{15}$$

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,

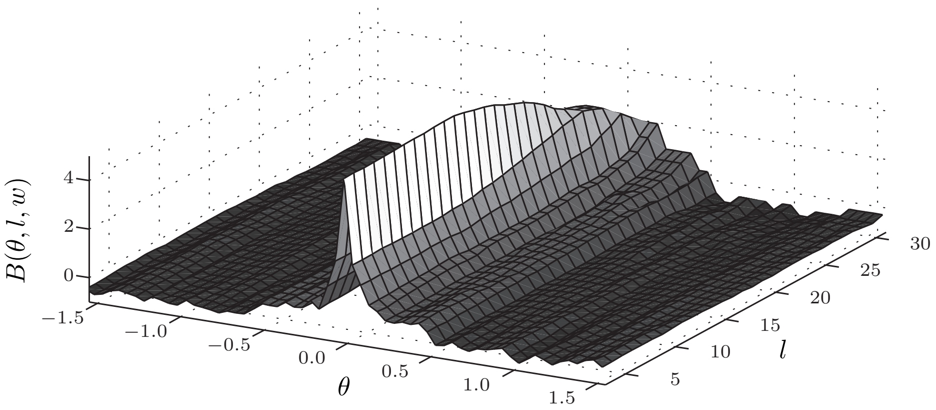


Figure 5: The mean response of the rotated bar operator centred upon a vertical line 21 pixels in length (of varying SNRs). The bar is varied in length between 3 and 31 pixels whilst its width, w, is fixed at 1 pixel.

as shown in Figure 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 Equation (15) is replaced with

$$\theta_l = \arg\max_{\theta} \sum_{l \in L} \bar{B}(\theta, l, w), \tag{16}$$

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 Figure 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

$$L_p = \{l \in L : \bar{B}(\theta_l, l, w) > \frac{3}{4} \max(\bar{B}(\theta_l, l, w))\}. \tag{17}$$

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).

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 efficient binary search algorithm outlined in Algorithm 1. Implementing the search in this way reduces the associated search costs from $O(n)$ to $O(\log n)$, 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.

2.2. 'Constrained' Feature Detectors

A limitation of the 'unconstrained', correlation detection methods is that they are computationally feasible only for models with few parameters and small amounts of data. As the number of parameters increase, the size of the search space increases exponentially—forming an intractable solution. For example,

Algorithm 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, $\boldsymbol{S}$, a spectrogram image
Output: l_l, the length of an underlying line.

if $\bar{B}(\theta_l, l_{\text{low}}, w) > T$ **then**
    $p_{\text{low}} \leftarrow l_{\text{low}} + 1$
    $p_{\text{high}} \leftarrow l_{\text{high}} + 1$
    while $p_{\text{low}} \neq l_{\text{low}} \wedge p_{\text{high}} \neq l_{\text{high}}$ **do**
        $p_{\text{low}} \leftarrow l_{\text{low}}$
        $p_{\text{low}} \leftarrow l_{\text{high}}$
        $l \leftarrow \lfloor \frac{l_{\text{low}} + l_{\text{high}}}{2} \rfloor$
        if $\bar{B}(\theta_l, l, w) > T$ **then**
            $l_{\text{low}} \leftarrow l$
        else {*the line's length has been exceeded*}
            $l_{\text{high}} \leftarrow l$
        end if
    end while
    $l_l \leftarrow l_{\text{low}}$
else {*a line does not exist*}
    $l_l \leftarrow 0$
end if
return l_l

a simple deterministic sinusoidal model contains five free parameters: fundamental frequency position; scaling; track amplitude; phase; and frequency, and 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 [12]; 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 [6]. 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 'unconstrained' detectors due to information loss. Nevertheless, the increase in computing performance, and the non-specificity that occurs as a result of the problem simplification ('unconstrained' 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 subsection presents feature detection methods that are examples of both approaches.

2.2.1. Data-Based Subspace Learning

It is common in the area of machine learning that a classification, or decision, is based upon experience [37]. 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 [12]. In the problem of

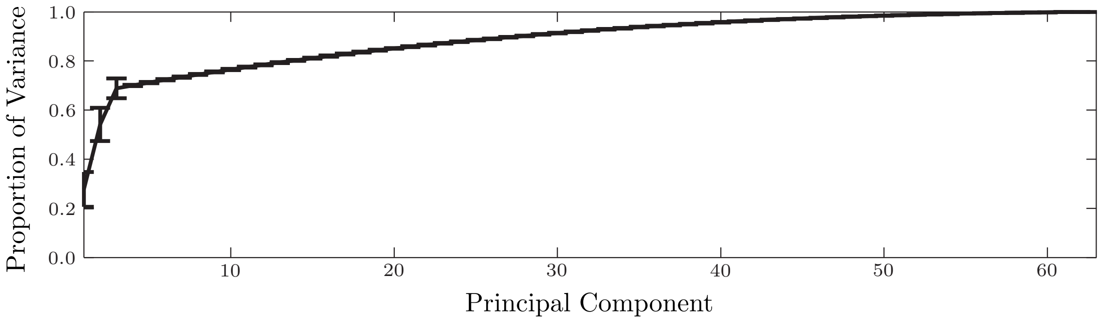


Figure 6: Windowed spectrogram PCA eigenvalues. The eigenvalues were determined using a data set of 1000 samples data samples of each class taken from spectrograms having a mean SNR of 8 dB.

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 2.1.2 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 subsection.

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.

Recently 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 are numerous and include, to name but a few: Laplacian eigenmaps (LE) [4], local tangent space aligning (LTSA) [53], essential loops [33], neural networks [19], t-SNE [49], and general graph based frameworks to unify different dimensionality reduction techniques [51]. 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 [5, 31, 21]. It is beyond the scope of this manuscript to evaluate these methods with application to this data and therefore this subsection concentrates on evaluating the well established techniques of principal component analysis (PCA) [42, 15], linear discriminant analysis (LDA) [3] 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 [6]. 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 [38].

The eigenvalues obtained by applying principal component analysis to a training set comprising 1000 data samples (3×21 pixel, width and height, window instances) of each class randomly selected from a spectrogram having a SNR of 8 dB are presented in Figure 6. A majority of the data's variance is

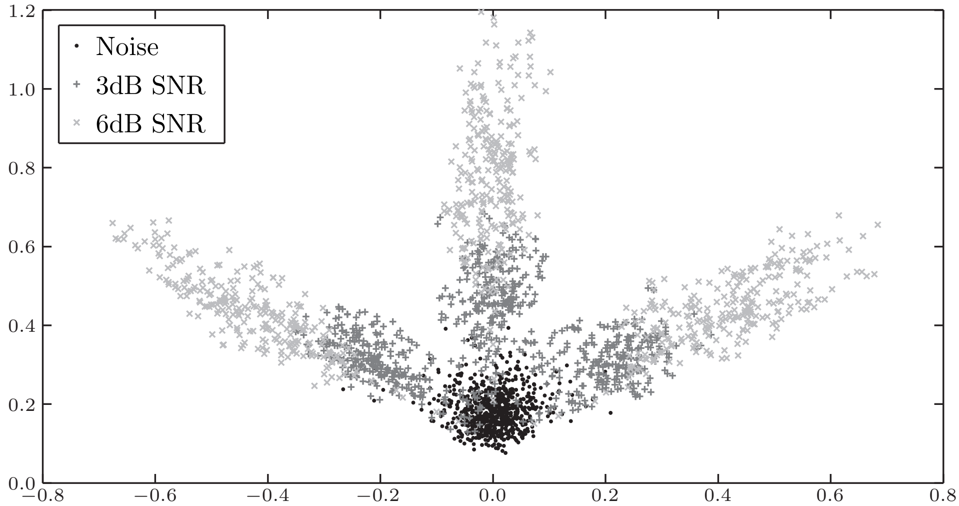


Figure 7: A windowed spectrogram projected onto the first two principal components. The noise, which is clustered between the three spokes, is represented as dots. Increasing the SNR of a track increases it's distance from the noise cluster; windows containing a 3 dB track are represented as a horizontal crosses and those with 6 dB by diagonal crosses.

contained within the first three principal components and the remaining components have little variance. Figure 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 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 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 Figure 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.

Implicit Dimension Reduction. Neural networks perform dimensionality reduction when in specific topologies [23]—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 [6]. The same 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) [27, 28] 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.

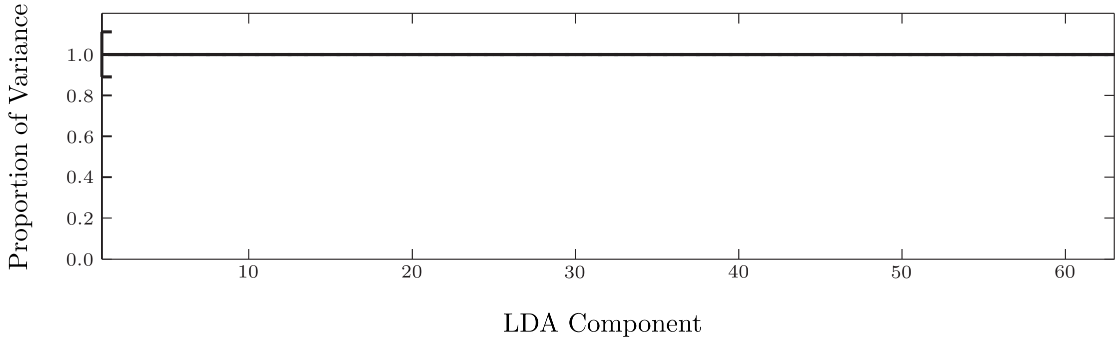


Figure 8: Windowed spectrogram LDA eigenvalues. The eigenvalues were determined using a data set of 1000 samples data samples of each class taken from spectrograms having a mean SNR of 8 dB.

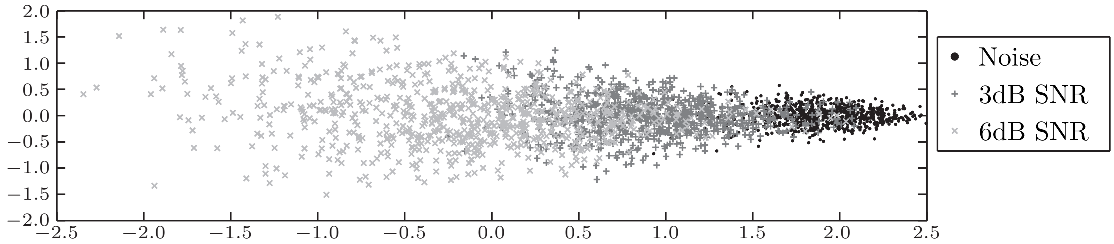


Figure 9: A windowed spectrogram projected onto the first two LDA principal components. The noise is clustered at the right of the distribution and is represented as dots. The windows containing tracks form clusters which overlap with the noise cluster and are projected within increasing distance from the noise cluster as their SNR increases; windows containing a 3 dB track are represented as a horizontal crosses and those with 6 dB by diagonal crosses.

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 $\boldsymbol{V}^{ij}$, is taken to be the classification of the feature vector, such that

$$c^k = \underset{c \in C}{\arg\min}\; d(\boldsymbol{V}^{ij}, \boldsymbol{\mu}_c). \tag{18}$$

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

$$d_1(\boldsymbol{V}^{ij}, \boldsymbol{\mu}_c) = \sqrt{(\boldsymbol{V}^{ij} - \boldsymbol{\mu}_c)^T (\boldsymbol{V}^{ij} - \boldsymbol{\mu}_c)} \tag{19}$$

$$d_2(\boldsymbol{V}^{ij}, \boldsymbol{\mu}_c) = \sqrt{(\boldsymbol{V}^{ij} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\boldsymbol{V}^{ij} - \boldsymbol{\mu}_c)} \tag{20}$$

where $\boldsymbol{V}^{ij}$ 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

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	<u>81.8</u>	<u>81.9</u>	<u>**84.4**</u>	<u>83.8</u>	<u>83.3</u>	81.8	<u>82.1</u>	<u>81.8</u>	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	<u>**82.3**</u>	81.4	80.5	<u>80.9</u>
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 ($G(\boldsymbol{V}^{ij})$) — tr	50.1	66.1	71.8	73.5	**74.8**	61.0	65.6	67.4	69.5
Gaussian ($G(\boldsymbol{V}^{ij})$) — te	50.3	76.1	81.5	82.0	**82.2**	68.1	72.3	74.4	74.8

Table 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 2.

is therefore formed by modelling the noise class with a multivariate Gaussian distribution,

$$G(\boldsymbol{V}^{ij}) = \frac{1}{2\pi^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{V}^{ij} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{V}^{ij} - \boldsymbol{\mu})\right\}, \tag{21}$$

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

$$h = \begin{cases} h_1 & \text{if } G(\boldsymbol{V}^{ij}) > \epsilon, \\ h_2 & \text{otherwise.} \end{cases} \tag{22}$$

The data used during this experiment was as follows; the training set consisted of 6732 samples of 3×21 pixel windows (width and height) 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.15). 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 when compared with the RBF and SOM networks and therefore results obtained using this classifier are not included in this manuscript.

Each of the classifier's parameters was chosen to maximise generalisation performance and was 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 [6]; 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 [6]. 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 [28]. The Gaussian model defined in Equation (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 1 (and the standard

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 ($G(\boldsymbol{V}^{ij})$) — tr	0.32	6.74	2.82	4.09	3.30	5.92	5.80	4.75	5.00
Gaussian ($G(\boldsymbol{V}^{ij})$) — te	0.50	10.69	2.59	4.80	2.07	2.84	5.77	5.47	3.07

Table 2: Standard deviation of the classification performance presented in Table 1.

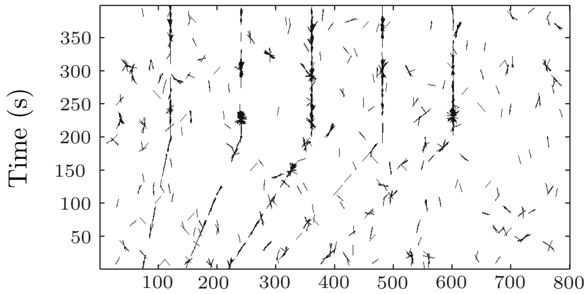
deviations attributed to these results are presented in Table 2). These results demonstrate that classification performance using these features can reach 84% with a standard deviation of 4% when applied to the test dataset (using the RBF classifier in a three-dimensional subspace derived through PCA). 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 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.

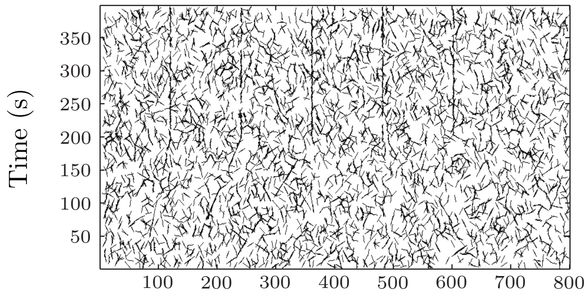
It was shown by Kendall et al. [25] that the generalisation performance of a neural network classifier, when applied to this problem, can be further improved through Ockham's networks [24]. These experiments, however, were conducted, and shown to perform best, on a low number of training samples (121 examples) and therefore this technique was not tested in this section.

2.2.2. 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. [40]. 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.



(a) Bar detection method.



(b) Parametric manifold detection method.

Figure 10: Spectrogram detections (2.18 dB SNR in the frequency domain) using the proposed bar method and the parametric manifold detection method.

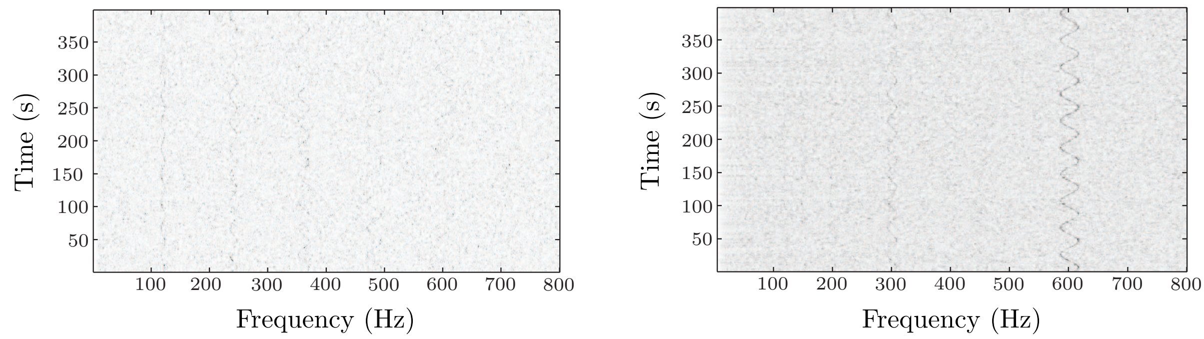
The feature detector proposed by Nayar et al. [40], like the bar detector proposed in Section 2.1.3, is a model-based feature detection method. The primary difference between the two is that Nayar et al. propose to construct a sampled manifold in a feature space derived through PCA. Detection is achieved 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 'unconstrained' and a 'constrained' 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 ($N \times M$) 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 Figure 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 'constrained' 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.

2.3. 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 [2], however, subsequent research has ignored this and instead has focussed on detecting individual tracks.

The acoustic signal emitted by a source comprises of a fundamental 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



(a) Original spectrogram.

(b) Result of the harmonic transform.

Figure 11: 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.

detector, which integrates the energy or detection from harmonically related positions, such that

$$s'_{ij} = \frac{1}{h} \sum_{k=1}^{h} s_{i,[kj]} \tag{23}$$

for $i = 1, 2, \ldots, N$ and $j = 1, 1\frac{1}{h}, 1\frac{1}{2h}, \ldots, M$ and where $[kj] \leq M$, the transformed spectrogram is $\boldsymbol{S}' = [s'_{ij}]_{N \times hM}$, the notation $[kj]$ denotes the nearest integer function and the term h 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 Figure 11. 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 (Section 3.1). 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. Evaluation of Feature Detectors

The feature detectors that are outlined in this paper have been evaluated along with several common line detection methods found in the literature: the Hough transform [11] applied to the original grey-scale spectrogram; the Hough transform applied to a Sobel edge detected spectrogram; convolution of line detection masks [18]; Laplacian line detection [18]; Line Segment Detector (LSD) [50]; and pixel value thresholding [18]. Due to its simplicity and comparable performance to more complex methods, the classification scheme that combines PCA and the Gaussian classifier outlined in Section 2.2.1 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 (see Appendix A, Figure A.14) and, as discussed in Section 2.2.1, that using a window size of 3×21 (width and height) provided acceptable results (Appendix A, Figure A.15).

The remaining operators are now formally defined and are all based upon the convolution operator which is defined such that

$$(\boldsymbol{S} * \boldsymbol{g})_{ij} \stackrel{\text{def}}{=} \sum_{n=-\omega}^{\omega} \sum_{m=-\omega}^{\omega} s_{i-n,j-m} g_{nm}, \tag{24}$$

where $\omega = \lfloor W/2 \rfloor$, $W \in \mathbb{N}$ is odd and is the size of the convolution filter (the size is equal in both dimensions), the origin of the filter is the central element, and where $i = \omega + 1, \ldots, N - \omega$ and $j = \omega + 1, \ldots, M - \omega$.

The convolution output was taken to be

$$\boldsymbol{S}' = [s'_{ij}]_{N-2\omega \times M-2\omega} = [\max_{\boldsymbol{g} \in G} |(\boldsymbol{S} * \boldsymbol{g})_{ij}|]_{N-2\omega \times M-2\omega}, \tag{25}$$

where $G = \{\boldsymbol{g}_1, \boldsymbol{g}_2, \boldsymbol{g}_3, \boldsymbol{g}_4\}$ and the line detection masks used during the convolution experiments were defined such that

$$\boldsymbol{g}_1 = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}, \boldsymbol{g}_2 = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}, \boldsymbol{g}_3 = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}, \boldsymbol{g}_4 = \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix}. \tag{26}$$

The Laplacian operator can be defined as the following kernel and implemented through a convolution operation

$$\boldsymbol{S}' = [(\boldsymbol{S} * \boldsymbol{g})_{ij}]_{N-2\omega \times M-2\omega}, \tag{27}$$

where

$$\boldsymbol{g} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{28}$$

The Sobel edge detector, used as a preprocessing stage to the Hough transform, is implemented as the magnitude of the gradient of two convolution operations, such that

$$\boldsymbol{S}' = \left[\sqrt{(\boldsymbol{S} * \boldsymbol{g}_1)_{ij}}^2 + \sqrt{(\boldsymbol{S} * \boldsymbol{g}_2)_{ij}}^2 \right]_{N-2\omega \times M-2\omega} \tag{29}$$

where

$$\boldsymbol{g}_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \boldsymbol{g}_2 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{30}$$

The performance of each feature detector can be characterised by determining its receiver operating characteristic (ROC) [13]. 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

$$TPR = \frac{TP}{TP + FN} \tag{31}$$

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

$$FPR = \frac{FP}{FP + TN} \tag{32}$$

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 [14], which appears in a special issue of pattern recognition letters dedicated to ROC analysis in pattern recognition.

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

Table 3: Parameter values spanning the synthetic data set.

3.1. Experimental Data

A data set containing 748 spectrogram images is generated for use in the evaluation of the proposed low-level feature detectors (this data set is available from `http://www-users.cs.york.ac.uk/~tomal/data_sets/`). 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 $f_s = 4000\,\text{Hz}$ (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. 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 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 $\text{SNR} = 10\log_{10}\left(\frac{\bar{P}_t}{\bar{P}_b}\right)$ where $\bar{P}_t = \frac{1}{|P_t|}\sum_{(i,j)\in P_t} s_{ij}$, $\bar{P}_b = \frac{1}{|P_b|}\sum_{(i,j)\in P_b} s_{ij}$ and where $P_t = \{(i,j)|s_{ij}$ belongs to a track$\}$ is the set of points related to the frequency components of the signal 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 $P_b \neq \emptyset$.

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. 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 paper are calculated in this manner).

Ground truth spectrograms were created by generating a spectrogram for each parameter combination that have high SNRs (approximately 1000 dB), and then thresholding these to obtain 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.

During spectrogram image's construction, the sampling frequency, f_s, of the time-domain signal should be chosen with respect to the highest frequency component to be detected (according to the Nyquist frequency) to avoid aliasing. Assuming that this guideline is adhered to, each narrowband frequency component within the allowed bandwidth will be represented in the spectrogram as a track and will therefore be detectable

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)	$-\frac{\pi}{2}$–$\frac{\pi}{2}$ (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)	$-\frac{\pi}{2}$–$\frac{\pi}{2}$ (step 0.05)
	Threshold value range	0–255 (step 0.5)
Pixel Thresholding	Threshold value range	0–255 (step 0.2)
PCA	Window size $M' \times N'$ (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)	$-\frac{\pi}{2}$–$\frac{\pi}{2}$ (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 $M' \times N'$(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 4: The parameter values of each detection method that were used during the experimentation.

using low-level feature detection methods such as those outlined in this paper. A further consideration when constructing the spectrogram is the choice of the frequency resolution of the STFT, this should be chosen with respect to the bandwidth of the expected frequency components so as to not spread the tracks needlessly (which will also reduce their SNR and, in turn, make them harder to detect). Nevertheless, the multi-scale line detectors outlined and evaluated in this paper are able to detect tracks with varying widths and are therefore applicable to the detection of tracks with a single pixel width and those with greater widths.

3.2. Results

In this subsection are presented the results obtained during experimentation upon the data set described above. The implementations of the algorithms used during these experiments are available at `http://stdetect.googlecode.com`. The parameters used for each method are described in Table 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 (discounting regions at the border of the images where detection is not possible). The ROC curves are calculated over the whole data set described in Section 3.1, that is, combining results from various SNRs. The performance of each detector at specific SNRs is presented in Appendix A, Figure A.16.

3.2.1. Comparison of 'Unconstrained' Detection Methods

One of the hypotheses proposed in this paper 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

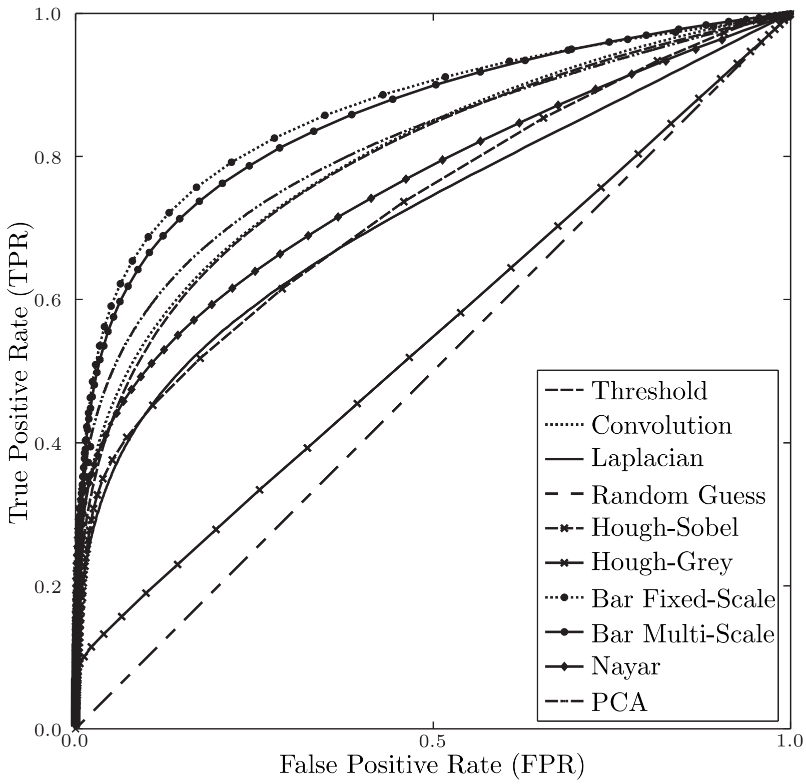


Figure 12: Receiver operating characteristic curves of the evaluated detection methods. The true and false positive rates are described by Equation (31) and Equation (32) respectively.

of this hypothesis is presented in the form of performance measurements for each detector described in this paper, each of which acts upon a different amount and type of information, which is presented in Figure 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 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. Two forms of the bar detector were evaluated, a multi-scale version and a fixed-scale version. The assumption of the feature's length in the fixed-scale implementation allows it to achieve a higher detection rate. This is in contrast to the multi-scale version which empowers the detector to better fit piecewise linear features and approximate curvilinear features, however, this increases the method's sensitivity which impedes higher TPRs and FPRs. Both detectors produce ROC curves that have large separations from existing line detection methods.

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 multi-scale (FPR: 0.134); and bar fixed-scale (FPR: 0.102). These results show that the combination of intensity information and structural information, rather than relying on intensity information alone, increases detector reliability.

3.2.2. Comparison of 'Constrained' Detection Methods

The second hypothesis proposed in this paper was that 'unconstrained' detection methods will outperform 'constrained' 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 Figure 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 'constrained' 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. LSD achieves a TPR of 0.029 and a FPR of 0.001; the detection strategy can be classed as a 'constrained' detector as it is limited by its dependence upon the gradient orientation in the image and therefore does not operate on the original data. In very low SNRs, such as those encountered in spectrogram images, defining features in this way proves to be unreliable. None of the existing methods that were evaluated had comparable performance to the 'unconstrained' or 'constrained' methods outlined in this paper.

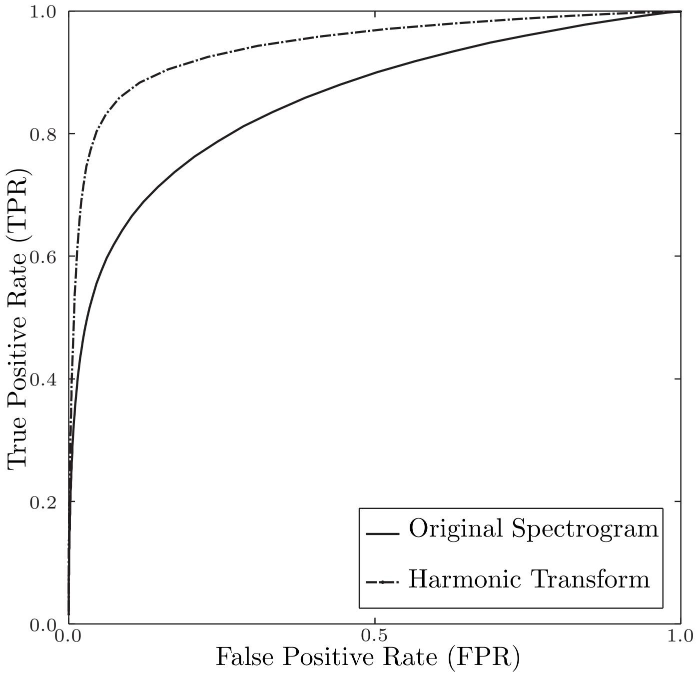


Figure 13: Receiver operating characteristic curves of the bar detector with and without harmonic integration. The true and false positive rates are described by Equation (31) and Equation (32) respectively.

3.2.3. Harmonic Integration

To demonstrate the effectiveness of this simple transformation, the previous experiment is repeated using the top performing detector, the bar detector, and this is applied to the transformed spectrograms, $\boldsymbol{S}'$, as defined by Equation (23) 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 Figure 13 and they demonstrate the vast improvement in the detector's performance that is afforded by this relatively simple transformation.

4. Conclusions

This paper 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 'unconstrained' 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 hidden Markov model, multistage decision process and simulated annealing, that are present in the literature cannot reach the performance of methods that utilise more information in the low-level detection process.

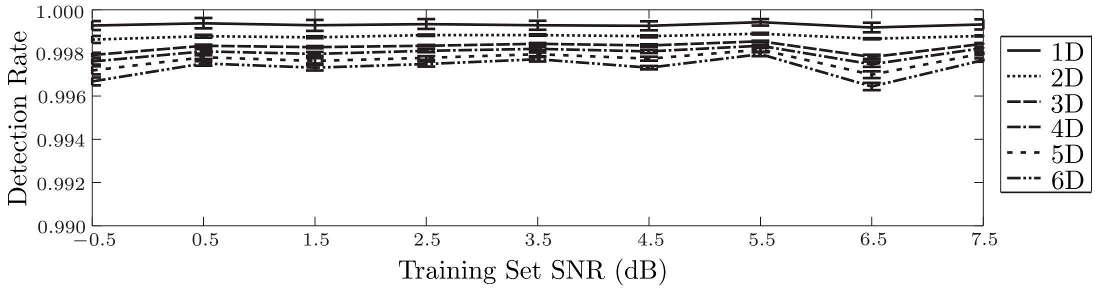
Subsequently, a group of 'constrained' feature detectors were defined that utilise machine-learning principles to simplify the detection process. These were also defined to utilise the maximum amount of information available to facilitate their comparison to the 'bar' detector and were grouped into the 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 'unconstrained' 'bar' detector. Nevertheless, a novel data-based feature detector that utilises principal component analysis was found to be the best performing 'constrained' 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 'unconstrained' and a 'constrained' model-based feature detector, which have equivalent data models, found that the dimension reduction technique used in the 'constrained' detector, whilst reducing computational complexity, vastly reduces detection abilities.

Finally, this paper 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.

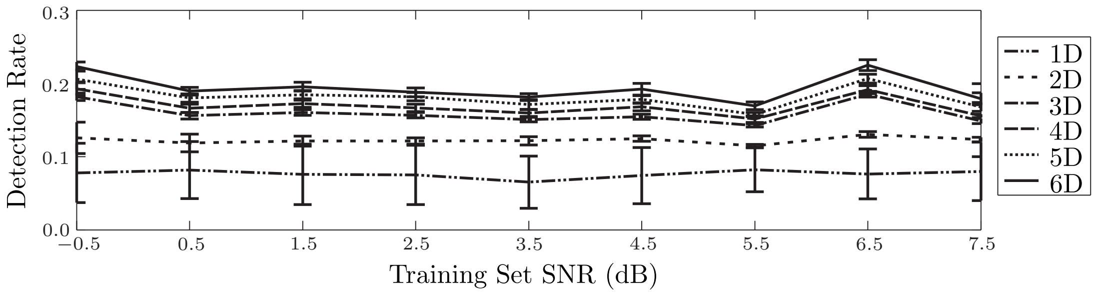
Acknowledgements

This research has been supported by the Defence Science and Technology Laboratory (DSTL) and QinetiQ Ltd., with special thanks to Duncan Williams (DSTL) for guiding the objectives and Jim Nicholson (QinetiQ Ltd.) for guiding the objectives and providing the synthetic data.

Appendix A. Effect of Parameters on Performance

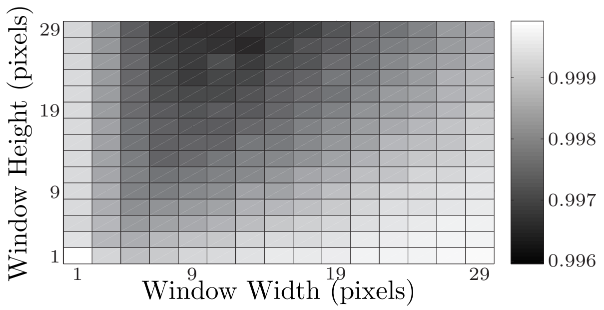


(a) Noise performance.

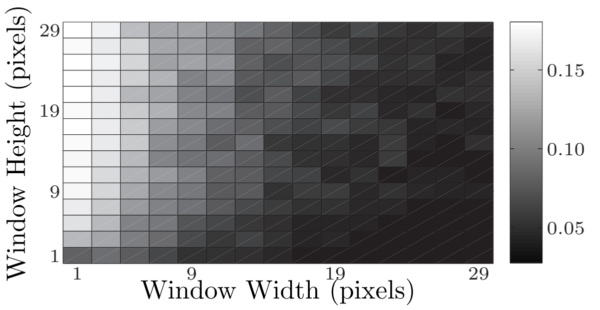


(b) Track performance.

Figure A.14: 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 1000 samples of each class.

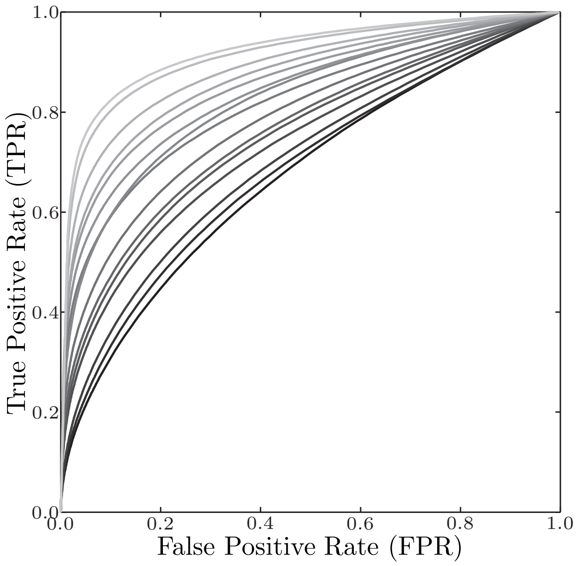


(a) Noise performance.

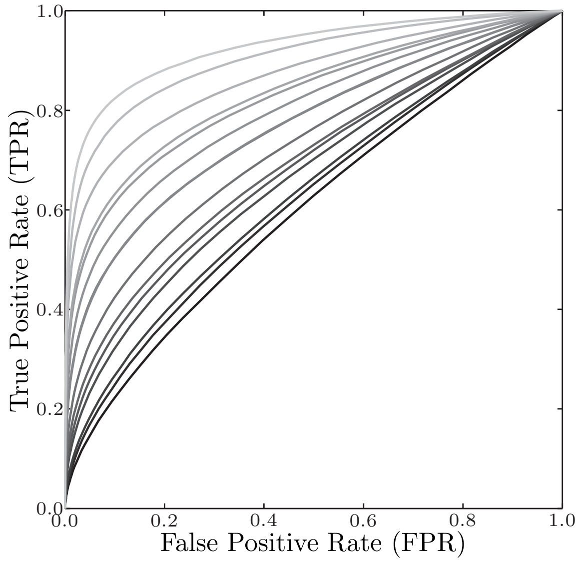


(b) Track performance.

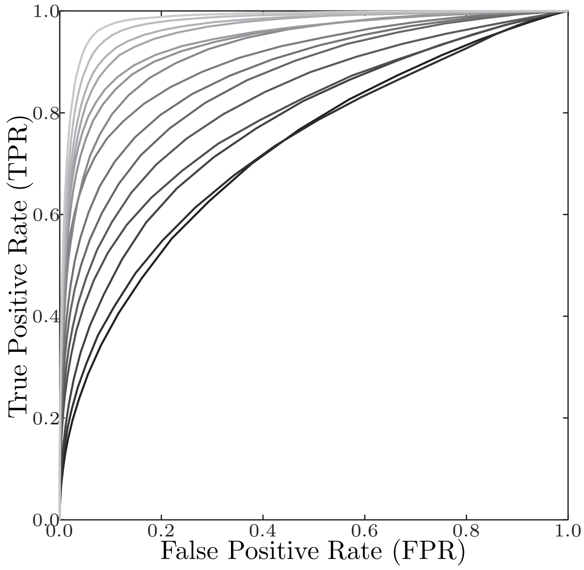
Figure A.15: PCA low-level feature detection performance as a function of the window's height and width. The training set comprised of 1000 samples of each class, the track class having a SNR of −0.5 dB.



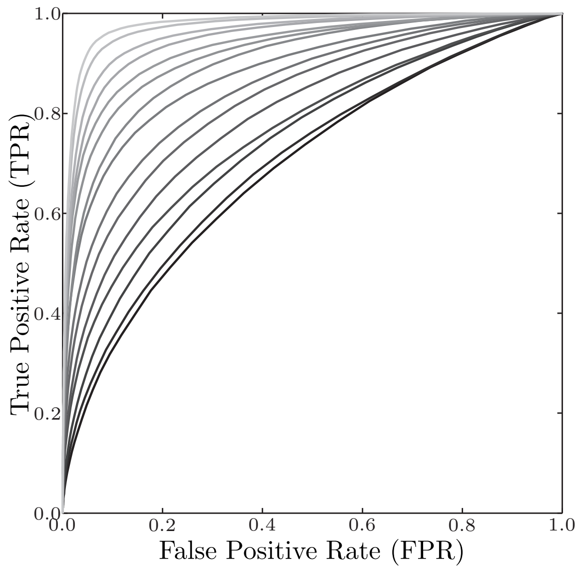
(a) PCA detector.



(b) Nayar detector.



(c) Bar fixed-scale detector.



(d) Bar multi-scale detector.

Figure A.16: Receiver operating curves of each detector's performance at various signal-to-noise ratios. SNRs range from $-1\,\mathrm{dB}$ (darkest) to $6\,\mathrm{dB}$ (lightest) in $0.5\,\mathrm{dB}$ steps.

References

[1] Abel, J. S., Lee, H. J., Lowell, A. P., March 1992. An image processing approach to frequency tracking. In: Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Process. Vol. 2. pp. 561–564.

[2] Barrett, R. F., McMahon, D. R. A., August 1987. ML estimation of the fundamental frequency of a harmonic series. In: Proc. of Int. Conf. on Inf. Sci., Signal Process. and their Appl.ṗp. 333–336.

[3] 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.

[4] Belkin, M., Niyogi, P., 2003. Laplacian eigenmaps and spectral techniques for embedding and clustering. Neural Computations 15 (6), 1373–1396.

[5] 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.

[6] Bishop, C. M., 1995. Neural Networks for Pattern Recognition. Oxford University Press Inc.

[7] 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.

[8] Di Martino, J.-C., Colnet, B., Di Martino, M., 1994. The use of non supervised neural networks to detect lines in lofargram. In: Proc. of the Int. Conf. on Acoust. Speech & Signal Process. pp. 293–296.

[9] Di Martino, J.-C., Haton, J. P., Laporte, A., April 1993. Lofargram line tracking by multistage decision process. In: Proceedings of the IEEE Int. Conf. on Acoustics, Speech and Signal Process. pp. 317–320.

[10] Di Martino, J.-C., Tabbone, S., January 1996. An approach to detect lofar lines. Pattern Recognition Letters 17 (1), 37–46.

[11] 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.

[12] Duda, R. O., Hart, P. E., Stork, D. G., 2000. Pattern Classification. Wiley-Interscience Publication.

[13] Egan, J. P., 1975. Signal detection theory and ROC analysis. Series in Cognition and Perception. Academic Press, New York.

[14] Fawcett, T., June 2006. An introduction to ROC analysis. Pattern Recognition Letters 27 (8), 861–874.

[15] Fukunaga, K., 1990. Introduction to Statistical Pattern Recognition. Elsevier.

[16] 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.

[17] 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.

[18] Gonzalez, R. C., Woods, R. E., 2006. Digital Image Processing, 3rd Edition. Prentice-Hall, Inc.

[19] Hinton, G., Salakhutdinov, R. R., July 2006. Reducing the dimensionality of data with neural networks. Science 313 (5786), 504–507.

[20] 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.

[21] 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.

[22] Jolliffe, I., 2002. Principal Component Analysis, 2nd Edition. Springer.

[23] Karhunen, J., Joutsensalo, J., 1995. Generalizations of principal component analysis, optimization problems, and neural networks. Neural Networks 8 (4), 549–562.

[24] 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.

[25] 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 & Applications 1 (2), 147–159.

[26] Koenig, W., Dunn, H. K., Lacy, L. Y., July 1946. The sound spectrograph. Journal of the Acoustical Society America 18 (1), 244–244.

[27] Kohonen, T., January 1982. Self-organized formation of topologically correct feature maps. Biological Cybernetics 43 (1), 59–69.

[28] Kohonen, T., 2001. Self-Organizing Maps, 3rd Edition. Vol. 30 of Springer Series in Information Sciences. Springer, Heidelberg.

[29] Lampert, T. A., O'Keefe, S. E. M., February 2010. A survey of spectrogram track detection algorithms. Applied Acoustics 71 (2), 87–100.

[30] Lampert, T. A., Pears, N., O'Keefe, S. E. M., September 2009. A multi-scale piecewise linear feature detector for spectrogram tracks. In: Proceedings of the IEEE 6th International Conference on AVSS. pp. 330–335.

[31] 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.

[32] Le, K. N., January 2011. A mathematical approach to edge detection in hyperbolic-distributed and Gaussian-distributed pixel-intensity images using hyperbolic and Gaussian masks. Digital Signal Processing 21 (1), 162–181.

[33] Lee, J. A., Verleysen, M., August 2005. Nonlinear dimensionality reduction of data manifolds with essential loops. Neurocomputing 67, 29–53.

[34] 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.

[35] 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.

[36] 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.

[37] Mitchell, M., 1996. An Introduction to Genetic Algorithms. MIT Press, Cambridge, U.S.A.

[38] Mitchell, T. M., October 1997. Machine Learning. McGraw-Hill, New York.

[39] 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.

[40] Nayar, S., Baker, S., Murase, H., March 1998. Parametric feature detection. International Journal of Computer Vision 27 (1), 471–477.

[41] 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.

[42] Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. Philosophical Magazine 2 (6), 559–572.

[43] 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.

[44] Quinn, B. G., May 1994. Estimating frequency by interpolation using Fourier coefficients. IEEE Transactions on Signal Processing 42 (5), 1264–1268.

[45] 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.

[46] 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.

[47] 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.

[48] Sildam, J., 2010. Masking of time-frequency patterns in applications of passive underwater target detection. EURASIP Journal on Advances in Signal Processing 2010.

[49] Van der Maaten, L., Hinton, G., November 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9, 2579–2605.

[50] von Gioi, R. G., Jakubowicz, J., Morel, J.-M., Randall, G., April 2010. LSD: A fast line segment detector with a false detection control. IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (4), 722–732.

[51] 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.

[52] 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.

[53] 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.