

Detection and One Class Classification of Transient Events in Train Track Noise

J. Arturo Lozano-Angulo

DTU



Kongens Lyngby 2012
IMM-M.Sc.-2012-120

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk IMM-M.Sc.-2012-120

Summary (English)

The thesis is about detection and one class classification of transient events in train track noise. Two different detection approaches have been designed to locate impulsive noise events in train track noise data. They make use of a selected set of features to perform the detection of these events. These approaches are novelty detection based approaches and simple threshold based approaches.

The novelty detection approaches take advantage of the abundance of train track noise, containing no transient events, to create a model of normality of the system. To perform detection, they compare any incoming data to the data model by assessing if the incoming data belongs or not to it.

The simple threshold based approaches apply a threshold to a specific set of feature values extracted from incoming data. Where abnormal high feature values indicate the presence of transient events.

Three different data sets have been extracted from a long duration train track noise measurement to create data models and to test and analyse the different proposed detection techniques. The performance of the detectors is studied from two different points of view. The first one is related to the ROC curve produced by the detectors using a training data set. The second one is related to the consistency of detection results in different data sets.

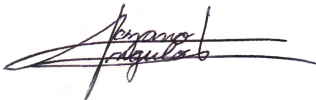
Preface

This thesis was prepared at the DTU Informatics and DTU Electrical Engineering departments at the Technical University of Denmark in collaboration with Brüel and Kjær Sound and Vibration Measurement A/S in fulfilment of the requirements for acquiring an M.Sc. in Engineering.

The thesis deals with the detection and one class classification of transient events in train track noise. Two different approaches are proposed, implemented and analysed, namely, novelty detection based approaches and simple threshold based approaches. The implementations are evaluated by means of a real measurement of train track noise. The thesis consists of a report covering a literature review on the actual related event detection techniques, a theory section, implementation and performance tests description, results, discussion and conclusion including future work. A CD with Matlab code is also included.

The main supervisors for this project were Jan Larsen from the IMM department at DTU, and Shashank Chauhan from Brüel and Kjær Sound and Vibration Measurement A/S. As co-supervisors, Finn T. Agerkvist from the Electrical Engineering department (Acoustic Technology group) at DTU and Karim Haddad from Brüel and Kjær Sound and Vibration Measurement A/S.

Lyngby, 21-September-2012

A handwritten signature in black ink, appearing to read 'J. Arturo Lozano-Angulo', with a stylized flourish at the end.

J. Arturo Lozano-Angulo

Acknowledgements

Firstly, I would like to thank my supervisors at Brüel and Kjær Sound and Vibration Measurement A/S, Shashank Chauhan and Karim Haddad for the opportunity, their support, patience and contributions. Also, thanks to the innovation group at B&K for showing their interest in the project and allowing me to participate in some of their events.

Thanks to my supervisors Finn T. Agerkvist and Jan Larsen at the Technical University of Denmark for their support, comments and feedback, especially for your enlightening different points of view. Thanks to the Acoustic Technology and Hearing Systems groups for their teachings through these 2 years of the Master's programme in Engineering Acoustics.

Thanks to all my friends for your support, enthusiasm and companionship.

Thanks to all my beloved family in Mexico and all around the world. Without you, this whole very enriching experience would not have been possible.

This thesis is dedicated to all of you, especially to my beloved grandfather Álvaro Lozano Vinalay.

Contents

Summary (English)	i
Preface	iii
Acknowledgements	v
1 Introduction	1
2 Literature Review	3
2.1 Novelty Detection	3
2.1.1 Statistical approaches	4
2.1.2 Neural network based approaches	7
2.2 Wavelets	8
2.3 Adaptive Filtering	10
2.4 Statistical Methods	11
2.5 Non-Negative Matrix Factorization	13
2.6 Energy based approaches	14
3 Detection and One Class Classification	17
3.1 Feature extraction	18
3.1.1 Root Mean Square filtering	18
3.1.2 Filtering	19
3.1.3 Short-Time Fourier Transform	19
3.1.4 Non-Negative Matrix Factorization	20
3.1.5 Teager Energy Operator	22
3.1.6 Short Term Energy	22
3.1.7 Coefficient of variation	23
3.1.8 Mel Frequency Cepstral Coefficients	23
3.1.9 Maximum	24

3.1.10 New Features	24
3.2 Event Detection	28
3.2.1 Novelty Detection	28
3.2.2 Simple Threshold Detection	32
4 Methodology	35
4.1 Detector	36
4.1.1 Novelty detectors	38
4.1.2 Simple threshold detectors	46
4.2 Receiver Operating Characteristic	48
4.3 Consistency test	51
4.4 Train track noise	52
4.4.1 Data sets	54
4.4.2 Manual event detection	54
5 Results and Discussion	57
5.1 ROC results	57
5.1.1 Novelty Detectors	57
5.1.2 Simple Threshold Detectors	70
5.2 AUC(0.2) results	75
5.3 Consistency test results	76
6 Conclusion	83
6.1 Future work	85
A Results for ROC and Consistency tests	87
B Detection Framework Structure	95
B.1 Consistency package	96
B.2 Features package	97
B.3 NoveltyDetectors package	99
B.4 ROC package	107
B.5 SimpleThresholdDetectors package	108
B.6 Utilities package	110
Bibliography	113
List of Figures	119
List of Tables	124
List of Abbreviations	127

CHAPTER 1

Introduction

Our modern era is characterized by the enormous amount of information generated everywhere. This increase of information is, at some extent, linked to the capacity of storing it and generating it. For example, the improvements of sensor technology allows us to retrieve information from our surroundings at higher rates and more resolution, increasing in that way the amount of information produced. Most of the time, the storage of this information is not an issue, for example, hard disks have incremented their capacities greatly, where it is common to have a 1 Terabyte hard disk at home. On the other hand, the capacity to analyse all this information has not increased at the same rate. This issue can be observed from two different points of view, a hardware processing point of view and a software processing point of view. We are focused in the software point of view where we are interested in extracting certain events of information from a huge data set.

Every day, society becomes more conscious about life quality in terms of services and habitat surroundings. For example, noise regulations around apartment buildings or office buildings become stricter because the life quality or efficiency of workers would be degraded in the presence of noise. However, our modern life style has also introduced a lot of noise to our lives, from very powerful stereo systems, to huge and noisy vehicles. Thus, a conflict between comfort and quality becomes present. Hence, the importance of noise control in buildings, vehicles, etcetera. Train has been an important means of transportation all

around the world, however, the size of the rail vehicles and the speeds it reaches makes it more susceptible to create high levels of noise. One of the train track defects that produce the highest levels of noise are due to excessive gaps between train track segments. Hence the importance of being able to find this kind of defects. An approach to deal with the problem is to mount microphones on a train and record its journey. Then, by analysing the recorded signals, noisy events can be detected. However, the amount of information collected can be large due to the long distances a train can travel. Thus, the following questions arise. Is it possible to detect train track defects based on a measurement of train track noise? Could the proposed detection techniques perform adequately regardless of the noise level present in the measurements?

The objective of this project is to detect transient events in train track noise by analysing large amounts of recorded train track noise data. A detection task typically consists of feature extraction and application of a detection strategy based on the extracted features. Thus, a set of relevant features is studied and applied through different detection strategies. Moreover, it is important to know if the proposed detection methodologies perform well in terms of the number of transient events they can detect and the consistency of the detection in different data sets. Hence, the proposed detection strategies are evaluated through two performance measures applied to different train track noise data sets. Finally, a framework implemented in Matlab, for feature extraction, event detection and performance evaluation is proposed in this project.

This thesis report is structured in the following way: *Chapter 1* covers this introduction. *Chapter 2* shows a summary of the detection techniques in the relevant fields of transient event detection providing an insight of the useful techniques and features for the goals of this project. *Chapter 3* provides the details of the selected techniques and features based on the literature review, as well as 2 new features proposed by the author. Also, the two main detection strategies are covered. *Chapter 4* provides the details of the designed framework starting with a detailed description of the proposed detectors. The proposed performance measures follow and finally, the data sets with which the detectors were tested are described. *Chapter 5* shows the results obtained as well as an analysis of them. Finally, *Chapter 6* contains the conclusions as well as the future work suggestions. In the *appendix* section, detailed test results can be found as well as a schematic of the framework structure in terms of the Matlab scripts created.

CHAPTER 2

Literature Review

Transient event detection is a really broad subject. The fields that make use of transient event detection techniques range from medical applications, to seismology and machine monitoring, among others. However, not all of the fields are relevant for transient detection in rail track noise. Thus, in this chapter, a brief overview of the main transient event detection approaches related to transient event detection in rail track noise are presented as well as their fields of application.

2.1 Novelty Detection

Novelty detection involves the characterization of the normal behaviour of a system in order to identify when the system is performing outside its normal operation state. The motivation behind novelty detection is that it is often difficult to sufficiently train a system with the data pertaining to the events that are to be detected, due to the fact that this data is not often available, as the event does not occur frequently. Moreover, the deliberated generation of the event might be difficult and expensive. Instead, it is easier to obtain data corresponding to the normal state of the system. Hence, the normal state of the system is modelled by using data corresponding to normal behaviour.

The motivation behind novelty detection is that is often difficult to train a system in all the possible kind of events it may encounter. Therefore, it makes good sense to train it with the information that prevails [MS03a].

According to Markou and Singh [MS03a][MS03b], novelty detection can be divided into two different branches, i.e., statistical approaches and neural network based approaches.

2.1.1 Statistical approaches

The statistical approaches basically model the normal operation of the system based on its statistical properties and define a novelty threshold. Any samples that lie above the novelty threshold are considered as novelties or outliers from the "normal" data. There are different ways in which a distribution model can be built, i.e., using parametric or non-parametric distribution models.

2.1.1.1 Parametric approaches

Parametric approaches assume that the data belongs to a known distribution. The parameters of a distribution are chosen in such a way that the distance between the model and the data is minimized.

Gaussian Mixture Model (GMM)[WZ08] is often used to create data models. However, one of the main problems with GMMs is the dimensionality of the data. If the dimensionality is high, a large number of samples is required to create an accurate model. In the literature, this is known as the curse of dimensionality [Das10]. An example of this limitation can be found in [TNTC99], where the dimensionality of the data consisted of 18 dimensions and only 52 training samples were available. In high dimensional distributions "*the major portion of the probability in the joint distribution lies away from the central region of the variable space. ... Therefore, it becomes difficult to learn about what the distribution is doing in the central region*" [Das10]. Thus, is very unlikely that those 52 training samples could provide enough information about the 18 dimensional distribution. When using GMM, the parameters of the model are estimated such that the log likelihood of the data with respect to the model is maximized. This optimization problem can be solved by using, for example, the Expectation Maximization algorithm [DLR77].

In [CBT07], Clifton et al. monitor aerospace gas-turbine engines and perform on-line novelty detection on it. The vibration of the engine is measured

through transducers placed on various points of the engine. The amplitude of the recorded data is modelled with GMM and the novelty thresholds are related to sensor noise ratings.

In [NPF11], novelty detection is applied to sound surveillance in different environments. The "normal" behaviour of the system is modelled into a GMM. The chosen features are based in those provided by the MPEG-7 audio protocol, Mel frequency cepstral coefficients, intonation, Teager Energy Operator and wavelets. The number of GMM kernel components was varied, showing a strong influence in the final results. The novelty threshold is given by the minimum probability of the training data.

In [WZ08], Weiss et al. classify different terrain types for a moving robot. Their system is able to identify novel terrain types by modelling the training data into a GMM and defining a threshold. If the test data is considered novel, the information is stored until a certain number of similar test data samples are gathered. In that moment the GMM is retrained to be able to identify other kinds of terrains apart from the already modelled.

When using statistical methods there is always the question of where to set the novelty threshold. A good approach to decide this is proposed by Extreme Value Theory (EVT). Extreme Value Theory models the distribution of the extreme values of a distribution. This theory is well developed for uni-modal and univariate distributions [CHT09]. However, the univariate uni-modal EVT is not applicable to GMM. In [CHT09] [CHT11] EVT is extended to multimodal multivariate distributions and is applied to engine and vital-sign monitoring.

Hidden Markov Models (HMM) [Rab89] are also used to do novelty detection, however, they are not very popular in transient event detection. HMMs are used to model sequential data or time dependent data. In [Mil10] [NPF11], they are used to model the "normal" behaviour of a sound surveillance system. The HMM breaks the training data sequences into a predefined number of states and each state is modelled by a GMM. The HMM is trained and its parameters are estimated using the Baum-Welch algorithm [Wel86]. The novelty threshold is defined as the minimum log-likelihood among all training sequences. Moreover, during testing, if a test sample is considered as normal, this new sample is used to update the parameters of the HMM.

In relation to this project, a GMM could be suitable for the modelling of the distribution of the normal state of the system. In this case, the normal state of the system is the train track noise signal containing no transient events in it. However, EVT will not be implemented in this project, it may be considered for the definition of a suitable novelty threshold.

2.1.1.2 Non-parametric approaches

The k-nearest neighbour [IPH09] technique is also used to model the "normal" behaviour of a system, but in a non-parametric manner. The main idea is to find how close a test data point is in relation to the k-nearest neighbours in the training data. One of the main problems with this technique is that the number of computations increase with the number of data points [MS03a].

An example of the application of the k-nearest neighbour technique, among other techniques, can be found in [GMES⁺99]. In this work, the Euclidean distances [TK08] between each point and its neighbours are found for a training data set. A distance proportional to the maximum of these distances is set as the novelty threshold. On every incoming data test point, the distances between its neighbours is calculated and if, it exceeds the threshold, it is considered as a novelty.

In [CO10], Cabral et al. produce a model of normality using an algorithm known as Chameleon. This algorithm uses the k-nearest neighbour technique to create improved quality clusters, which are built in the following way: the training data is split into different clusters defined by the k-nearest neighbour, then the clusters are reorganized through partitioning and merging them using a special criteria. Once the model is created, the authors are able to identify novelty based on the distance to the clusters and also in the number of consecutive test data samples that belong to one particular cluster. This methodology is applied to electrocardiograms and respiration time series.

Parzen density estimation [Sil86] is another method, based on kernel functions, used to estimate data density functions with only few parameters or not parameters at all. In [YC02], Yeung et al. choose radially symmetric Gaussian functions as kernel functions for the estimation method. The main reason, according to the authors, is because this functions are smooth and can be defined by just a variance parameter. The novelty threshold is found using a subset of the whole training data. Even if the application field of this work is different from transient event detection (intrusion detection), the main idea of their method can be applied to transient event detection.

In relation to this project, the k-nearest neighbour technique could be used to find the distances between each of the data points in the transient-free train track noise signal. In this way a novelty threshold could be defined as a proportion of the maximum of the distances between these data points. However, as the amount of data in this work is expected to be high, the k-nearest neighbour technique might not be suitable. On the other hand, Parzen density estimation could be used to model the transient-free train track noise signal. In this way,

a comparison between this technique and other parametric approaches can be done, and an evaluation of which of the two models produced a better representation can be performed.

2.1.2 Neural network based approaches

Many different kinds of neural networks [Sam07] have been used to do novelty detection. Examples of those networks can be found in [MS03b] and [Mil10]. However, according to [MS03b], there is not enough comparative work between the different approaches to assess which techniques work better on different types of data.

In [JMG95] an auto-encoder neural network is applied to helicopter fault detection, DNA pattern recognition and classification of sonar targets. An auto-encoder is a neural network composed of a certain number of input neurons and the same number of output neurons. It also has one hidden layer with less number of neurons than the input. The idea is to train the network with "normal" training data. Then the network is presented with a test sample. If the sum of the absolute error between the input and the output is sufficiently small, the sample is considered "normal", and when the error is above the novelty threshold, the sample is considered "novel".

A Support Vector Machine (SVM) [JF00] is another kind of neural network that can be used to do novelty detection. According to [MMB11], SVMs are able to classify by finding a hyper-plane that divide the classes involved. However, in a novelty detection style, the classification problem is reduced to only one class. Hayton et al. [HUK⁺07] train a SVM that finds a hyper-plane that separates the training data from the origin in feature space, with the largest margin. The test samples that lie in the origin's hyper-plane are considered as novel. Their method is applied to jet engine monitoring. In [MMB11], they use Support Vector Regression (SVR) to find anomalies in water distribution systems. Apart from classification, SVMs can also be used to do regression estimation in time series. SVR models and predicts time series. The modelling or training is done using a "normal" time series. The model is presented to test data and the predicted values are compared to the observed values. When the error between these variables exceed a certain novelty threshold, novelty is detected.

In relation to this project, an auto-encoder could be trained with a transient-free train track noise signal. Then, when presented to any transient events, the network should not be good enough to reconstruct the input in its output, as it has not been trained with this kind of events. In this way, the error would be

larger for transient events and detection could be performed. A Support Vector Machine could be used to find a hyper-plane that separates the transient-free train track noise signal data points. Then, when presented to any transient events, these new points should be located in a hyper-plane different from that of the transient-free data points. However, due to time restrictions, only the auto-encoder technique is explored in this project.

2.2 Wavelets

Wavelet analysis is considered one of the most recent tools for signal processing. Its properties makes it suitable for analysing non-stationary signals in detection applications. Unlike the Fourier transform, which uses sines and cosines as basis functions, the wavelet transform uses a family of other basis functions, or mother wavelet, that describe a signal in space and scale domains, somehow equivalent to time and frequency domains. Thus, the selection of these basis functions is critical for the application of the technique [ABSC11]. The wavelet transform is divided into three kinds, namely, Continuous Wavelet Transform (CWT), Discrete Wavelet Transform (DWT) and Wavelet Packet Transform (WPT). Equation 2.1 shows the definition of the Continuous Wavelet Transform [Mal99], where ψ is the mother wavelet, u is the space or position parameter and s is the scale parameter.

$$Wf(u, s) = \int_{-\infty}^{\infty} f(t) \frac{1}{\sqrt{s}} \psi^* \left(\frac{t-u}{s} \right) dt \quad (2.1)$$

One of the fields that makes extensive use of wavelet transforms is Power Quality (PQ). This field of application consists on the opportune detection and correct identification of electric power supply transient disturbances to protect an increasing number of sensitive electrical equipment connected to the electric supply network [CDS07]. In [RAMG10] and [TR11] the DWT and CWT are used, respectively, to do basic transient event detection. In both studies, the coefficients of one specific scale of the transformed signal are used to detect transient events. When any of the coefficients exceeds a determined threshold a transient event is detected. The basis functions used are the discrete Meyer and Morlet, respectively. No further information in the selection of the basis functions is mentioned.

Another example in PQ is found in [MJG10]. Masoum et al. use the DWT combined with a neural network approach to perform detection and classifica-

tion of transient events. The proposed procedure is to first filter the captured power signals using a DWT. Then, the mother wavelet is selected. The authors went through a thorough test analysis phase considering 40 different types of basis wavelet functions. They concluded "*The choice depends on the nature of the application. For detection of low amplitude, short duration, fast decaying and oscillating types of signals, the most popular wavelets are Daubechies and Symlets families (e.g. db2, db3 etc. and sym2, sym3 etc.). Furthermore, the accuracy of disturbance time localisation decreases as the scale increases. Also, wideness and smoothness of mother wavelet depends on its number. Therefore careful considerations are required for the selection of the suitable wavelet family and its number. In this paper, after many examinations, the sym4 mother wavelet was selected...*"[MJG10]. Once the signal is transformed, the total harmonic distortion in the wavelet domain of the processed input signal is calculated. If this quantity exceeds a certain threshold value, an event is detected and the classification procedures are started.

Machinery monitoring is another example where wavelets are applied for the detection of critical transient events which may lead to faults in the system. Al-Badour et al. [ABSC11] use the CWT and WPT to study transient detection in turbo-machinery. Based on a simulation study, they determine that the proper wavelet basis functions to use for impulsive signals are the Gaussian, Daubechis and discrete Meyer wavelets. The authors are able to detect a specific kind of transient event based on the CWT by adopting the following procedure. After obtaining the transform of the test signal, the local maxima in the coefficients given by the transform are found. From the behaviour of the resulting local maxima lines they are able to detect transient events. The WPT is then applied to the same kind of transient event for comparison, showing that the CWT is able to produce better results. Furthermore, they combine the WPT and CWT to detect other kind of transient events. The procedure consists of decomposing the signal with WPT, reconstructing a specific level of the signal using the coefficients given by the transform and then applying the CWT to the reconstructed signal. Finally, the same local maxima lines criterion to identify the events is used. In this work no automatic detection procedures are mentioned.

Seismology is other field that gathers experience in the detection of transient events [Bar07]. The STA/LTA (Short Time Average to Long Time Average) detector [Bar07] is one of the classic tools for detection in this field and it is based on the analysis of the ratio of the amplitude of a seismic signal in short and long time windows. It consists of two consecutive time windows of different length that move synchronously along a seismograph. At every sample the STA/LTA ratio is calculated. If the ratio exceeds a detection threshold then an event is detected. This approach has two limitations, it cannot specify when an event is over and it does not account for the frequency content of the detected signal. The authors extend the STA/LTA detector to account for the duration of

the event and the frequency analysis of the signal, using an envelope based and WPT based approaches, respectively. Once again, the wavelet basis function and the scale useful for the application are determined empirically.

In summary, wavelets appear to be a powerful tool to analyse non-stationary signals with a great degree of detail and flexibility. However, the selection of the basis function seems to play an important role for the success of the application of the tool. Hence, a thorough study searching for the suitable wavelet basis functions would be needed for the application of this tool in this project. Thus, due to time limitations, wavelets are not covered in this work.

2.3 Adaptive Filtering

An adaptive filter is a special kind of filter that adjusts its parameters depending on the environment where it is used, it can also track signals or system characteristics varying on time [DSS07]. The adjustment of the coefficients is done by an adaptation algorithm. One of the most used adaptation algorithms is the Least Mean Square (LMS) algorithm. This algorithm makes use of a reference signal representing the desired output of the filter. Thus, the algorithm adapts the coefficients of the filter based on its inputs, which are the reference signal and the error. The error is defined as the difference between the reference signal and the output of the filter.

In [Cam99], an adaptive linear predictor is used to enhance electroencephalogram (EEG) signals making easier the manual detection of transient events. The adaptive linear predictor is used to attenuate the stationary components of the EEG enhancing the non-stationary particularities of the signal. The authors put special interest in the order of the filter claiming that a very low order would not attenuate sufficiently the stationary peaks in the EEG, while a very high order would retain false spectral peaks within the filter response. Thus, the authors suggest a modification of the LMS algorithm that enhances the adaptation of the filter coefficients.

An example in PQ where adaptive filtering is applied, not as the main detection technique but as a crucial preprocessing step, is found in [RDR03]. Ribeiro et al. filter the fundamental component of a sinusoidal power signal leaving any transients present in the signal intact. To achieve this filtering step a very good estimation of the fundamental frequency is required. Thus, a cascade adaptive notch filter is used to estimate this frequency. The algorithm to adapt the filter coefficients could be the classical Recursive Least Squares (RLS) or the LMS algorithm. After the transients had been obtained from the original signal, a

DWT and Modulated Lapped Transform (MLT)[RDR03] are applied to obtain a 14-dimensional feature vector to represent them.

Another work in the PQ field is given by [GE05]. As in the last example, the authors use an adaptive filter as a crucial preprocessing step previous to detection. Gerek et al. use an adaptive filter to obtain the non-predictable portion of the signal. The process consists of a decomposition of the input signal into a lower resolution signal and a reference signal. The lower resolution signal is obtained by down-sampling by a factor of 2 the input signal. The reference signal is obtained by delaying one sample of the input signal and down-sampling the resulting signal by a factor of 2. The adaptive filter estimates the reference signal using the lower resolution and the error between the estimate and reference signals as inputs. Thus, the error signal contains the non-predictable portions of the input signal. The main identification idea is that the non-predictable portions of the signal are larger in magnitude when any transient events are present in the input signal. This is due to the imposition of instantaneous spectral components by any transient event. As the adaptation process cannot react instantly to this sudden change, the error of the estimation is increased. The adaptation algorithm used by the filter is again the LMS. After the non-predictable portion of the signal is obtained, its statistical properties are analysed and the detection of transient events is performed.

Adaptive filtering could be used in transient event detection in train track noise by filtering the train track noise signal in such way that the stationary components were eliminated from the signal. The stationary components could be regarded as the part of the signal containing no transient events. Thus, the remaining part of the signal, after filtering, would make more evident the presence of transient events. While this might be possible, it is not clear that the transient-free portion of the signal could be assumed to be stationary. Moreover, other techniques to make more evident the presence of transient events are studied in this project.

2.4 Statistical Methods

The statistical properties of a signal are often used to detect transient events. As already mentioned in the last section [GE05], Gerek et al. use an adaptive filter to obtain the non-predictable portion of a power signal. Then, this error signal is analysed by a statistical decision block to state if there are any transient events. The statistical block analyses the error signal in a sliding window manner and calculates a data histogram for each window. This histogram is considered as an estimate of the local probability density function. The authors state that, when

an event is found, the error signal presents an increase in its variance, which is observed in a heavily tailed histogram. To quantify this increase of variance the authors calculate a ratio between the weight of the central portion of the histogram and the tail portions of the histogram. The resulting ratio is then compared to a threshold to determine if an event happens or not in the specific window. The selection of the threshold is defined as 90% of the ratio between the weight of the central portion of the histogram and the tail portions of the histogram in no event data.

Halim et al. propose a fault detection technique for rotating machinery based on bi-coherence analysis [HCSZ06]. According to the authors, the presence of non-linearities in the signal can indicate the presence of faults in the machinery. Also, they mention that bi-coherence is a tool based on high order statistics [Men91] capable of showing non-linearities in a time signal, but it can only be used with stationary signals. Thus, the authors present a technique to remove the stochastic part of a vibration signal by synchronously averaging a sufficiently large number of rotations. Once the vibration signal is considered stationary, bi-coherence can be applied to find faults in a rotating machine. No automatic detection is suggested in this work.

Another example using high order statistics is found in [dlRML07]. This work is dedicated to the detection and classification of faults in the PQ field. González et al. state that an undisturbed power signal exhibits Gaussian behaviour, and deviations from Gaussianity can be detected with high order statistics. Their procedure indicates that, after filtering out the fundamental component of a sinusoidal power signal, the calculation of higher order cumulants from the remaining signal, namely, variance, skewness and kurtosis, is done. Furthermore, skewness and kurtosis are normalized to take into account shift and scale changes in the transient signal. No automatic detection is suggested by the authors, but a classification system based on neural networks is afterwards analysed, classifying the events into short and long duration ones, based on the obtained high order cumulants.

Statistical methods are definitely applicable in this project, looking for a change in the statistical properties of the signal in the presence of transient events. While bi-coherence analysis cannot be applied in this project as a train track noise signal cannot be regarded as stationary, other statistical approaches are studied such as a feature based on the standard deviation and mean of a train track noise signal.

2.5 Non-Negative Matrix Factorization

Non-Negative Matrix Factorization (NMF) [LS01] decomposes, in an approximated manner, a non-negative matrix V into the product of two non-negative matrices W and H . Each of the columns of V is approximated by a linear combination of the columns in W and the activation coefficients of each component are contained in the columns of H . *"Therefore W can be regarded as containing a basis that is optimized for the linear approximation of the data in V "* [LS01].

In [WLCS06], Wang et al. apply NMF to the onset detection, or the detection of the starting time, of sound events. The authors try two different choices of matrix V , that is, the magnitude of the spectrogram and RMS filtered values of the sound time series. Their method consists on, after obtaining the NMF, adding up the H matrix along its first dimension obtaining an approximation of the temporal profile or envelope of the original time signal. Then, they calculate the absolute difference between the neighbouring samples of the added up H matrix. This enhances any sudden changes in the added up H matrix. A threshold can then be applied to the resulting signal finding the onset of the sound events. The spectrogram approach is shown to have better results than the RMS filtered approach. However, the method is applied to a very simplified sound example without any background noise. In [CTS10b], Costantini et al. also use the magnitude of the spectrogram for onset detection in piano music. Their method consists in building a binary representation of the magnitude of the spectrogram by normalizing it and applying a threshold. Then, the binary magnitude of the spectrogram is processed to point out only the spectral changes and remove isolated spectral bins in the time-frequency representation. Finally, they apply the NMF to this processed binary magnitude of the spectrogram decomposing the matrix into one base component. The resulting activation matrix H successfully corresponds to the onset of the piano notes present in their test sample. Constantini et al. continued using NMF for onset detection in piano music in [CTS10a], this time they apply the already described onset detection by Wang et al. [WLCS06] showing also good results.

O'Grady et al. [OP06] illustrate that, when V is formed with the magnitude of a spectrogram, NMF is not expressive enough to decompose a signal with auditory objects that evolve over time. The authors propose an extension to NMF known as convolutive NMF to account for this shortcoming. Moreover, a sparseness constraint is applied on H . This basically reduces the probability of the activation of more than two basis components at once. The methodology is applied to a musical signal where 6 different musical notes are played sequentially and at the same time. Convolutive NMF fails to decompose the signal into just 6 different basis components while convolutive NMF with a sparseness constraint successfully decompose the signal into 6 different basis components.

NMF could be used in this project by decomposing a processed train track noise signal into different basis components and its activation coefficients. Then, the decomposed data might be able to reveal new aspects of the data, useful for the detection purposes of this work.

2.6 Energy based approaches

Some works that use the signal's energy, directly or indirectly, to detect transient events are presented in this section. In [CK10], Chandrika et al. use a psycho-acoustical model for the detection of squeak and rattle events in vehicle cabins. The purpose of using a psycho-acoustical model is to be able to detect events detectable by human operators. The psycho-acoustical model is based in two different psycho-acoustical models. The loudness model by Zwicker and the Glasberg and Moore temporal loudness integration model. Zwicker's model [Zwi77] accounts for the non-linear spectral processing characteristics of the human auditory system. The model consider how the sound energy is distributed along the different auditory filters in humans. Glasberg's and Moore's model [FZ07] accounts for the time domain masking in the temporal integration of loudness. The author's psycho-acoustical model obtains the perceived transient loudness (PTL) which is the main feature of the detection strategy. After the calculation of the PTL, a threshold strategy is applied to obtain the better detection performance. The threshold was obtained for different signal to noise ratios when the sound events were barely noticeable. It was found that the threshold varied with the background noise level. A correlation study showed that the found thresholds were highly correlated with the 75th percentile value of the PTL curve. Thus, an equation was found to determinate a dynamic threshold as a function of the 75th percentile value of the PTL curve.

There is always a great interest to develop fast methods in the detection of transient events in the PQ field. An example of this is found in [SYT11] where the Teager Energy Operator (TEO) [SYT11] is applied obtaining satisfactory results. According to the authors the TEO is a non-linear high pass filter that enhances high frequency components in transient signals and suppresses the low frequency background. One of the most notable properties of the operator is that it is nearly instantaneous, namely, only three samples of the signal are needed for the calculation of the energy at each time instant. The detection algorithm uses a reference signal to obtain a threshold. The reference signal is just a power sinusoidal signal without any transient events. A multiple of the average of the TEO operated reference signal is set as the threshold. The TEO operated test signal is averaged every 5 samples and compared to the threshold. This method allows for the detection of transients and its duration.

Gritti et al. use Short Term Energy (STE) [GBR⁺12] to detect snore sounds during the sleep of people [GBR⁺12]. The STE operator is the energy signal in dB calculated in a window by window manner. A signal is analysed by applying the operator. Then, the histogram of the operated signal is calculated. The author's detector works using a lower and an upper thresholds found with the Otsu method [Ots75]. Once the lower threshold is obtained, the values of the histogram below the lower threshold are removed and the Otsu method is used again to obtain an upper threshold. The Otsu method is used in image processing to select an adequate threshold of grey level to extract objects from their background [Ots75]. The detection mechanism works as follows. When the STE value of an operated signal exceed the upper threshold an event is detected. The beginning of it, is registered as the point in time where the STE value exceeded the lower threshold. When the STE value falls below the lower threshold, the ending of the event is registered. In this way all the events in a signal are detected and some features such as the duration of the event, the maximum value of the event and the average STE of an event are used to train a neural network and perform classification of the found events. The authors report that their method is not very reliable for low intensity events.

The relevance of the energy based approaches is mostly focused in the features they provide. As the kind of transient events to detect are impulsive in nature, it is expected to note changes in energy at higher frequencies. Thus, Teager Energy Operator, as a non-linear high-pass filter, might be suitable for the characterization of this kind of events. Moreover, the presence of transient events represent an increase of the energy in the signal. Hence, Short Term Energy might be useful too, to characterize any transients in the signal.

CHAPTER 3

Detection and One Class Classification

As seen in last chapter, there are many approaches for doing detection of transient events. Some of those approaches are general and can be applied in a broader selection of application fields, while other approaches are designed to be applied in very specific fields. Thus, a first selection of the detection methodologies and features reviewed in the last chapter has been done mostly based on two factors, the relevance of the methodology to the project and the available time for the completion of the project. For example, previously it was shown that the wavelet approaches require a thorough study of the basis functions and relevant scales suitable for the application. Hence, such a study would not be suitable for the time and scope of this project.

Thus, based on the literature survey done in the previous chapter, the theory behind the feature extraction, classification strategies and methodologies used in this project are now described. Next chapter will cover the specifics of each of the detectors implemented as well as the used methodology.

A general description of the detector structure proposed in this project is shown in figure 3.1. Firstly, the input data, in which transient events are to be detected, is processed to obtain relevant features. Then, the detection block applies a detection strategy to define if any events are found or not in the signal. Depending on the detection strategy, the detector might need a training step in order to

function properly. More details of the training step are provided later.

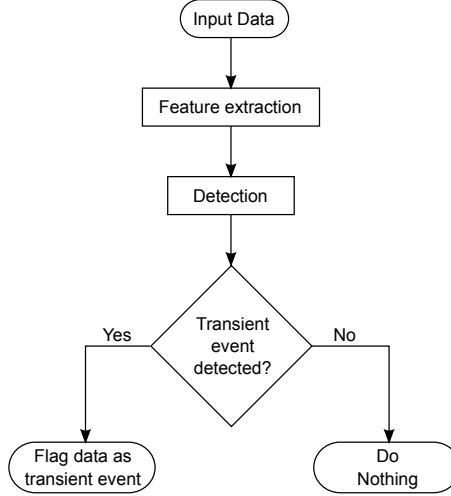


Figure 3.1: Flow chart of a detector's internal process.

3.1 Feature extraction

Features are the main quantities in which a detector bases its decision of flagging a data sequence as a transient event or not. Hence its importance. In this section, the relevant features used among the literature and in this work, are described.

3.1.1 Root Mean Square filtering

Root Mean Square (RMS) filtering consists in reducing a τ samples window to its RMS value. This process is applied to a whole signal, dividing it in consecutive windows, obtaining an RMS filtered signal. Thus, some time resolution and frequency information is lost, but the energy of the signal is not changed [Pon05]. Equation 3.1 shows the RMS filter equation where N is the number of samples in a discrete signal x .

$$x_{RMS}[m] = \sqrt{\frac{1}{\tau} \sum_{n=(m-1)\tau+1}^{m\tau} x^2[n]} \quad , m = 1, 2, \dots, \lfloor \frac{N}{\tau} \rfloor \quad (3.1)$$

3.1.2 Filtering

In the pattern recognition field, filters are used to remove low frequency noise or to analyse a signal in frequency bands. In acoustics, the most common filter banks are octave band and one-third octave band filters [JPR⁺11]. The standard centre frequencies for each filter band are shown in table 3.1.

20	25	31.5	40	50	63	80	100	125
160	200	250	315	400	500	630	800	1000
1250	1600	2000	2500	3150	4000	5000	6300	8000
10000	12500	16000	20000					

Table 3.1: One-third octave and octave band (bold) center frequencies (Hz).

For an octave band the upper and lower frequencies can be calculated with equations 3.2 and 3.3, respectively, where f_c is the centre frequency of the band.

$$f_u = f_c \cdot 2^{1/2} \quad (3.2)$$

$$f_l = f_c / 2^{1/2} \quad (3.3)$$

In a similar way, for an one-third octave band the upper and lower frequencies can be calculated with equations 3.4 and 3.5, respectively, where f_c is the centre frequency of the band.

$$f_u = f_c \cdot 2^{1/6} \quad (3.4)$$

$$f_l = f_c / 2^{1/6} \quad (3.5)$$

3.1.3 Short-Time Fourier Transform

The Fourier Transform cannot be used to analyse non-stationary signals as it lacks time localization. The Short-Time Fourier Transform (STFT) [Sta05] compensates this disadvantage by obtaining the Fourier Transform of small portions

of a signal where stationarity is assumed [JVB⁺00]. To achieve this, the STFT uses a window function shifted to the desired time in a signal. The STFT in discrete time is showed in equation 3.6, where x is the discrete signal and w the window function.

$$X(m, \omega) = \sum_{n=-\infty}^{\infty} x[n] w[n-m] e^{-j\omega n} \quad (3.6)$$

3.1.4 Non-Negative Matrix Factorization

As already mentioned in the previous chapter, Non-Negative Matrix Factorization (NMF) decomposes, in an approximated manner, a non-negative matrix V into the product of two non-negative matrices W and H , that is,

$$V \approx WH \quad (3.7)$$

The size of V is m by n , the way to interpret this is, n column vectors of m elements. The basis matrix W is of size m by r , where each column r represents a basis component. H is an r by n matrix, where each row r contains the activation coefficient of each basis component in W . So, each column vector in V is approximated by a linear combination of each basis component of W and the activation coefficients in H . "Therefore W can be regarded as containing a basis that is optimized for the linear approximation of the data in V " [LS01]. Thus, a very important aspect, core of the usage of NMF in this project, is that, a vector containing sample data can be projected into a given W . The resulting values of H from such projection can now be used as features. But previously, the basis components matrix W has to be calculated.

There are three different kinds of algorithms to achieve a NMF, namely, multiplicative update algorithms, gradient descent algorithms and alternating least squares algorithms [BBL⁺07]. The multiplicative update algorithm used in this project is shown in equations 3.8 and 3.8.

$$H = H \otimes \frac{W^T \cdot V}{W^T W H} \quad (3.8)$$

$$W = W \otimes \frac{V \cdot H^T}{W H H^T} \quad (3.9)$$

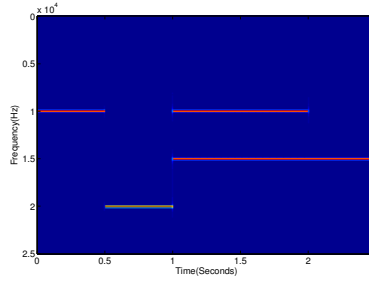
where \otimes is an element-wise multiplication, the division is also element-wise.

For this algorithm, matrices W and H are initialized randomly. Then, equation 3.8 is applied obtaining a new value of H . This new value of H is then used to calculate W from equation 3.9. The new value of W is used to improve H again, and so on. This process continues until an accepted measure between V and WH is reached or a defined number of iterations is exceeded.

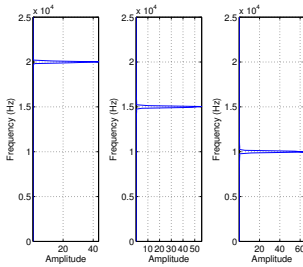
To evaluate the quality of the approximation there are two basic measures, the square of the Euclidean distance between V and WH (equation 3.10) and the divergence D of V from WH (equation 3.11) [LS01].

$$\|V - WH\|^2 = \sum_{mn} (V_{mn} - (WH)_{mn})^2 \quad (3.10)$$

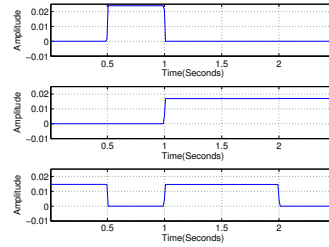
$$D(V \| WH) = \|V \otimes \log \frac{V}{WH} - V + WH\| \quad (3.11)$$



(a) Matrix V obtained from the spectrogram of a signal formed by 3 sine waves.



(b) Columns of matrix W .



(c) Rows of matrix H .

Figure 3.2: Example matrices V , W and H .

To better understand the concept of NMF, consider the following example. Figure 3.2a shows matrix V . It represents the magnitude of a spectrogram obtained from the following signal:

$$f(t) = \begin{cases} \sin(2\pi 10000t) & , 0 < t \leq 0.5 \\ \sin(2\pi 20000t) & , 0.5 < t \leq 1 \\ \sin(2\pi 10000t) + \sin(2\pi 15000t) & , 1 < t \leq 2 \\ \sin(2\pi 15000t) & , 2 < t \leq 2.5 \end{cases}$$

After performing NMF for 3 basis components, matrices W and H , shown in figures 3.2b and 3.2c, are obtained. As already mentioned, each row in W represents a basis component. Note how NMF is able to identify that the original signal is conformed by 3 frequency components, creating 1 basis component for each frequency. Then, matrix H indicates when each of the basis components becomes active.

3.1.5 Teager Energy Operator

As mentioned in the last chapter, TEO is a non-linear high pass filter that enhances high frequency components in transient signals and suppresses the low frequency background [SYT11]. Moreover, "*TEO is able to follow the instantaneous energy of the signal*" [SYT11]. The transient events to detect in this project are impulsive in nature, thus, the increase of energy in the signal is instantaneous. This makes TEO a suitable feature to enhance the presence of any impulsive event, diminishing the effects of low frequency background. TEO is defined in equation 3.12 where x is a discrete signal.

$$TEO[n] = x^2[n] - x[n-1]x[n+1] \quad (3.12)$$

3.1.6 Short Term Energy

Short Term Energy (STE) is another feature based on the energy of the signal, where the average of energy is calculated in a windowed manner. The average energy in a window containing any transient events is higher than the average energy in a window without any transient events. Thus, STE might give a good indication of the presence of events in a window due to instantaneous energy increase in a window. Equation 3.13 shows how the STE is calculated where x is a windowed signal of length M and k is a small constant to avoid $\log_{10}(0)$.

$$STE = \log_{10} \left(\frac{1}{M} \sum_{n=1}^M x^2[n] + k \right) \quad (3.13)$$

3.1.7 Coefficient of variation

The coefficient of variation (CV) is defined by the ratio

$$CV = \frac{\sigma}{\mu} \quad (3.14)$$

where σ and μ are the standard deviation and the mean of a random variable, respectively. It provides a dimensionless measure of the variability of data with respect to its mean [KOR64]. However, in [MJG10], the standard deviation and the mean of the absolute value of a signal are used to obtain the feature CV. When transient events are present in the signal, the mean of the absolute value of the signal is expected to increase as well as the standard deviation of the absolute value of the signal. Thus, CV provides a meaningful way to relate both increases in standard deviation and mean and compare that to the same ratio in other segments of the signal with no transient events.

3.1.8 Mel Frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficients (MFCCs) [MBE10] have been mostly used in speech recognition [MBE10], however it has also been used to characterize music and other environmental sounds [L+00][NPF11]. MFCCs are based on perceptual characteristics of human hearing such as the non-linear perception of pitch and approximately logarithmic loudness perception [L+00]. Given these characteristics of human hearing, human beings are good at detecting impulsive transient events in train track noise. Hence, the interest to investigate if MFCCs can also be used in transient event detection applied to train track noise. The implementation of MFCCs used in this project is the one described in [MBE10]. A brief description of the process to obtain MFCCs is now mentioned.

The calculation of MFCCs is done in 6 steps:

Step 1: Emphasize the presence of high frequencies by applying a filter.

Step 2: Segmentation of the signal into windows.

Step 3: Multiplication with a Hamming window to remove edge effects.

Step 4: The windowed signal is transformed to the frequency domain using the Fast Fourier Transform (FFT).

Step 5: Calculation of a weighted sum of spectral components, according to a Mel scale filter bank, to obtain a Mel spectrum. That is, the output of each

Mel scale filter is the weighted sum of the spectral components in the FFT being filtered by it, in a logarithmic scale like dB SPL. Each Mel scale filter's magnitude frequency response has a triangular shape. Below 1 kHz the center frequency of the filters follows a linear scale, while above 1 kHz it follows a log scale. This models human pitch perception [L⁺00]. Hence, the usefulness of MFCCs for perception related tasks.

Step 6: Application of the Discrete Cosine Transform to the Mel spectrum.

3.1.9 Maximum

Another recurrent feature among the literature is the maximum of the signal or the maximum of the absolute value of the signal [NPF11] [dIRML07] [ASSS07] [Meh08] [BZ08] [MJG10]. Impulsive transient events introduce instantaneous energy in the signal, thus causing high amplitude peaks in the signal. Hence, the maximum of the absolute value of a signal can be used as a feature to characterize transient events.

3.1.10 New Features

Two new features have been developed for this project. They are based on the energy of the signal in the time domain and the energy of the signal in the frequency domain, respectively. The proposed features are applied to the signal in figure 3.3 to demonstrate their characteristics. This sample signal is taken from a train track noise measurement.

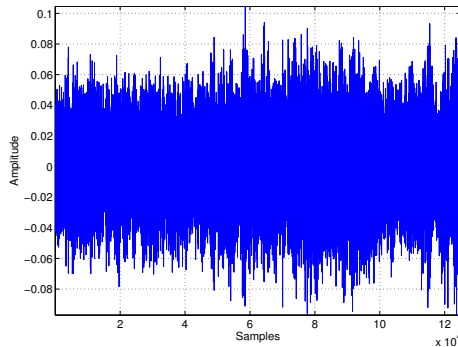


Figure 3.3: 2.5 seconds sample signal taken from a train track noise measurement. Sampling frequency 50000 Hz.

3.1.10.1 Change of energy

Each time instant an increase of energy is observed. During normal circumstances the increase of energy is constant, but when a transient event is present, a high increase of energy is observed. Figure 3.4 shows the energy of a signal as a function of time. Note that when there is an increase of energy in the signal there is an increase of the slope of the energy curve. Thus, to calculate the slope or change of energy, the concept of derivative can be used. Moreover, the change of energy at different time instants can be also introduced.

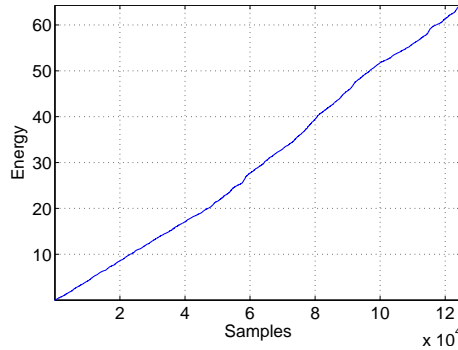


Figure 3.4: Energy of a signal as a function of samples.

The first step to compute this feature is to obtain the energy of the signal as a function of time, or in this case, as a function of samples, i.e. [JD96],

$$E[n] = \sum_{i=1}^n x^2[i] \quad (3.15)$$

where x is a discrete signal.

Then, the change of energy every d samples is calculated as

$$dE[n] = E[n] - E[n-d] \quad (3.16)$$

where $0 < d < N$, N is the number of samples in x .

Note that this can be interpreted as calculating the energy only in the past d

samples. That is,

$$\begin{aligned}
 dE[n] &= E[n] - E[n-d] \\
 &= \sum_{i=1}^n x^2[i] - \sum_{i=1}^{n-d} x^2[i] \\
 &= \sum_{i=1}^{n-d} x^2[i] + \sum_{i=n-d+1}^n x^2[i] - \sum_{i=1}^{n-d} x^2[i] \\
 &= \sum_{i=n-d+1}^n x^2[i]
 \end{aligned} \tag{3.17}$$

An example of this feature, obtained from the signal shown in figure 3.3, is shown in figure 3.5.

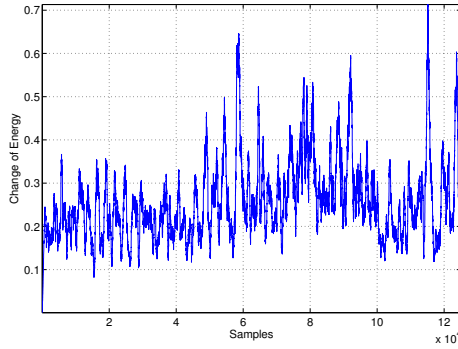


Figure 3.5: Example of the change of energy feature obtained from a signal. $d = 500$.

3.1.10.2 Logarithmic Power Spectrum

As already mentioned in the theory, the presence of transient events impose instantaneous components in the signal [GE05]. This could be observed in the increase of energy across the whole frequency range of the signal where transients are present. However, the increase of energy might be very small at some frequencies, hence, the logarithm can be used to emphasize low energy increments and attenuate high energy increments. The STFT, calculated in a sequential manner, produces a spectrogram. Thus, the logarithmic magnitude of the power spectrum S is calculated from the spectrogram at every instant of time considered. An example of S , obtained from the signal in figure 3.3, is shown in figure 3.6.

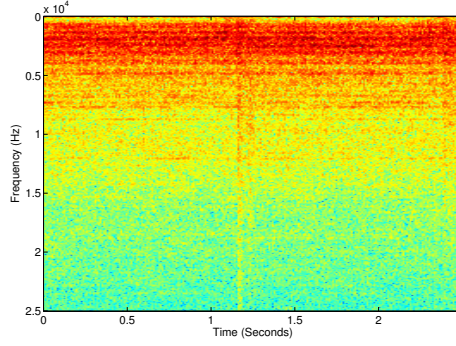


Figure 3.6: Spectrogram of a 2.5 seconds signal taken from a train track noise measurement.

The last step to obtain the feature is to sum the energy in each frequency bin for a given time instant. Equation 3.18 represents this last step.

$$LPS[n] = \sum_m S_{m,n} \quad (3.18)$$

where m represents the number of frequency points obtained from the calculation of the STFT and n is the number of STFT sequences in which a signal is split.

An example of this feature, obtained from the signal shown in figure 3.3, is shown in figure 3.7.

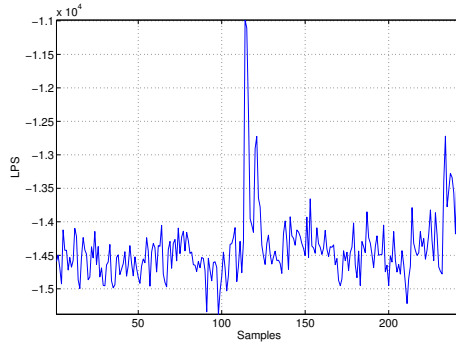


Figure 3.7: Example of the LPS feature obtained from a signal.

3.2 Event Detection

Two different strategies to event detection have been used in this work. The first approach belongs to the novelty detection methodology, where the normal behaviour of the system is modelled. The choice of a novelty detection approach is motivated by the large amount of available data without any events, making viable the possibility of building a model representing the normal behaviour of the system. The second approach is simply applying a threshold to a feature obtained from test data. When the value of a certain feature exceeds a threshold, then the test sample is classified as an event. Both approaches are now described.

3.2.1 Novelty Detection

As described in the previous chapter, novelty detection has been divided into two different categories, statistical and neural network approaches. Some statistical approaches are considered in this project as well as one neural network approach.

3.2.1.1 Statistical

One of the most used common models used to describe data statistically is the Normal or Gaussian distribution [Das10]. The univariate version of the Normal density function is shown in equation 3.19, where μ is the mean and σ is the standard deviation.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, -\infty < x < \infty \quad (3.19)$$

To fit this model to data only the data's μ and σ are needed. Moreover, the Kolmogorov-Smirnov test of goodness of fit [MJ51] can be used to verify if the data truly has normal distribution. The Kolmogorov-Smirnov test is based on a distance between a hypothetical cumulative density function and the cumulative density function of the data. If this distance exceeds a certain level of significance then there is evidence that the data does not belong to the hypothesized distribution.

However, a multivariate normal distribution is more often needed to model the joint distribution of more than one variable. The n -variate normal distribution density function [HS11] is shown in equation 3.20, where \mathbf{x} is an n -dimensions

variable, μ is an n -dimensions vector with the mean values for each dimension in \mathbf{x} . Σ is an n by n covariance matrix of the form

$$\begin{bmatrix} \sigma_1^2 & \rho_{1,2}\sigma_1\sigma_2 & \cdots & \rho_{1,n}\sigma_1\sigma_n \\ \rho_{2,1}\sigma_2\sigma_1 & \sigma_2^2 & \cdots & \rho_{2,n}\sigma_2\sigma_n \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n-1,1}\sigma_{n-1}\sigma_1 & \rho_{n-1,2}\sigma_{n-1}\sigma_2 & \cdots & \rho_{n-1,n}\sigma_{n-1}\sigma_n \\ \rho_{n,1}\sigma_n\sigma_1 & \rho_{n,2}\sigma_n\sigma_2 & \cdots & \sigma_n^2 \end{bmatrix}$$

where σ_n is the standard deviation of the n -th dimension of variable \mathbf{x} and $\rho_{n-1,n}$ is the correlation between the data in the $n - 1$ -th dimension and the n -th dimension of variable \mathbf{x} .

$$f(\mathbf{x}) = (2\pi)^{-\frac{n}{2}} |\Sigma|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{x}-\mu)'\Sigma^{-1}(\mathbf{x}-\mu)} \quad (3.20)$$

However, some distributions of data need a more flexible model when data densities are not necessarily concentrated together. Thus, the usage of GMMs is a common practice. The density function of a GMM is shown in equation 3.21 [WZ08], where $P(j)$ are the mixing coefficients and $p(\mathbf{x}|j)$ are the Gaussian density functions. Note that, as $\int p(\mathbf{x}|j) = 1$, $\sum P(j) = 1$. Figure 3.8 shows an example of a mixture of 3 Gaussians.

$$f(\mathbf{x}) = \sum_{j=1}^M P(j) p(\mathbf{x}|j) \quad (3.21)$$

The Expectation Maximization algorithm is commonly used to fit any of this multivariate distributions [WZ08]. Note that the amount of parameters to estimate by the Expectation Maximization algorithm can be reduced if the covariance matrices Σ are restricted to a diagonal form. More over, the Akaike's Information Criterion (AIC) [BA02] provides a means to evaluate how accurately the data is represented by the model. It is defined as shown in equation 3.22 where $\hat{\theta}$ are the parameters of the density function, y is the training data, \mathcal{L} is the likelihood function and K is the number of parameters in the density function. As AIC gives a measure of the relative distance between a fitted model and the unknown true mechanism that generated the training data [BA02], the model with the minimum AIC, among a set of models, is said to better describe the data.

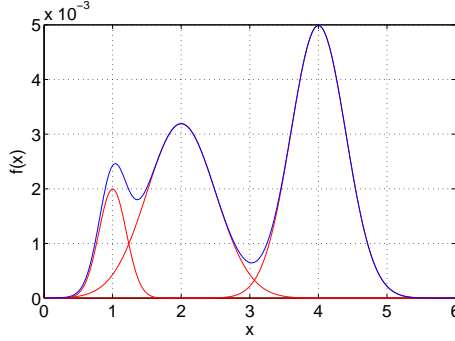


Figure 3.8: 3 Gaussians in \mathbb{R}^1 with $\mu = 1, 2, 4$ and $\sigma = 0.2, 0.5, 0.4$, respectively (red). GMM of the same Gaussians with mixing coefficients equal to 0.1, 0.4 and 0.5, respectively (blue).

$$AIC = -2\log\left(\mathcal{L}\left(\hat{\theta} \mid y\right)\right) + 2K \quad (3.22)$$

A different approach can be used to generate a model with very few parameters, namely, Parzen-Window density estimation or kernel density estimation (KDE) [Sil86]. If $f(\mathbf{x})$ is the density function to estimate, and \mathbf{x}_n is a set of n independent and identically distributed random variables, the Parzen-Window density estimate is given by

$$\hat{f}(\mathbf{x}) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{\mathbf{x} - \mathbf{x}_n}{h}\right) \quad (3.23)$$

where K is a kernel function that satisfies $\int_{-\infty}^{\infty} K(x) dx = 1$ and h is the width of the kernel. The kernel function is typically Gaussian [YC02][Sil86] as it provides a means to find the optimal width of the kernel [Sil86]. For univariate density estimation using Gaussian kernel functions the optimal width h_{opt} is

$$h_{opt} = \left(\frac{4}{3n}\right)^{\frac{1}{5}} \sigma \quad (3.24)$$

where n and σ are the number of samples and standard deviation of the data, respectively. A way of finding the optimal width h in multivariate densities is covered in [Sil86].

Once the model of normal data is obtained the event detection can be performed choosing an adequate threshold k . Every incoming test data point \mathbf{x}_t is evaluated in the density function of the model. If $f(\mathbf{x}_t) \geq k$ then the data point is classified as normal. If $f(\mathbf{x}_t) < k$ then the data point is classified as 'event'.

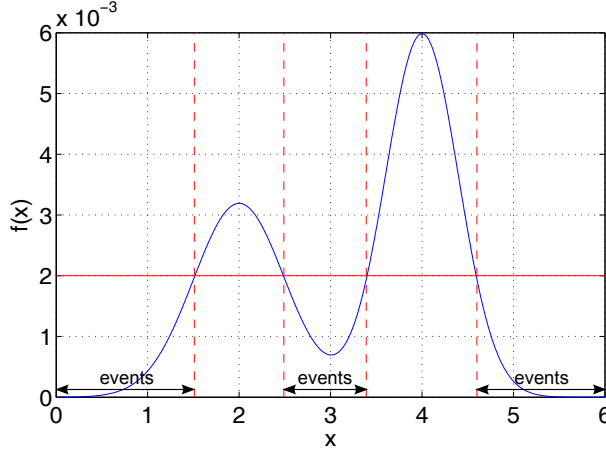


Figure 3.9: GMM of two Gaussians in \mathbb{R}^1 with $\mu = 2, 4$ and $\sigma = 0.5, 0.4$, respectively. Mixing coefficients equal to 0.4 and 0.6, respectively. $k = 0.002$.

For the example in figure 3.9, all \mathbf{x}_t with $f(\mathbf{x}_t) \geq 0.002$ are classified as normal and those with $f(\mathbf{x}_t) < 0.002$ are detected as events.

3.2.1.2 Neural Networks

The only neural network studied in this project is the so called auto-encoder [JMG95]. This neural network tries to replicate the input signal in its output, but this can only be achieved if the input is similar to the data used to train the network. Hence, the network has to be trained with data related to the normal behaviour of the system. When the network is tested with a transient event, it should not be able to replicate this input as it has not been trained with it, indicating the presence of a transient event. The auto-encoder neural network is composed of a certain number of input neurons, this quantity depends on the dimensions of the input variable. For example, if the input variable was taken from a 5 bands filtered signal, 5 input neurons could be used. The number of output neurons is the same as the number of input neurons, and there is a hidden layer with less neurons than the input or output. According to [JMG95], the reduction of neurons in the hidden layer forces the network to compress any redundancies in the data, retaining non-redundant information. Thus, the auto-encoder performs dimensionality reduction [Bel06]. The topology of an auto-encoder is shown in figure 3.10.

As already mentioned, the main purpose of the auto-encoder is to replicate the

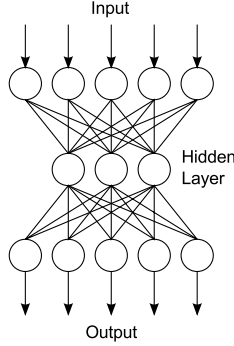


Figure 3.10: Auto-encoder with 5 input neurons, 5 output neurons and 3 neurons in the hidden layer.

input in its output. Thus, the quality of the reconstruction is evaluated as shown by equation 3.25, where I_i is the input of the i -th input neuron and O_i is the output of the i -th output neuron.

$$Error = \sum_i |I_i - O_i| \quad (3.25)$$

After training using the back-propagation algorithm [Bel06], the detection criteria is based on the error. Finally, if $Error \leq k$, no event is detected, if $Error > k$, an event is detected.

3.2.2 Simple Threshold Detection

The simple threshold detection methodology is directly based on a feature extracted from test data, thus, no models are required. If there exists a value of the extracted feature that exceeds a certain k , then the test data sample is detected as an event. Otherwise, it is considered as normal data. More over, to account for different levels of background noise, the extracted feature is normalized to its RMS value. Hence, a windowed approach is encouraged for the application of this detection method.

When the feature is obtained in frequency bands, two different criterion are applied to decide if the test sample contains an event or not, that is, an AND or OR criteria. To explain better this idea, figure 3.11 can be used. In the example in figure 3.11 the input data has been filtered in 3 frequency bands. Then, per

each frequency band, the TEO feature has been extracted and normalized to its respective RMS value. The 3 signals shown in figure 3.11 represent these normalized feature values. Now, the comparison to threshold k is done. Note that another benefit of the RMS normalization is that all frequency bands can now be compared to the same threshold k . The first frequency band contains no events as no feature value exceeds the threshold. However, for the second and third frequency band, there are feature values that exceed the threshold. So, the AND and OR criterion define when to decide that there is an event of the signal depending on the number of frequency bands that exceed the threshold. The AND criteria states that all frequency bands have to exceed the threshold to define that there is an event in that signal. The OR criteria states that if only one frequency band exceeds the threshold then there is enough evidence that a event is happening in that signal.

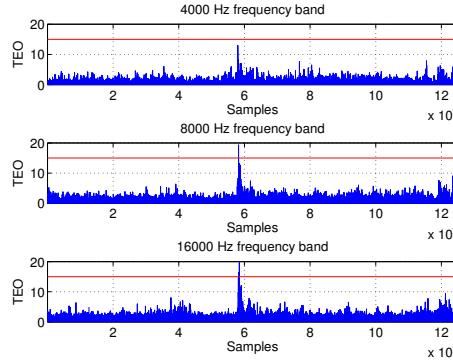


Figure 3.11: TEO feature obtained from the filtered signal in figure 3.3. $k = 15$.

Methodology

The proposed methodology is now described, taking into account the features and detection strategies described in the last chapter.

Figure 4.1 shows, in a schematic way, the proposed methodology to analyse the performance of the promising detectors designed during this project. By promising detectors is understood those detectors that perform well during the first of two steps that conform the proposed methodology.

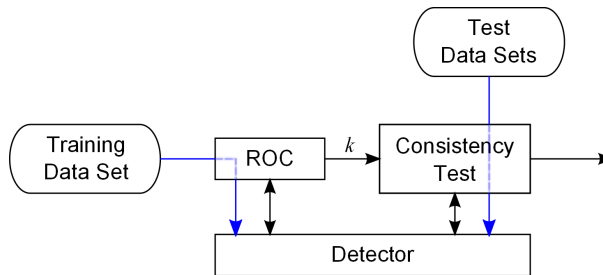


Figure 4.1: Schematic of the methodology proposed to evaluate different detection techniques.

The first step makes use of a training data set to generate a Receiver Operating Characteristic (ROC) [Faw04] curve for each detector. In our context, the ROC

curve shows relative trade-offs, as a function of threshold k , between correctly identified data samples containing events (true positives) and data samples not containing events but marked as 'event' (false positives). Having obtained the ROC curve, a threshold k is selected based on the admissible true positives-false positives relationship. This step will be better understood in the ROC section, what is important to consider for now is that the ROC curve provides a means to select promising detectors and a threshold k that meets the true positive-false positive trade-off requirement under training data. Figure 4.2 shows an example of the ROC curve. Each point ($FalsePositiveRatio, TruePositiveRatio$) is obtained by applying a detection strategy for a threshold k .

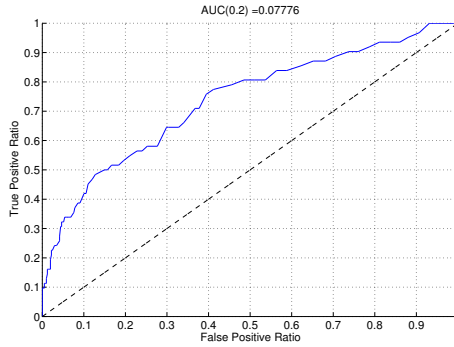


Figure 4.2: Example ROC curve.

The second step evaluates the performance of the detector for the selected k based on the consistency of its results in different test data sets. The proposed performance measures will be explained later in this chapter.

The whole detection and one-class classification framework (figure 4.1) was implemented in Matlab. A description of each of the blocks that conform the framework now follows, where the different training sets are described in the train track noise section 4.4.

4.1 Detector

The general behaviour of a detector is to receive an input signal, extract a specific set of features and apply a detection strategy to give a result on the localization of any transient events in the input signal. The detector yields an accuracy in the localization of an event in terms of short segments of the input signals (windows), that is, the analysis to search for transients is done window by window. Note that no window overlapping is used in this implementation. Even

if there is a trade-off between window size and localization accuracy, the length of the window has been fixed in this project due to the large number of other important tunable factors that affect the detection performance. Localization accuracy means that, when a window has been flagged as containing an event, there is no accurate data of the exact moment in which the event starts and ends, it is just known that an event happens along the window. The selected window length w , during the whole experimentation phase, has been set to 0.5 seconds. In samples, this length is $0.5 \cdot F_s$, where F_s is the sampling frequency of the input signal. This number has been chosen based on the observation of the duration of transient events in train track noise, that is, no transient event plus a clearance exceeds a duration of 0.5 seconds. Also, this value provides an acceptable localization accuracy of events for the purposes of the project. Finally, the output of the detector is a binary sequence where each sample n corresponds to the n -th window of the input signal. Zero values indicate no events detected and one values indicate events detected. Figure 4.3 shows an example of the output of the detector block. Note how each 2 windows in the detector output (lower figure) correspond to 1 second of the signal.

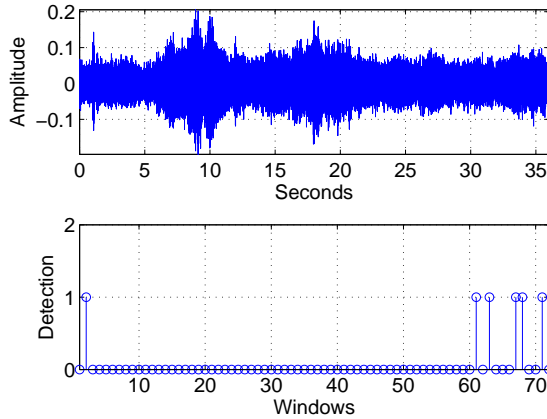


Figure 4.3: Sample signal (upper) and detector output (below).

The implemented detectors can be divided, depending on its detection strategy, in novelty detectors and simple threshold detectors. The specifics of its implementation is now covered.

4.1.1 Novelty detectors

As this kind of detectors makes use of a normal-behaviour-data model, the model and the input test data has to be normalized to make them comparable. Note that each detector works with a specific model, hence the feature extraction process to build the model and to perform detection is the same. Thus, the features have to be extracted from input data at same time intervals and with the same scale. The time interval issue is solved by splitting the test data into windows of length w . The test data is normalized to its Z-score [Kal11] to account for scaling issues. The Z-score is defined in equation 4.1 and it tells how many standard deviations a sample $x_w[n]$ is away from the window mean.

$$Z[n] = \frac{x_w[n] - \mu}{\sigma} \quad (4.1)$$

where μ and σ are the mean and standard deviation of a windowed signal x_w .

Several novelty detectors were designed, its characteristics are summarized in tables 4.1 and 4.2. The following subsections describe the obtention of the data model and the specifics of each detector.

4.1.1.1 Training data model

Novelty detectors make use of a model to perform detection, thus, a model is first generated from training data removing any transient events in it. This approach needs an a priori knowledge of the location of the events in the signal. The way to eliminate any transient events from the training data is by removing the window of length w where events happen, as well as the past and next consecutive windows surrounding the event window.

The general procedure to generate the data model consists of normalizing the transient-free training data as previously described, feature extraction and, in the case of statistical novelty detectors, fitting of the feature data set to a model. In the case of the neural network detector, the network is trained with the feature data set. This training could be seen as, the neural network 'learning' the model of the data.

For the *STEuni* detector, the model is obtained using the STE feature. One STE value is obtained per window. As it will be shown later in the results chapter, the density distribution of this feature data set is unimodal, but not exactly normal. However, the feature data set was fit to an unimodal Gaussian distribution. For

Table 4.1: List of novelty detectors, features and models.

Name	Features	Model
<i>STEuni</i>	STE	Unimodal Gaussian
<i>CVuni</i>	CV	Unimodal Gaussian
<i>MAXuni</i>	Maximum	Unimodal Gaussian
<i>STEparz</i>	STE	Parzen window estimate
<i>CVparz</i>	CV	Parzen window estimate
<i>MAXparz</i>	Maximum	Parzen window estimate
<i>STECVMAXgmm</i>	STE CV Maximum	GMM
<i>MFCCgmm</i>	12 MFCCs	GMM
<i>FREQNMFgmm</i>	one octave band filtering RMS NMF	GMM
<i>TIMENMFgmm</i>	RMS filtering NMF	GMM
<i>FREQnn</i>	one octave band filtering RMS	Neural Network
<i>TIMEnn</i>	RMS filtering	Neural Network

Table 4.2: List of novelty detectors and chosen parameters.

Name	Parameters
<i>STEuni</i>	N/A
<i>CVuni</i>	N/A
<i>MAXuni</i>	N/A
<i>STEparz</i>	N/A
<i>CVparz</i>	N/A
<i>MAXparz</i>	N/A
<i>STECVMAXgmm</i>	gmm components: 1, AIC
<i>MFCCgmm</i>	GMM components: 1, AIC
<i>FREQNMFgmm</i>	GMM components: 1, AIC NMF basis components: 3, 4
<i>TIMENMFgmm</i>	GMM components: 1, AIC NMF basis components: 3, 12
<i>FREQnn</i>	hidden neurons: 3,4
<i>TIMEnn</i>	hidden neurons: 3, 12

this detector there are no parameters to select manually. The same description fits for the *CVuni* and *MAXuni* detectors. Figure 4.4 exemplifies this process.

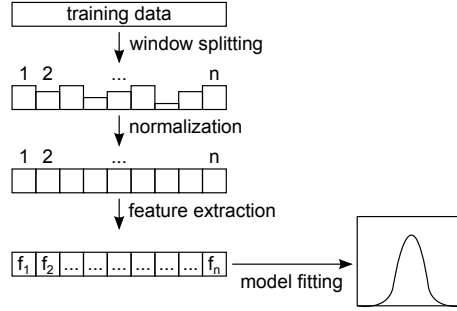


Figure 4.4: Process for univariate model fitting.

The models of the *STEparz*, *CVparz* and *MAXparz* detectors are obtained using Parzen window estimates. The normalization and feature extraction process is the same as with the *STEuni*, *CVuni* and *MAXuni* detectors. Width h of the Gaussian kernels used is calculated using equation 3.24. These univariate detectors have been designed to investigate if the unimodal Gaussian distribution is enough to describe the transient-free training data or a more 'descriptive' model of the training data is needed.

The features used by the model of the *STECVMAXgmm* detector are STE, CV and Maximum. As this model is obtained with a GMM, the only parameter to set is the number of mixture components. Hence, two different values are investigated. That is, the components suggested by AIC and 1 component. The process is shown in figure 4.5.

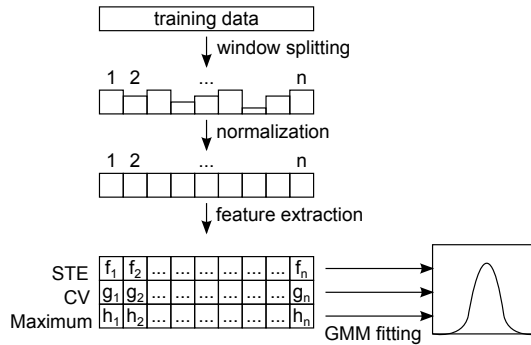


Figure 4.5: Process for STE, CV and Maximum features, GMM fitting.

The *MFCCgmm* detector uses as features 12 MFCCs taken from a frequency range of 1000 – 25000 Hz. The model is a 12 dimensions GMM. Once more, the only parameter to set is the number of mixture components. Thus, the number of components suggested by AIC and 1 component are investigated. Figure 4.5 applies for this model, the only difference is that 12 MFCCs are obtained as feature for each data window so the model is a 12 dimensional GMM.

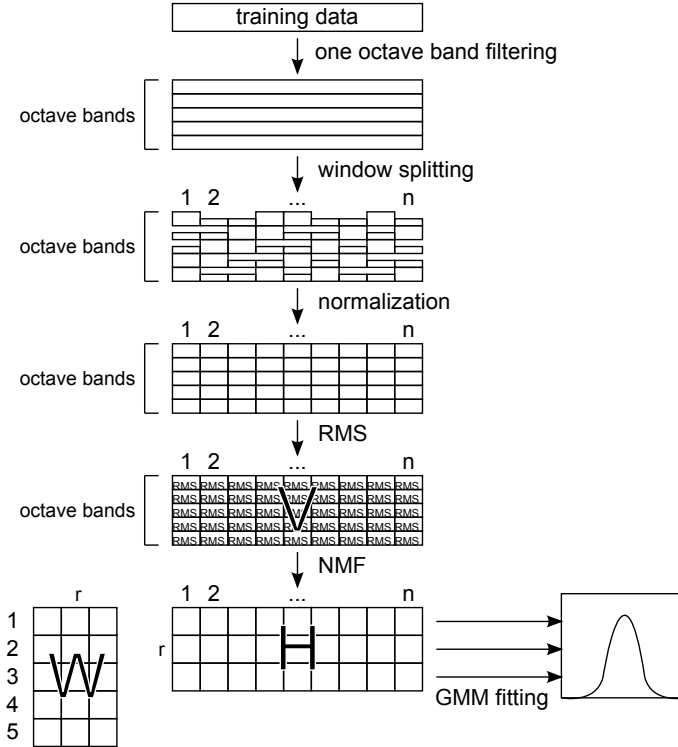


Figure 4.6: Process for RMS filtered training data features and NMF, GMM fitting.

The model for the *FREQNMFgmm* detector takes into account frequency information in frequency bands. The transient-free training data set is filtered using a one octave band filter from 1000 Hz to 25000 Hz. In total 5 frequency bands are obtained. Thus, 5 windows are obtained at each time interval. The RMS value is obtained from each window conforming a 5 dimensional column vector. Thus, the size of matrix V is 5 by n , where n is the number of windows in the transient-free training data set. As no suggestion of the number of basis components was found, the values tested are 3 and 4 basis components. A low number of basis components is a good starting point as it prevents any curse of dimensionality issues from happening. To apply NMF, matrices W and H are initialized randomly, the stopping condition is 1000 iterations or when the

Euclidean distance between V and WH (equation 3.10) is less than 10^{-5} . The same settings are used for all the NMFs in this work. After applying NMF to matrix V , each row of H is considered as a dimension of a random variable. Hence, a 3 and 4 dimensions GMM can be fitted. Once again, the number of components suggested by AIC and 1 component are investigated. Figure 4.6 shows this process. Matrix W is also kept as part of the model in order to project incoming data into it.

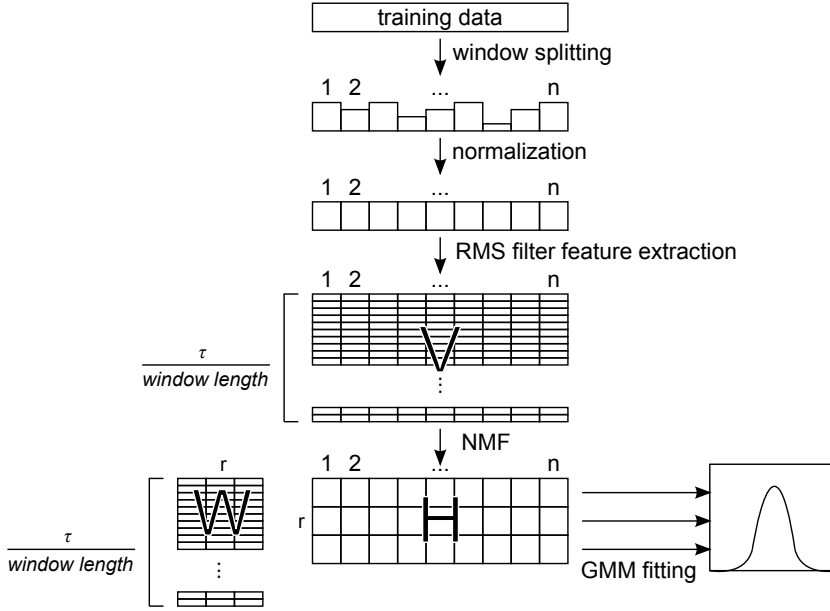


Figure 4.7: Process for RMS filtered transient-free training data features and NMF, GMM fitting.

The model for the *TIMENMFgmm* detector uses as features matrix H obtained from a NMF. Matrix V is obtained by applying an RMS filter to each window of the transient-free training data set. Thus, for a $\tau = 40$ samples, the size of V is 625 by n , where n is the number of windows of length w in the transient-free training data set. Once again, as no suggestion of basis components was found, the values tested were chosen arbitrarily to be 3 and 12 basis components, taking into account possible curse of dimensionality. Then, NMF is applied to V and the resulting H is used to form a 3 and 12 dimensional GMM. The number of components suggested by AIC and 1 component are investigated. Figure 4.7 shows this process. Matrix W is also kept as part of the model in order to project incoming data into it.

To compare the dimensionality reduction capacities of NMF and neural net-

works, neural network detectors using the same features as the NMF detectors are proposed. Thus, the model for the *FREQnn* detector takes into account frequency information. The transient-free training data set is filtered using a one octave band filter from 1000 Hz to 25000 Hz. In total 5 frequency bands are obtained. Thus, 5 windows are obtained at each time interval. The RMS value from each window is obtained, conforming the input to the neural network to train. Thus, the number of input and output neurons is 5. Figure 4.8 shows this process. Again, the number of hidden neurons is to be found experimentally. So 3 and 4 hidden neurons are used.

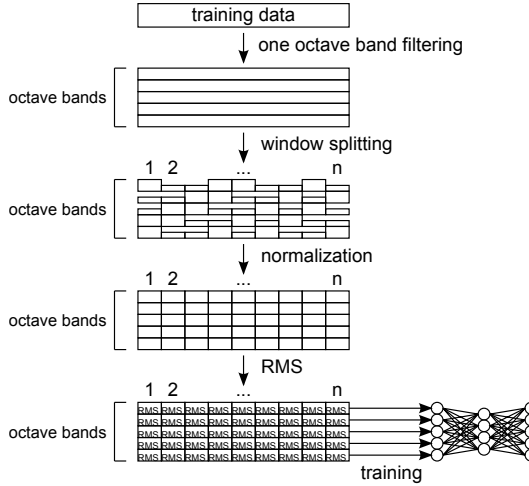


Figure 4.8: Process for one octave band filtered transient-free training data features, neural network training.

The model for the *TIMEnn* detector uses RMS filtering as feature. The τ value for the RMS filter is set to 40 samples for comparison reasons with the *TIMEN-MFgmm* detector. Note that for $\tau = 40$ samples, the maximum frequency in the signal after RMS filtering is 625 Hz. This may be understood as reducing the sampling frequency in a factor of 40, that is, a sampling frequency of 1250 Hz. Note, that even if the frequency information is lost, the energy of the signal is preserved. Thus, the number of RMS filtered values in a window of 25000 samples is 625. Thus, the number of input and output neurons is 625. In this case, the number of hidden neurons to test is 3 and 12. Figure 4.9 exemplifies this process.

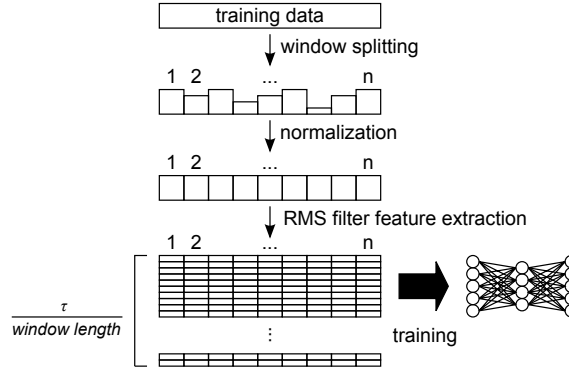


Figure 4.9: Process for RMS filtered transient-free training data features, neural network training.

4.1.1.2 Detectors description

There are no particularities in the detection strategy of the *STEuni*, *CVuni*, *MAXuni*, *STEpaz*, *CVparz*, *MAXparz*, *STECVMAXgmm*, *MFCCgmm*, *STECVMAXnn*, *FREQnn* and *TIMEnn* detectors. They follow the general approach, that is, the input data is split in windows of length w , each window is normalized to its Z-score. Then, the respective features are obtained from the normalized windows. Finally, these feature vectors are compared to the specific statistical model or, in the case of the neural network detectors, the feature vectors are fed to the neural network. Then the detection strategy for statistical models or neural network models described in the last chapter is applied. Figure 4.10 shows this description in a schematic manner.

In the case of the *FREQNMFgmm* and *TIMENMFgmm* detectors, the detection strategy is slightly different. First, the input data is split in windows of length w , each window is normalized to its Z-score. Then, the respective features are obtained from the normalized windows to form a column vector v' per window. Note that the features are extracted in the same manner as they are obtained to build the data model for the respective detector. Afterwards, each vector v' is projected into the respective matrix W , i.e., the solution in the least squares sense of the over-determined system of equations $v' = Wh'$ is found. In that way, the resulting h' vector can be compared to the statistical model and the detection strategy can be performed. This description can be seen in figure 4.11.

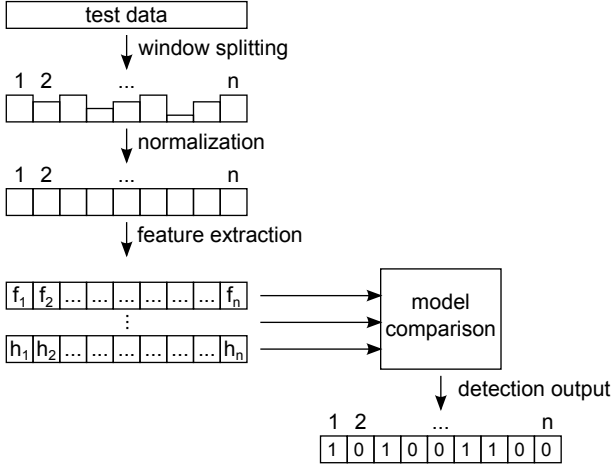


Figure 4.10: Detailed detection process. Novelty detectors.

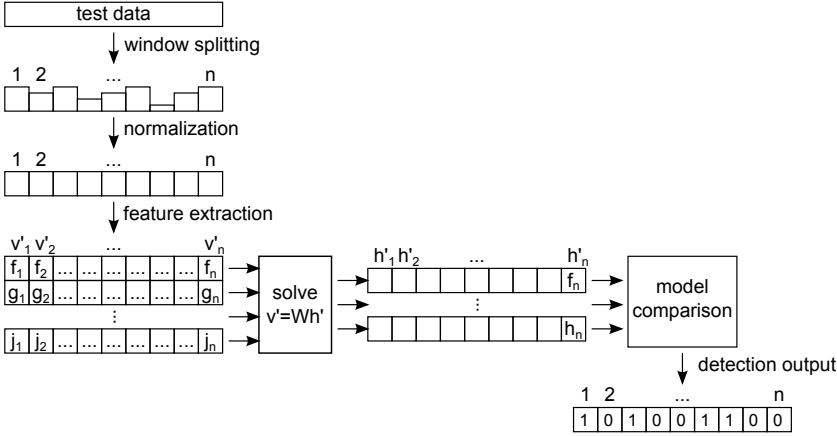


Figure 4.11: Detailed detection process for detectors that use NMF as feature. Novelty detectors.

4.1.2 Simple threshold detectors

Simple threshold detectors are much more simpler in the sense that they do not require a model obtained from transient-free training data. Table 4.3 shows the list of simple threshold detectors as well as the used features and the varied parameters, if applicable. For the *COE* and *LPS* detectors, the parameters d and $\text{STFT}_{\text{window}}$, respectively, could be related to the duration of transient events. Then, for both detectors, the best parameter value could be one that

fitted exactly the duration of a transient event as all the energy injected by the event would be considered by the window implicit in the feature. This would produce high values of the feature in this time instant. However, the duration of a transient is variable. Thus, the parameters to test are $d = 1, 512, 2048$ samples and $\text{STFT}_{\text{window}} = 512, 2048$. For the *TEOFREQ* and *COEFREQ* detectors, its AND and OR detection strategies are tested.

Table 4.3: List of simple threshold detectors, features and parameters.

Name	Features	Parameters
<i>TEO</i>	Teager Energy Operator	N/A
<i>COE</i>	Change of Energy	d : 1, 512, 2048
<i>LPS</i>	Logarithmic Power Spectrum	$\text{STFT}_{\text{window}}$: 512, 2048
<i>TEOFREQ</i>	one octave band filtering TEO	detection operator: AND, OR
<i>COEFREQ</i>	one octave band filtering COE	d : 1, 512, 2048 detection operator: AND, OR

The *TEO*, *COE* and *LPS* detectors split the test data in windows of length w , then, for each window, the specific feature is calculated. This result is normalized to its RMS value. The simple threshold detection strategy is then applied to each of the RMS normalized feature windows. Figure 4.12 shows a schematic that describes this process.

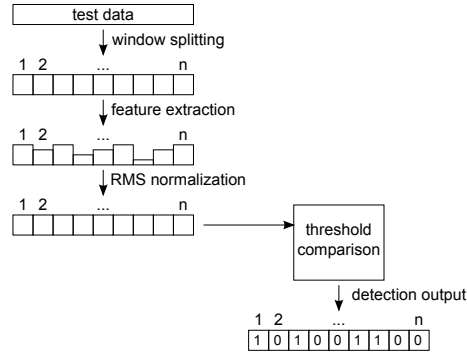


Figure 4.12: Detailed detection process for simple threshold detectors without frequency band analysis.

The *TEOFREQ* and *COEFREQ* detectors function in a very similar way to the other simple threshold detectors, but before the features are extracted, the test signal is filtered in one octave bands in a frequency range of 1000 – 25000 Hz.

Then, each frequency band is split into windows, the features are extracted from each window and finally each window is normalized by its RMS value. Figure 4.13 shows a schematic that describes this process.

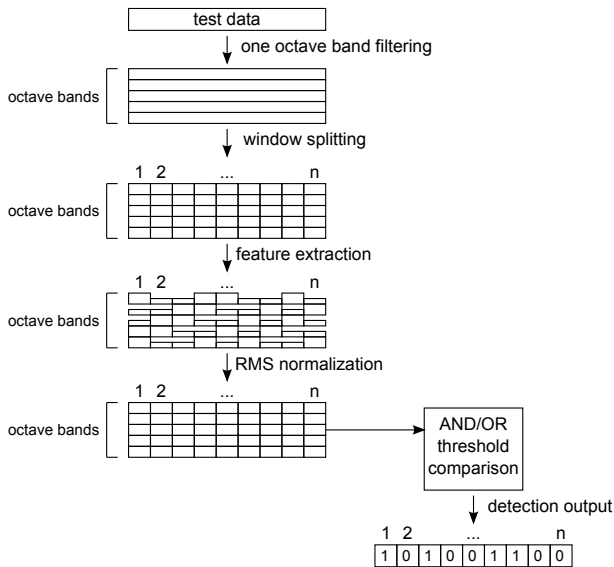


Figure 4.13: Detailed detection process for simple threshold detectors with frequency band analysis.

4.2 Receiver Operating Characteristic

As mentioned before, the Receiver Operating Characteristic provides a means to evaluate detector performance and threshold selection. This section provides a definition of ROC curve and its interpretation.

When performing detection on a test sample, there are 4 possible outcomes. If the sample contains an event and it is detected as an event, then it is called a *true positive (TP)*. If the sample does not contain an event and it is detected as an event, it is called a *false positive (FP)*. Moreover, if the sample does not contain an event and no events are detected, it is called a *true negative (TN)*. Finally, if the sample does not contain an event and an event is detected, it is called a *false negative (FN)*. These 4 outcomes can be summarized in a confusion matrix, shown in table 4.4.

Table 4.4: Confusion matrix. Redrawn from [Faw04].

		True event	
		p	n
Hypothesized event	Y	True Positives	False Positives
	N	False Negatives	True Negatives
Column totals:		P	N

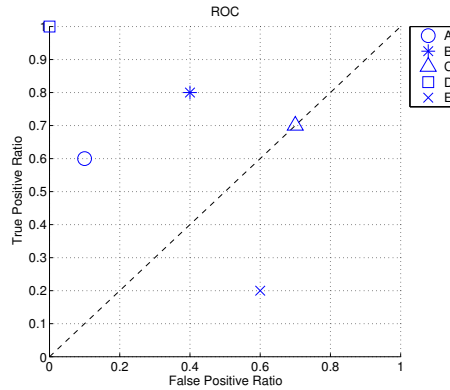
Several metrics can be obtained from the confusion matrix [Faw04], however, the ones used in this work are shown in equations 4.2 and 4.3.

$$FPR = \frac{FP}{N} \quad (4.2)$$

where N is the total number of negative samples, that is, samples containing no events. FPR stands for False Positive Ratio.

$$TPR = \frac{TP}{P} \quad (4.3)$$

where P is the total number of positive samples, that is, samples containing events. TPR stands for True Positive Ratio.

**Figure 4.14:** ROC space with 5 detectors at constant threshold. Redrawn from [Faw04].

An ROC is a curve showing the relation of TPR and FPR as a function of threshold k . An example of the ROC space is shown in figure 4.14, where 5 detectors at a fixed threshold are shown. Note that the diagonal line represents the strategy of deciding randomly if a test sample contains or not an event [Faw04]. An example of a detector in such situation is C. Hence, the diagonal is the main reference, where all detectors above this line perform better than random guessing (detectors A, B and D) and all the detectors below perform worse (detector E). Detectors appearing close to the TPR axis can be thought as "conservative" as they yield positives only when there is a strong evidence of it, obtaining a low FPR . From figure 4.14 it can be seen how A is more conservative than B. Detectors performing in the upper-right corner of the curve can be considered as "liberal" because they yield positive detections with little evidence finding in that way nearly all the events, but at the expense of producing many false positive results. The perfect detector is the one that find all positives without doing any false positives, like detector D.

An important characteristic of a ROC curve is the area under the curve (AUC). It allows to compare detectors, where a detector yielding a greater AUC performs better in average [Faw04]. The interest of this project is to find detectors with high performance, so the AUC for $FPR \leq 0.2$ is of mayor concern. If a perfect detector is considered, then the AUC for $FPR \leq 0.2$ would be 0.2. Thus, an AUC for $FPR \leq 0.2$, relative to an area of 0.2, is the first indicator of detector performance. This AUC will be denoted as $AUC(0.2)$ from now on. Note then, that the maximum $AUC(0.2)$ for a perfect detector is $AUC(0.2)=1$.

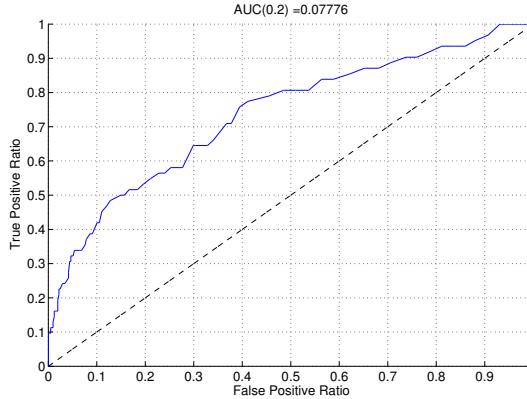


Figure 4.15: Example ROC curve.

Moreover, to observe the performance of a detector in the whole ROC space, the ROC curves extracted from each detector are obtained for $0 \leq FPR \leq 1$ and $0 \leq TPR \leq 1$. Figure 4.15 shows an example of a ROC curve where the detector yields an $AUC(0.2) = 0.07776$.

4.3 Consistency test

After obtaining an ROC for a specific detector, the next step in its evaluation is to choose a threshold k_{ref} for a specific reference (FPR_{ref}, TPR_{ref}) point in the ROC curve. Then, the idea is to detect transient events in a new data set using threshold k_{ref} . Thus, obtaining a new (FPR, TPR) point. The closest the new point to the reference point, the more consistent a detector performs on different data sets. However, it is also important to find out if the new (FPR, TPR) point moves towards an improvement of the performance of the detector in terms of its FPR and TPR values or not. In this project, the reference point is set arbitrarily as $(0.2, TPR_{ref})$.

As it has been described, the diagonal line between points $(0,0)$ and $(1,1)$ in ROC space indicates when a detector guesses randomly if an event is happening or not. Thus, it is natural to think a detector improves if it moves away from this diagonal line with respect to its reference point. Hence, part of the proposed measure takes into account the displacement of a new (FPR, TPR) point with respect to the reference point and the diagonal line.

Thus, the measure to evaluate the consistency of a detector is defined by the pair

$$score_d = (D, \cos(\varphi - 135)) \quad (4.4)$$

$$D = \sqrt{(FPR - FPR_{ref})^2 + (TPR - TPR_{ref})^2} \quad (4.5)$$

$$\varphi = \text{atan2}(TPR - TPR_{ref}, FPR - FPR_{ref}) \quad (4.6)$$

where D is the euclidean distance between (FPR_{ref}, TPR_{ref}) and (FPR, TPR) , ϕ is in degrees and atan2 is the 4 quadrant arctan. Note that a perfectly consistent detector generates a $score_d = (0, 0)$, a detector with $\cos(\varphi - 135) > 0$ is a detector that improves its performance in terms of its FPR and TPR values, and a detector with $\cos(\varphi - 135) < 0$ is a detector that degrades its performance in terms of its FPR and TPR values.

Figure 4.16 shows an example of two detectors. Note that, for this case, detector A has a reference of $(FPR_{ref}, TPR_{ref}) = (0.2, 0.5)$ and moves to $(0.1, 0.6)$ with other test data set. For detector B, the reference is $(FPR_{ref}, TPR_{ref}) = (0.2, 0.6)$, and moves to $(0.4, 0.5)$ with the same other test data set. Thus, following equation 4.4,

$$score_A = (0.14, \cos(135 - 135)) = (0.14, 1)$$

$$score_B = (0.22, \cos(-26.5 - 135)) = (0.39, -0.95)$$

In this sense, detector A is more consistent than B, and also improves its performance in terms of FPR and TPR .

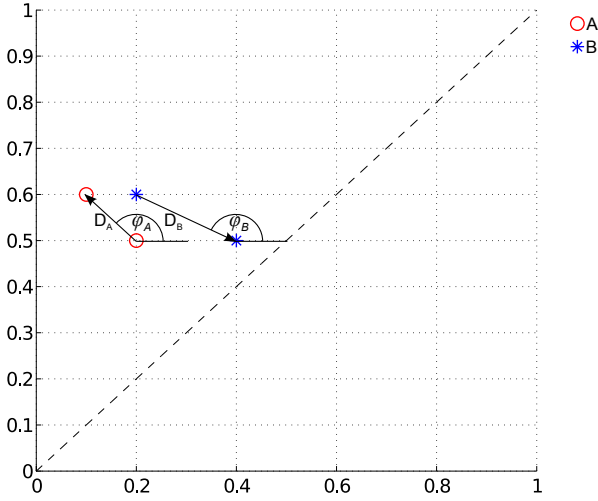


Figure 4.16: Comparison of two detectors.

4.4 Train track noise

As already mentioned, the interest of this project is to apply the transient event detection techniques to train track noise measurements. The noise produced by a train is caused by different mechanisms. The most important component of train track noise is known as rolling noise. It is produced by the interaction of wheels and rails. This kind of noise is broad band, increasing towards high frequencies at higher speeds. Another product of the rolling movement of the wheels is impact noise, it is mainly caused by irregularities in the tracks and wheels. This project is devoted to the detection of impact noise events, specifically to transient events produced by rail joints. Rail joints are present where different segments of a track come together. There is a gap between both segments and sometimes also a difference in height. The noise produced by the wheel when it passes over a rail joint can be regarded as a product of a discrete input applied to the wheel/rail system. This induces quite high force variations [Tho09].

Another component of train track noise is known as curve squeal characterized by its extreme tonality. This kind of noise has a high frequency content consisting also of harmonics, not just the fundamental tone. This tonal characteristic is related to the dimensions of the wheel. Aerodynamic noise contributes also to noise generation. It can be broad-band and also tonal, depending on the

structures in the train. This kind of noise also increases with speed, however, much of the energy is located in the lower part of the frequency region. Also vibrations transmitted through the ground contribute to the train track noise, but this contribution is mainly at low frequencies [Tho09]. Thus, it makes good sense to high pass filter the train track noise signal from 1000 Hz before any detection is attempted.

Figure 4.17 shows an example of two signals containing transient events and without transient events, respectively. Their spectrograms is shown as well. Observe the broadband nature of a transient event in figure 4.17c.

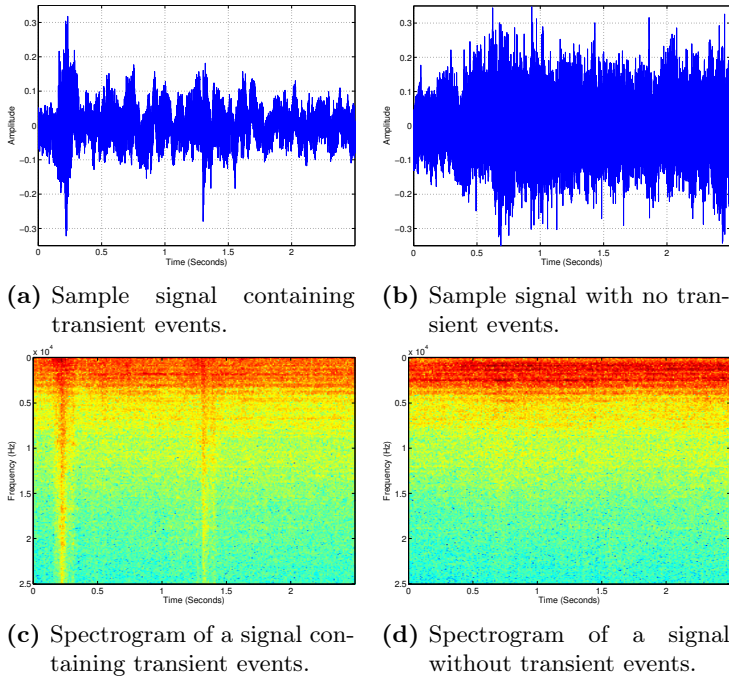


Figure 4.17: Sample of two train track noise signals and respective spectrograms.

The train track noise measurement used in this project was obtained by placing a microphone below the train. The signal has a duration of approximately 1 hour and it is sampled at 50000 samples per second. The speed of the train was also recorded during the noise measurement. The speed profile of the train is shown in figure 4.18, note how the train varies its speed from 0 to a maximum speed of approximately 180 km/h , exhibiting some areas at constant speed.

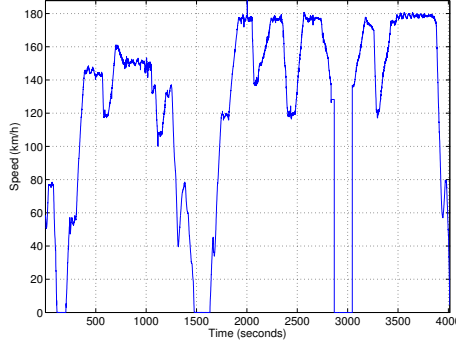


Figure 4.18: Speed profile of the train recorded for this project.

4.4.1 Data sets

One training data set was considered with a duration of 235 seconds starting at second 3591, containing a total of 62 events. It consisted of a chunk of data at approximately constant speed of 176 km/h .

Two test data sets were considered. The first data set has a duration of 136 seconds starting at second 1916, containing a total of 50 events. The speed of the train in this section was approximately constant at 176 km/h . The second test data set consists of the first 300 seconds of the signal, without considering the part where the train is stopped. Thus, the total length of this data set is 204 seconds and the total number of transient events is 48. Note, in figure 4.18, how the speed varies from 0 to approximately 80 km/h in this last test data set.

The purpose of choosing a training data set at constant speed was to reduce the complexity of the data models. However, the second test data set includes data at different speeds. Thus, more information on the effect of speed on the performance of detectors with data models might become clearer.

4.4.2 Manual event detection

As there was no information on the location of the events of interest in the train track noise, the event detection was performed by three subjects running a listening test. The three subjects performed the detection of impulsive transient events in the data sets at constant speed. Whenever there was consensus by at least two of them on the occurrence of an event, that event was validated as existent. For the second test data set, only one of the subjects performed the

detection. However, the best way to localize this events would be by matching the signal with the actual physical defects in the tracks.

Results and Discussion

In the following sections, results and discussion of the ROC curves are shown for the 17 proposed detectors and its different parameters. While consistency results are only shown for a list of the 14 detector variations that exhibited the largest $AUC(0.2)$.

5.1 ROC results

As the detectors were divided in two categories according to their detection strategy, namely, novelty detectors and simple threshold detectors. The obtained ROC curves are also grouped in the same categories.

5.1.1 Novelty Detectors

For the novelty detectors, it is not possible to directly evaluate the performance of a feature set in terms of how good it is at describing transient events. As there exists another important variable, the data model, the feature set and the model are evaluated together. Hence, the results presented in this section show how good a data model, based on a particular feature set, is at discriminating

transient events from non-transient events in the training data set. However, these results serve as a first step to identify if a detector could work reasonably well under general data sets.

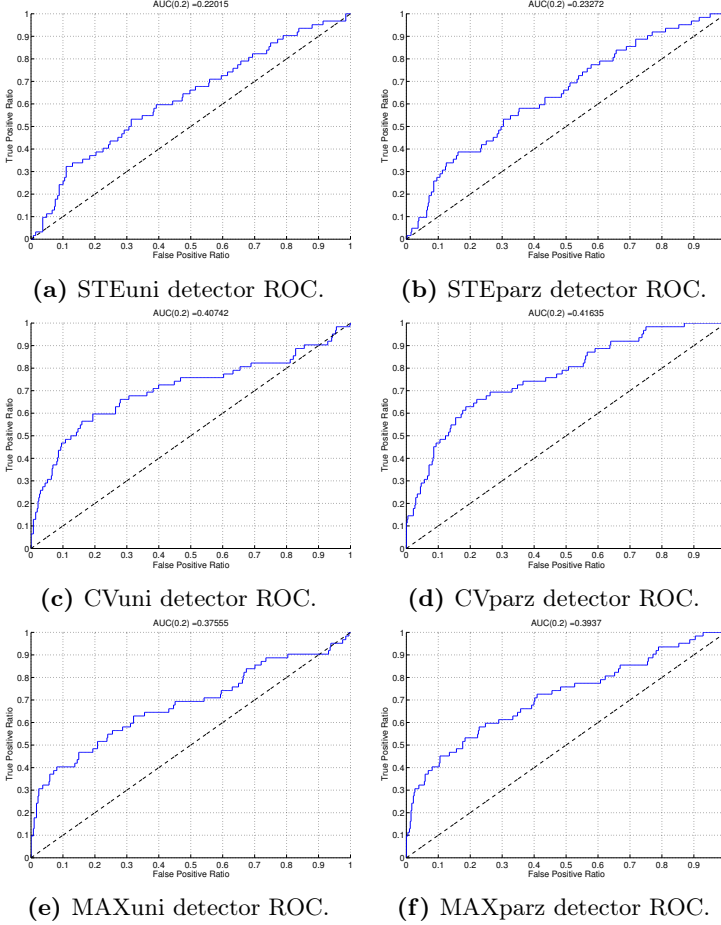


Figure 5.1: ROC curves of detectors with unimodal univariate models. Left column, Gaussian model. Right column, Parzen window estimate.

Figure 5.1 shows the ROC curves for the detectors *STEuni*, *CVuni*, *MAXuni*, *STEparz*, *CVparz* and *MAXparz* detectors, which use an univariate data model. The first thing to notice is that the $AUC(0.2)$ of the *STEuni*-*STEparz*, *CVuni*-*CVparz* and *MAXuni*-*MAXparz* detectors does not differ too much from each other, meaning that no significant improvement is obtained by using a more detailed data model as the one provided by a Parzen window density estimate. It could be thought that a Gaussian model does not provide a very accurate model

for this data as it is not Gaussian distributed as shown by the Kolmogorov-Smirnov test. Figure 5.2 shows the histogram of the data set used to create the model. The fitted Gaussian model as well as the Parzen density estimate (dashed lines) are also shown. Note how the main difference between both models happens in the centre part of the distribution. While the tails of the models approximately coincide. The novelty detection approach focuses on data points that lie outside of the model according to a threshold value. Thus, when the threshold is low enough, the feature values that lie outside of the model are concentrated in the tails of the distribution (as shown in figure 3.9) where the Gaussian model and the Parzen density estimate mostly coincide. This low threshold also produces low FPR values as the positives also concentrate in the tails of the distribution. Hence, the $AUC(0.2)$ using both kinds of models varies very little.

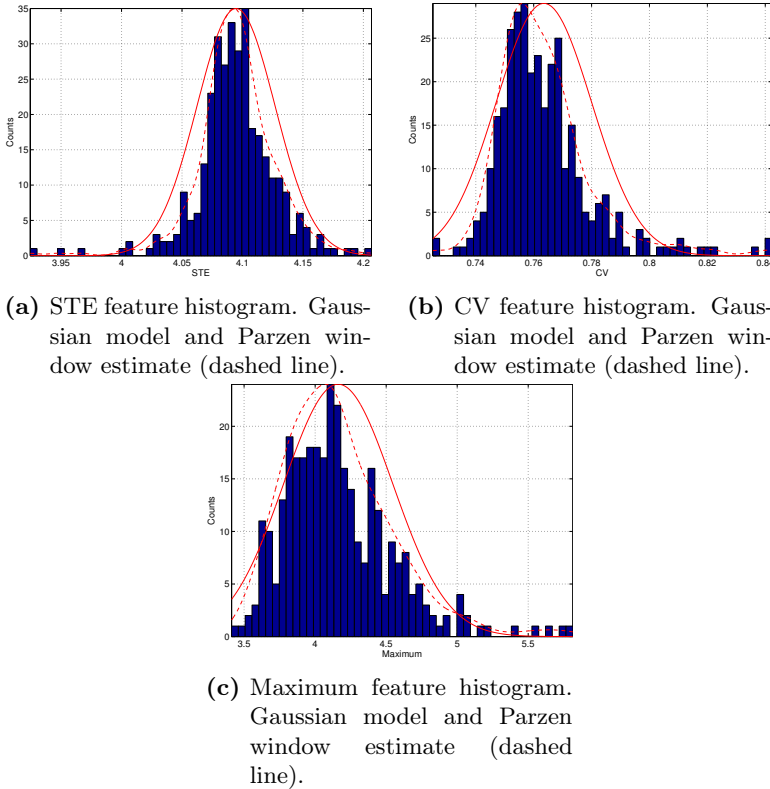


Figure 5.2: STE, CV and Maximum feature histograms for transient-free training data set. Gaussian and Parzen window models compared.

It can also be observed, from figure 5.1, that neither of the features used is able

to fully characterize the specific kind of transient events to be detected. This can be observed from the low $AUC(0.2)$ values obtained. However, it can be seen that, as the detectors that use these features perform above the random guessing diagonal line in ROC space, they provide some useful information on the transient events to detect. Hence, it makes good sense to combine them in a multivariate model. A GMM is flexible enough to characterize multi-modal multi-variate data making the Parzen window estimate unnecessary. The decision of not using Parzen window estimate further in the project is also motivated by the complexity of finding the optimal width of the used kernel functions in multi-dimensional space versus the well known theory on GMM.

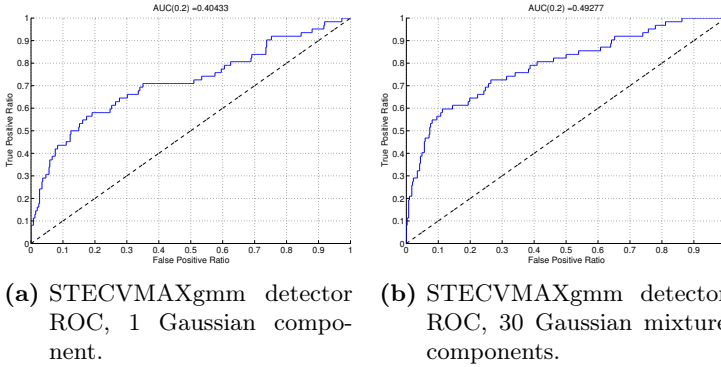


Figure 5.3: STECVMAXgmm detector ROC curves.

Figure 5.3 shows the ROC curves for the STECVMAXgmm detector for 1 and 30 Gaussian components in its GMM, respectively. It can be observed how, when the number of Gaussian mixture components increase, the $AUC(0.2)$ increases as well. This better result corresponds to an improvement of the data model. However, is hard to tell which number of components represent in the best way the real data distribution, taking into account that the model estimation is done based on a finite sample data set. Thus, the best approach to get the number of components for the data model is given by the AIC (equation 3.22) which penalizes the increase of number of components in the mixture. In this way, the AIC gives a measure of the least number of components that best model the data.

Figure 5.4 show the ROC curves for the MFCCgmm detector with 1 Gaussian component and 14 Gaussian mixture components, respectively. Note again, how the increase of Gaussian components improves the $AUC(0.2)$. In comparison with the STECVMAXgmm detector, the MFCCgmm detector yields greater $AUC(0.2)$, meaning that this feature-model is better at discriminating transient events than the combination of the *STE*, *CV* and *Maximum* features in a GMM, at least in the training data set. However, extreme caution has to be

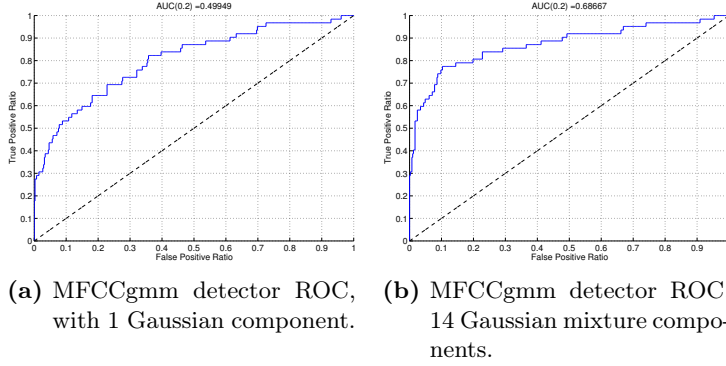


Figure 5.4: MFCCgmm detector ROC curves.

taken due to the dimensionality of the model used by the *MFCCgmm* detector. This will be discussed in following sections.

Figure 5.5 shows the ROC curves for the *FREQNMFgmm* detector. As expected, an increase of the number of Gaussian components increments the *AUC*. This improvement can be seen from figures 5.5a - 5.5b with lower *AUC*, to figures 5.5c - 5.5d with higher *AUC*. Another important aspect to notice is the variation in *AUC(0.2)* due to the change of number of basis components in the NMF. NMF can be regarded as a dimensionality reduction technique. Thus, to achieve this reduction, it has to locate redundant data in the variables and perform the reduction by concentrating more information in the new reduced dimensional space discarding some information in the process. In this detector, matrix W can be interpreted as a set of 5 dimensional variables. Through NMF, these data is reduced to 3 dimensions and 4 dimensions variables, respectively. As already mentioned, this reduction is only approximate and some information is lost. With the produced NMF, matrices W and H can be used to calculate how much information is lost in the squared Euclidean distance sense. Thus, for a greater distance between V and WH , more information is lost. For this particular detector, with a 4 components NMF, the squared euclidean distance is

$$\| V - W_{r=4}H \|^2 = 0.0085$$

while for a 3 components NMF,

$$\| V - W_{r=3}H \|^2 = 0.2433$$

reflecting the loss of information as the dimensionality of the data is reduced. However, it is difficult to tell if the lost information is vital for the correct detection of transient events. But for this case, the dimensionality reduction brings some benefit, an increment in the *AUC(0.2)*.

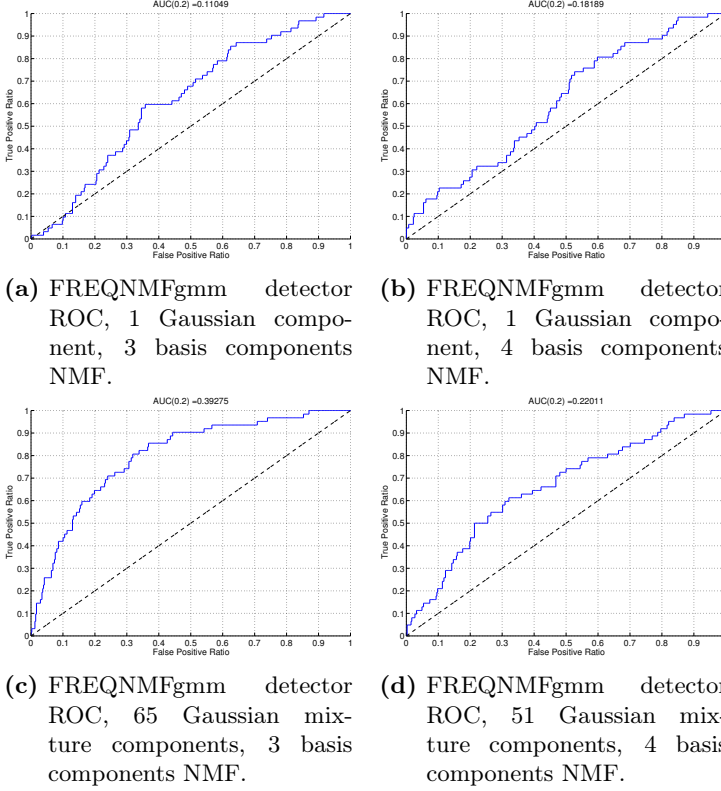


Figure 5.5: FREQNMFGmm detector ROC curves.

Figure 5.6 shows the ROC curves for the *TIMENMFgmm* detector. As with the *FREQNMFGmm* detector, an improvement of the $AUC(0.2)$ is achieved by increasing the number of Gaussian mixture components and decreasing the number of NMF basis components. Once again, for this particular detector, the squared euclidean distance with 12 basis components is

$$\|V - W_{r=12}H\|^2 = 25755.14$$

while for a 3 components NMF,

$$\|V - W_{r=3}H\|^2 = 28757.7$$

Note how the squared distance is really big compared to the one found for the *FREQNMFGmm* detector. This is due to the huge dimensionality reduction in this detector. NMF reduces the dimensionality of the data from 625 dimensions to 3 and 12 dimensions, respectively. However, is important to note that, despite the loss of information, this detector is able to yield an $AUC(0.2)$ larger than

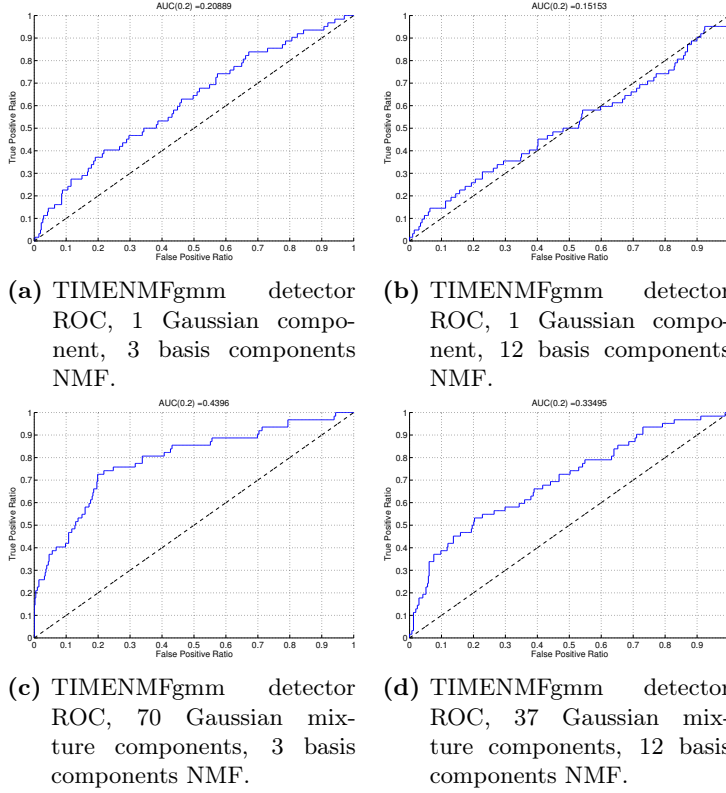


Figure 5.6: TIMENMFgmm detector ROC curves.

the *FREQNMFgmm* detector. This fact shows that the loss of information has not impacted the characterization of transient events using this feature set.

In order to determine why, even with high loss of information due to dimensionality reduction with NMF, transient event detection is possible, a deeper knowledge on the construction of NMF basis components is needed. The calculation of the basis components is totally related with the characteristics of the discarded data to perform the dimensionality reduction.

Figure 5.7 shows the ROC curves for the *FREQnn* detector. Note how an increase of neurons in the hidden layer contributes to an increase in the $AUC(0.2)$. This is contrary to what it has been found with the *FREQNMFgmm* detector. As it has been mentioned, both techniques perform a dimensionality reduction. However, both ROC curves cannot be directly compared as they depend on other particularities of the detectors. For instance, the *NMFgmm* detectors perform

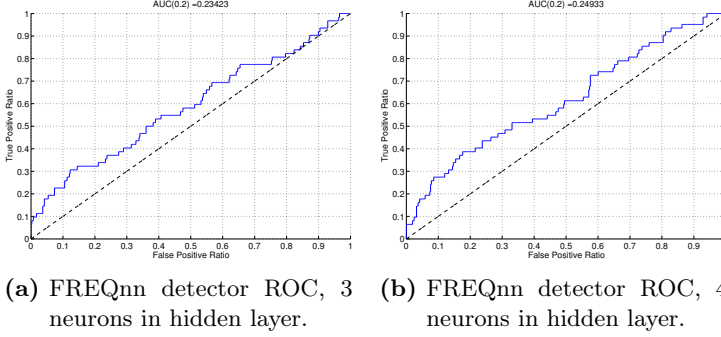


Figure 5.7: FREQnn detector ROC curves.

a dimensionality reduction and model the lower dimensional data with a GMM. The neural network detectors perform dimensionality reduction and then try to reconstruct the original input from the lower dimensional representation of the input. As the basis components reside in the internal structure of the network and this information is not accessible in a direct manner, a direct comparison with the *NMFgmm* detectors is not possible. However, it is possible to compare them using equation 3.25 as the reconstructed input values are accessible in both cases. For the *NMFgmm* detectors, the reconstruction error can be calculated as

$$Error = \sum_i |v_i - Wh_i| \quad (5.1)$$

where v_i is an input column vector and h_i is the projection of vector v_i into W . Then, note that the reconstructed input is the product Wh_i .

Figure 5.8 shows the error for the transient-free training data set. Figures 5.8a and 5.8c show the transient-free training data set reconstruction error using a 4 dimensions reduction for a neural network and for a NMF, respectively. Figures 5.8b and 5.8d show the transient-free training data set reconstruction error using a 3 dimensions reduction for a neural network and for a NMF, respectively. Note how the error increases for NMF and neural network when the dimensionality is reduced from 4 to 3 dimensions. This is expected as a lower dimensional representation loses more data.

On the other hand, if the reconstruction error is calculated from the transient events in the training data, something peculiar happens. For the neural network, the error from 4 neurons in the hidden layer does not change a lot when the number of neurons is decreased to 3. However, for the NMF, the error increases considerably when the number of basis components is decreased. This is shown in figure 5.9. Its worth to remember that the logic of using the neural network is to have higher errors in the reconstruction of transient events while having

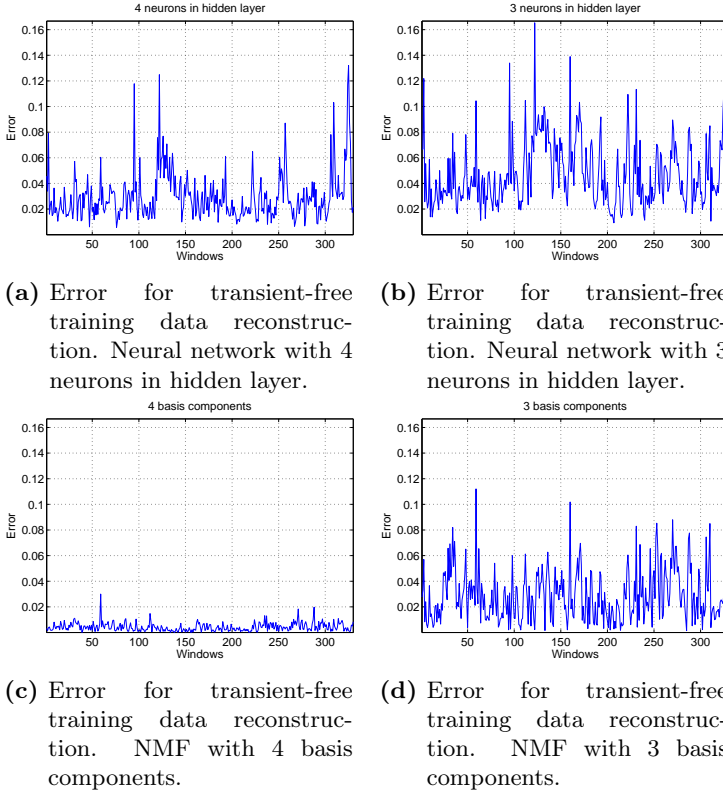


Figure 5.8: Reconstruction error of transient-free training data for NMF and neural network. Feature values as used in the *FREQnn* detector.

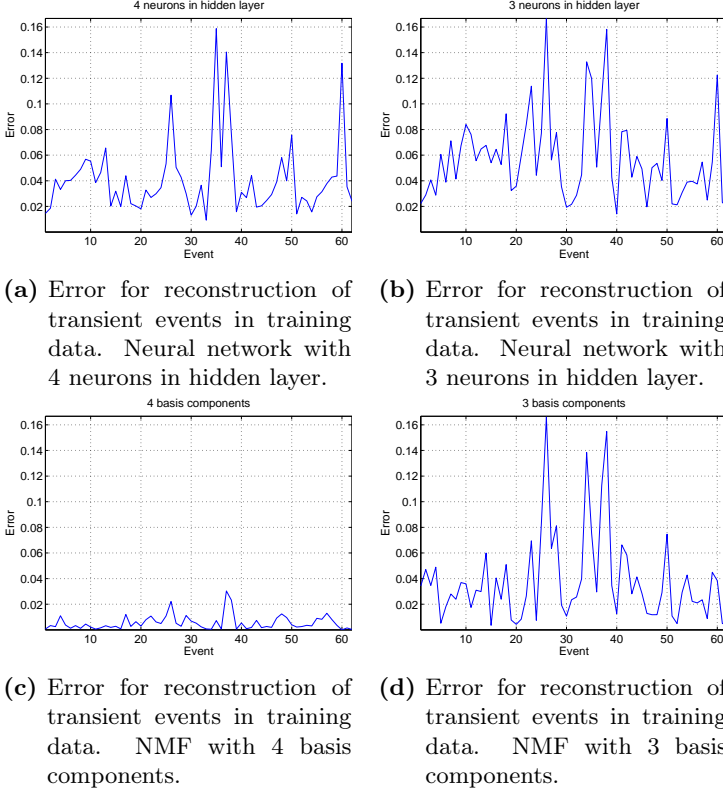


Figure 5.9: Reconstruction error of transient events in training data for NMF and neural network. Feature values as used in the *FREQnn* detector.

lower errors for the transient-free training data. In summary, as the number of neurons in the hidden layer increases, the reconstruction error in the transient-free training data set is reduced. On the other hand, the reconstruction error of the transient event data keeps having higher values even when the number of neurons in the hidden layer is increased. In this way, the *FREQnn* detector achieves higher $AUC(0.2)$ for a higher number of neurons in the hidden layer. Nevertheless, the $AUC(0.2)$ values are low compared to the ones obtained from the *FREQNMFgmm* detector.

Figure 5.10 shows the ROC curves for the *TIMEnn* detector. As with the *FREQnn* detector, the $AUC(0.2)$ is increased when the number of neurons in the hidden layer is increased. Thus, a similar analysis to the previous one shown for the *FREQnn* detector can be performed.

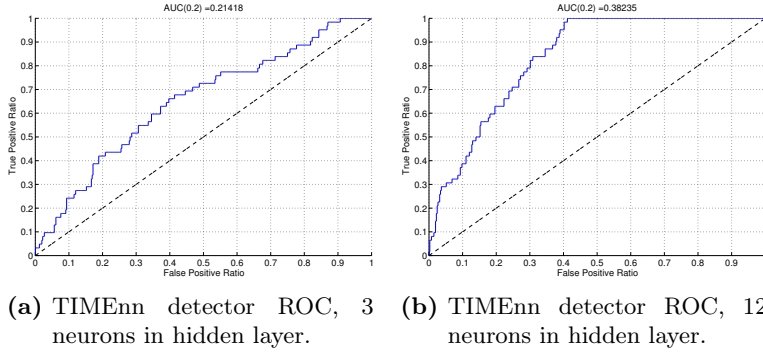


Figure 5.10: TIMEnn detector ROC curves.

Figure 5.11 shows the error for the transient-free training data. Figures 5.11a and 5.11c show the training error using a 12 dimensions reduction for a neural network and for a NMF, respectively. Figures 5.11b and 5.11d show the training error using a 3 dimensions reduction for a neural network and for a NMF, respectively. Again, note how the reconstruction error increases when the dimensionality is reduced from 12 to 3 dimensions. This is expected as a lower dimensional representation loses more data.

Figure 5.12 shows how the error for the reconstruction of transient events increases when the number of hidden neurons in the hidden layer increases. This is a desirable behaviour because, if the error decreases with the transient-free training data, then it means the network is able to discern between transient events and non transient events data. In this way, the *TIMEnn* detector achieves higher $AUC(0.2)$ for a higher number of neurons in the hidden layer.

The way in which the internal lower dimensional representations of the input data is generated in the auto-encoder, is totally related with the characteristics of the discarded data to perform the dimensionality reduction. Thus, in order to determine why, when the number of hidden neurons in a neural network increases, the $AUC(0.2)$ increases as well, a deeper knowledge on the dimensionality reduction capabilities of an auto-encoder is needed.

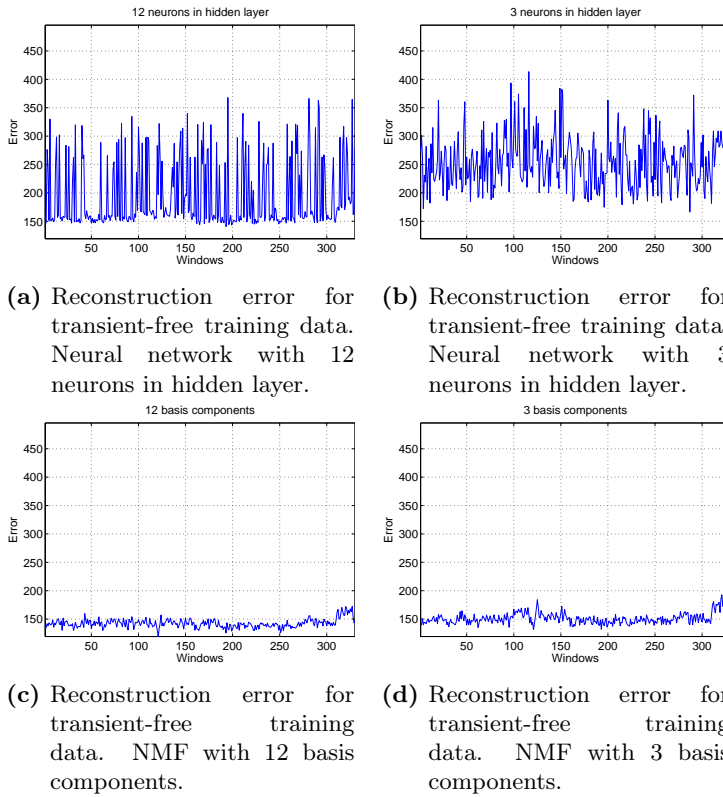


Figure 5.11: Reconstruction error of transient-free training data for NMF and neural network. Feature values as used in the *TIMEnn* detector.

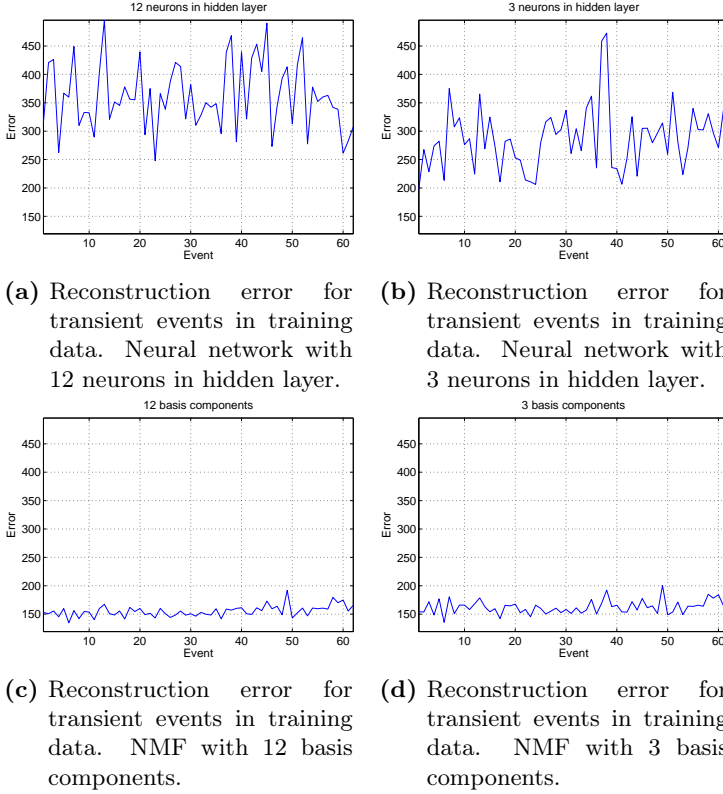


Figure 5.12: Reconstruction error of transient events in training data for NMF and neural network. Feature values as used in the *TIME_{nn}* detector.

5.1.2 Simple Threshold Detectors

Figure 5.13 shows the ROC curve for the *TEO* detector. As it has been mentioned, the Teager Energy Operator acts as a non-linear high-pass filter attenuating low frequency background noise. Impulse noise is characterized for being broadband, hence when extracting the TEO feature from a train track noise measurement, the low frequency background is expected to be attenuated while the high frequency components of any present transient events are preserved. In this way, the *TEO* detector is able to produce high $AUC(0.2)$ as shown in figure 5.13.

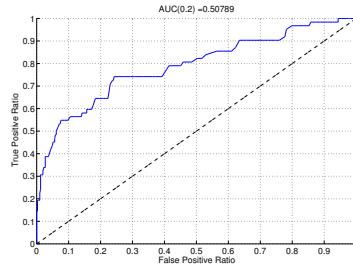


Figure 5.13: *TEO* detector ROC curve.

Figure 5.14 shows the ROC curves for the *TEOFREQ* detector. Note how an analysis in frequency bands does not improve the $AUC(0.2)$, on the contrary, it is decreased. The *TEOFREQ* detector filters a signal in frequency bands. At each frame it calculates the TEO feature values and normalizes that result to its RMS value. For low frequency bands, the TEO feature produces low value results as the high frequency components have been filtered out of the signal. Recall that the TEO acts as non-linear high-pass filter. Thus, after normalization to RMS, the window segment contains high levels of noise. For high frequency bands, the TEO feature is able to produce more accurate results as high frequency components are still present in the signal. From figure 5.14, it can also be observed how the *or* detection strategy performs worse than the *and* detection strategy. This happens because low frequency bands contain high levels of noise, thus, the *or* strategy tends to yield more false positive results when no transient events are present in the signal. While the *and* strategy encourages the detector to yield true positives only when there are high values of the TEO normalized feature in higher frequency bands.

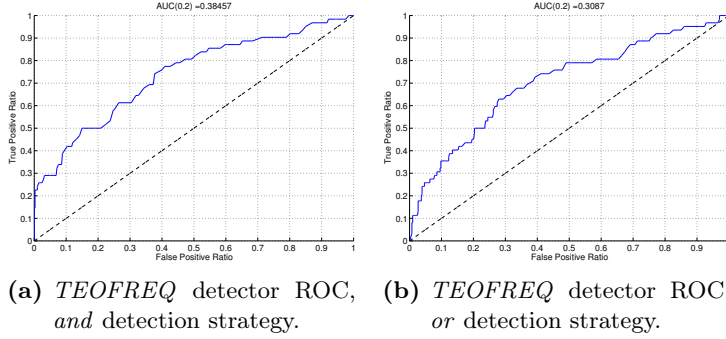


Figure 5.14: *TEOFREQ* detector ROC curves.

Figure 5.15 shows the ROC curves for the *COE* detector. Note how an increase of the parameter d improves the $AUC(0.2)$ up to a certain value, and then the $AUC(0.2)$ decreases. This could be related to the duration of transient events, however it remains unclear how to find an optimal d parameter.

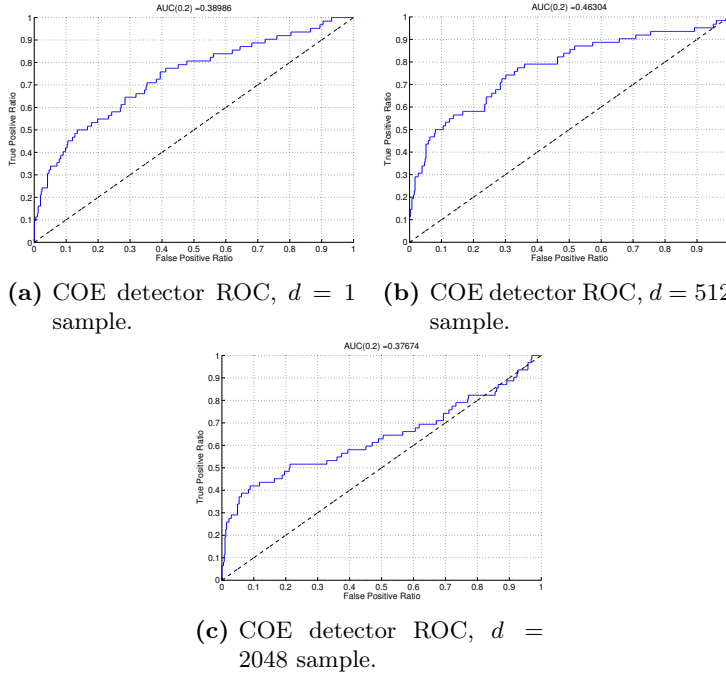


Figure 5.15: *COE* detector ROC curves.

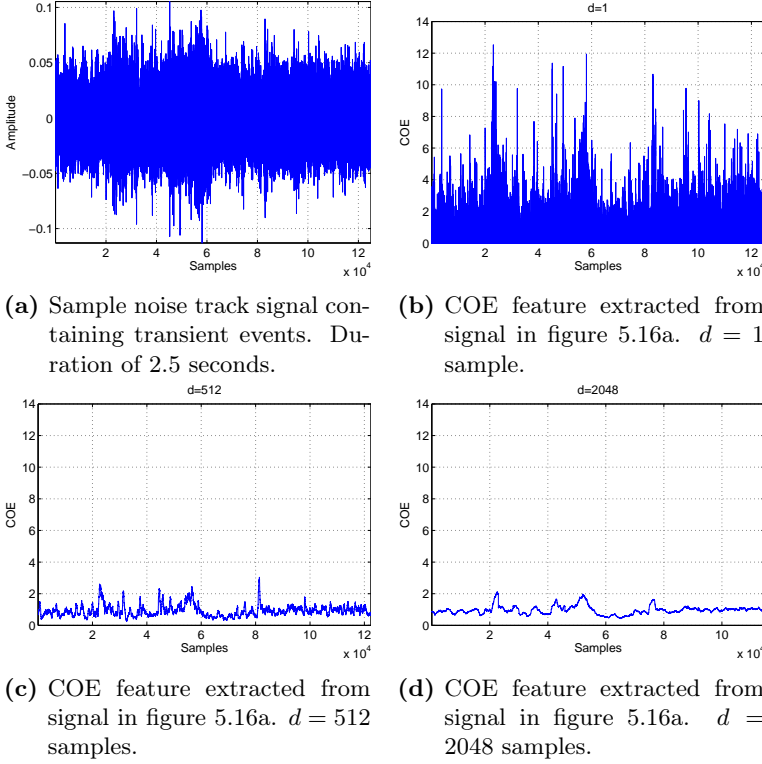


Figure 5.16: Noise track sample data containing transient events. COE feature at different d values.

To better understand the COE feature's behaviour as used by the *COE* detector, the following figures can be used. Figure 5.16 shows how the COE feature behaves for different values of d . The signal in figure 5.16a contains some transient events in it. Figures 5.16b, 5.16c and 5.16d show how this feature changes as the parameter d increases. Recall that, according to the processing of the detector, the COE feature values obtained by window are normalized by its RMS value in that window. The general effect of extracting the COE feature is a concentration of the energy of the signal. This concentration is, recalling the COE theory, a summation of the energy in the past d samples. However, if the value of d is high, an extreme concentration of the energy in the signal happens, making less evident the peaks of energy that characterize impulsive transient events. This is shown in figure 5.16d. Thus, there exists a parameter d that provides a good concentration of energy yielding better $AUC(0.2)$ values. In this case, $d = 512$ samples produced the highest $AUC(0.2)$ for this detector.

Figure 5.17 shows the ROC curves for the *COEFREQ* detector. In contrast with the TEO feature, the COE feature is suitable for its application in frequency bands.

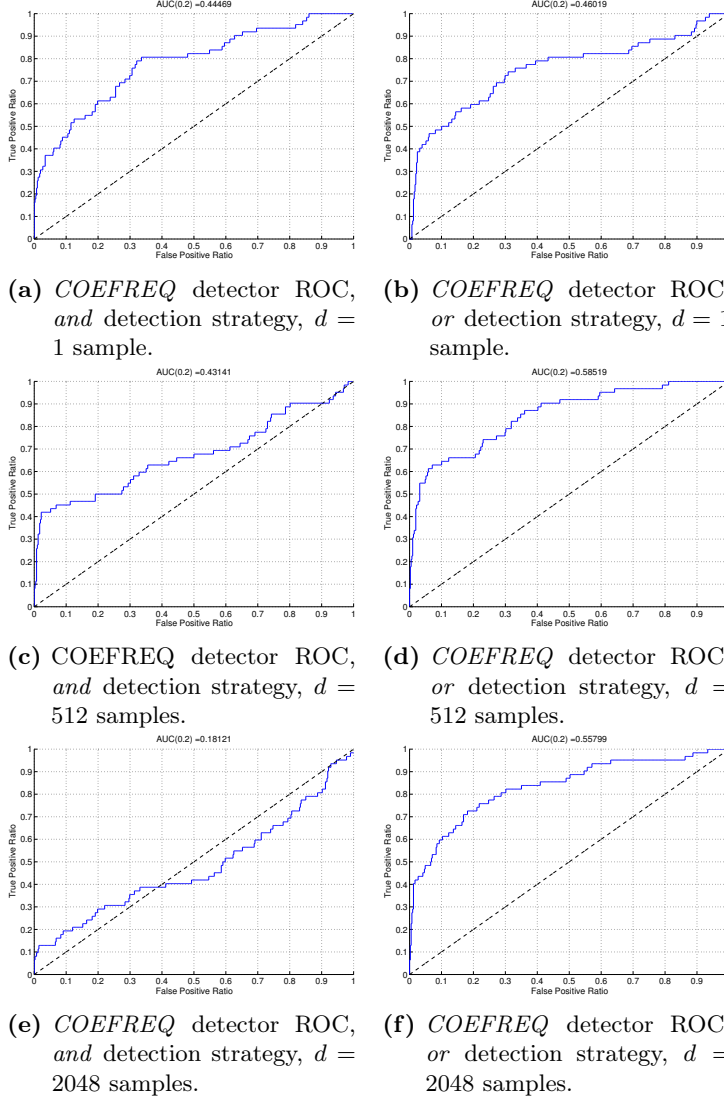


Figure 5.17: *COEFREQ* detector ROC curves.

The COE feature relies on the change of energy of the signal. Thus, a COE broadband analysis of a signal might not be able to detect tenuous changes of energy occurring in high frequency bands due to a masking effect by components

with more energy. Hence, the COE feature is suitable for a frequency band analysis.

This detector produces better results using the *or* detection strategy than when using the *and* detection strategy. This can be observed in figures 5.17b, 5.17d and 5.17f. Recalling the *or* detection strategy, a window segment is declared as a positive when one or more COE feature values exceed a threshold in any frequency band. Thus, the detector is able to detect transient events even if the feature exceeds the threshold value in one frequency band.

The effect of varying parameter d affects the *COEFREQ* detector in the same way it affects the *COE* detector, but in each frequency band. This is shown in figures 5.17b, 5.17d and 5.17f, where $d = 512$ produced the best $AUC(0.2)$ for this detector.

Figure 5.18 shows the ROC curves for the *LPS* detector. As it can be seen from the figure, parameter $STFT_{window}$ affects the $AUC(0.2)$ values obtained. However, in the STFT this parameter is related to the frequency resolution of the Fourier transform and to the time localization of any changes in frequency. Thus, the change in $AUC(0.2)$ is thought to be related to the duration of a transient event, where a window of the same length as an event would consider all of the energy injected by the event, producing high magnitude values in each frequency bin. Note how the best $AUC(0.2)$ values were achieved using a window of 512 samples for the *COE*, *COEFREQ* and *LPS* detectors.

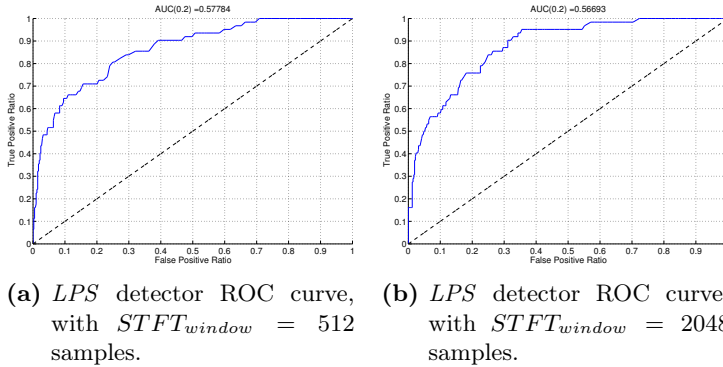


Figure 5.18: *LPS* detector ROC curves.

5.2 AUC(0.2) results

Table 5.1 shows the list of the promising detectors based on the highest $AUC(0.2)$. The detectors that had an $AUC(0.2)$ larger or equal to the $AUC(0.2)$ of the best univariate detector were chosen as promising. The main logic behind this arbitrary election is, if a complex detector cannot perform better than a simple one, then it is not worth considering it. However, this does not mean the discarded detector's settings cannot be tweaked to outperform the simplest detectors.

Table 5.1: List of promising detectors, parameters and $AUC(0.2)$, in $AUC(0.2)$ descending order.

Name	Parameters	$AUC(0.2)$
<i>MFCCgmm</i>	14 Gaussian mixture components	0.6867
<i>COEFREQ</i>	<i>or</i> detection strategy $d = 512$ samples	0.5852
<i>LPS</i>	$STFT_{window} = 512$ samples	0.5778
<i>LPS</i>	$STFT_{window} = 2048$ samples	0.5669
<i>COEFREQ</i>	<i>or</i> detection strategy $d = 2048$ samples	0.5580
<i>TEO</i>	N/A	0.5079
<i>MFCCgmm</i>	1 Gaussian component	0.4995
<i>STECVMAXgmm</i>	30 Gaussian mixture components	0.4928
<i>COE</i>	$d = 512$ samples	0.4630
<i>COEFREQ</i>	<i>or</i> detection strategy $d = 1$ sample	0.4602
<i>COEFREQ</i>	<i>and</i> detection strategy $d = 1$ sample	0.4447
<i>TIMENMFgmm</i>	70 Gaussian mixture components 3 NMF basis components	0.4396
<i>COEFREQ</i>	<i>and</i> detection strategy $d = 512$ samples	0.4314
<i>CVparz</i>	N/A	0.4164

The detectors and settings shown in table 5.1 were tested for consistency results. A list of all detectors, its relevant parameters and relevant ROC values can be found in table A.1 in the appendix.

The results shown in this section have to be considered carefully, as they only reflect the capacity of the detectors to perform transient event detection in a specific test. Hence, the results might change drastically in the evaluation of the detectors in other data sets. Those results are now analysed.

5.3 Consistency test results

Table 5.2 shows the results for the consistency test using test data set 1. In table A.2 in the appendix all the relevant information of these results can be consulted.

The first thing to notice from table 5.2 is how most of the detectors based on the novelty detection approach exhibited the lowest consistency results. This is even more surprising due to the fact that the test data set for this results had similar speed as the training data. Hence, it can be said that the training data set used to generate the models was not general enough to perform well on other data sets. Moreover, by listening to the training data set and test data set 1, it can be perceived how both present different background noise elements. This could explain at some extent the generalization issue of the obtained data models used by the detectors.

Another important fact to consider is the number of samples to create the models. For the training data set, the number of transient free samples available was 338. For the *STECVMAXgmm* and *TIMENMFgmm* detectors the dimensionality of the model was 3 dimensions. As the covariance matrix for the GMM was restricted to be diagonal, only P parameters had to be estimated for each Gaussian component in the mixture. Moreover, P more parameters had to be estimated for the mean values of each Gaussian component in the mixture. In total $2P$ parameters need to be estimated for each Gaussian component. If we consider that, at least, one data sample is needed to fit a Gaussian component, then we can obtain the number of minimum data samples needed to appropriately fit a model. Thus, for the *STECVMAXgmm* model with 30 Gaussian mixture components, at least $2 \cdot P \cdot 30 = 2 \cdot 3 \cdot 30 = 180$ data samples were needed. Hence, it can be concluded that this model was fitted appropriately. For the *TIMENMFgmm* model with 70 Gaussian components, the number of samples needed for an appropriate fit was $2 \cdot P \cdot 70 = 2 \cdot 3 \cdot 70 = 420$. As this number exceeds the actual number of samples it can be said that the model was over-fitting

Table 5.2: List of promising detectors. Results of consistency test in test data set 1, in *score* descending order.

No.	Name	Parameters	<i>score</i>
1	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 512$ samples	(0.0498, -0.9893)
2	<i>TEO</i>	N/A	(0.0500, -0.9668)
3	<i>CVparz</i>	N/A	(0.0905, -0.1860)
4	<i>LPS</i>	$STFT_{window} = 512$ samples	(0.0945, -0.8937)
5	<i>LPS</i>	$STFT_{window} = 2048$ samples	(0.1243, -0.8923)
6	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 2048$ samples	(0.1296, -0.6254)
7	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 1$ sample	(0.1516, 0.1630)
8	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 512$ samples	(0.1924, -0.1200)
9	<i>COE</i>	$d = 512$ samples	(0.2072, -0.1707)
10	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 1$ sample	(0.2156, -0.9895)
11	<i>STECVMAXgmm</i>	30 Gaussian mixture components	(0.3126, -0.3329)
12	<i>TIMENMFgmm</i>	70 Gaussian mixture components 3 NMF basis components	(0.3481, -0.7940)
13	<i>MFCCgmm</i>	1 Gaussian component	(0.4438, -0.2258)
14	<i>MFCCgmm</i>	14 Gaussian mixture components	(0.6254, -0.5119)

the data leading to a poor performance of the model. In the other hand, the *MFCCgmm* detectors did not have any fitting issues as, for the 14 Gaussian mixture components, the needed number of samples was $2 \cdot P \cdot 14 = 2 \cdot 12 \cdot 14 = 336$. However, is most likely that the high dimensionality of the model, 12 dimensions, led to an incorrect estimation of the model due to curse of dimensionality issues.

The detector with higher consistency score was the *COEFREQ* detector with $d = 512$ samples, using the *and* strategy. This is mainly due to the detection strategy, which detects an event in a certain window only if all the frequency bands exceed the threshold value. In this sense, the detector yields positives only when it has strong evidence of the existence of events. In this sense, as the training data set and test data set 1 have similar levels of background noise, the detector is expected to perform similar under similar background noise level conditions. Another important point to notice about the *COEFREQ* detector is that the parameter d seems to be associated to the consistency of the detector. Where the $d = 2048$ samples produced the best consistency for this detector under the *or* strategy. However, for $d = 1$, the detector improved its detection performance moving away from the diagonal line in the sense defined by its score. It is worth to mention that this was the only detector that improved its detection performance for this data set.

Another remarkable result is the one obtained by the *CVparz* detector, with a consistency score of 0.0905. This fact suggests there exists a good generalization of the model using the CV feature, at least for data at the same speed.

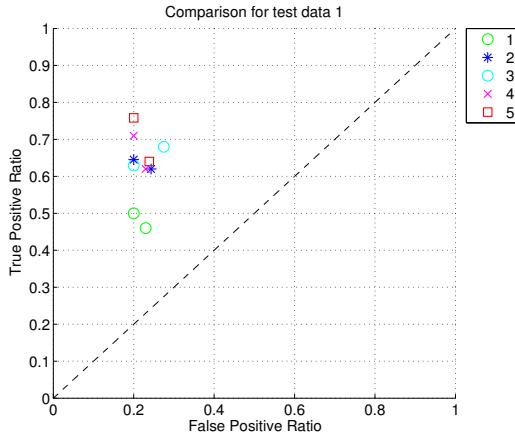


Figure 5.19: (FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 1 for the first 5 detectors according to table 5.2. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.

The *LPS* detector presented the highest TPR_{ref} of all the detectors. Even if its consistency was not the best, its new TPR was still among the highest, only outperformed by the *CVparz* detector. However, note that the *LPS* detector has a lower FPR than the *CVparz* detector, making it the best detector in detection performance for test data set 1. This can be seen in figure 5.19, which, for clarity, only shows how the first 5 detectors in table 5.2 moved from (FPR_{ref}, TPR_{ref}) to (FPR, TPR) . A comparison of all the promising detectors for this test data set can be found in figure A.1 in the appendix.

Table 5.3: List of promising detectors. Results of consistency test in test data set 2, in *score* descending order.

No.	Name	Parameters	<i>score</i>
5	<i>LPS</i>	$STFT_{window} = 2048$ samples	(0.0826,0.1386)
4	<i>LPS</i>	$STFT_{window} = 512$ samples	(0.1128,0.9353)
1	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 512$ samples	(0.2058,0.3524)
3	<i>CVparz</i>	N/A	(0.2267,0.6204)
12	<i>TIMENMFgmm</i>	70 Gaussian mixture components 3 NMF basis components	(0.2502,-0.4960)
6	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 2048$ samples	(0.2542,0.1972)
8	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 512$ samples	(0.2792,0.3610)
2	<i>TEO</i>	N/A	(0.3242,0.2059)
10	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 1$ sample	(0.3394,-0.0785)
9	<i>COE</i>	$d = 512$ samples	(0.3569,0.6993)
11	<i>STECVMAXgmm</i>	30 Gaussian mixture components	(0.3612,0.2610)
7	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 1$ sample	(0.3723,0.1468)
14	<i>MFCCgmm</i>	14 Gaussian mixture components	(0.8231,-0.5210)
13	<i>MFCCgmm</i>	1 Gaussian component	(0.8752,-0.3597)

Table 5.3 shows the results for the consistency test using test data set 2. In table A.3 in the appendix all the relevant information of these results can be

consulted.

The first thing to note about this set of results is that the consistency ratings are much higher than for test data set 1. Meaning that the detectors are less consistent in this data set. This is reasonable as test data set 2 was taken from a different chunk of data, where the speed was variable and lower.

For this data set, the *MFCCgmm* detector moved to point (1, 1) in ROC space. This particular point means that the detector defines every window in the test data set as containing an event, finding in that way all the events ($TPR = 1$), but also producing all possible false positives ($FPR = 1$). This can be seen in figure A.2 in the appendix. Figure 5.20 only shows how the first 5 detectors in table 5.3 moved from (FPR_{ref}, TPR_{ref}) to (FPR, TPR) in test data set 2. The reason for such a bad performance of the *MFCCgmm* detector could be related to the frequency characteristics of the data sets due to changes in speed. A lower speed reduces the background noise due to wind and other factors, which mainly manifest at higher frequencies.

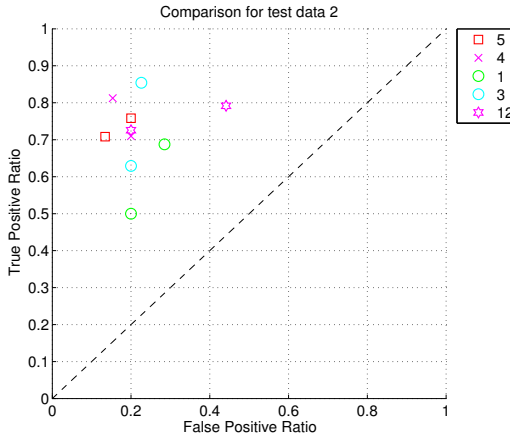


Figure 5.20: (FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 2 for the first 5 detectors according to table 5.3. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.

This time, the *LPS* detectors presented the best consistency results, and moreover, its displacement was towards an increase in detection performance. The *COEFREQ* detector kept being at the first places in the consistency test as well as the *CVparz* detector. Regarding the *COEFREQ* detector, the *and* strategy ensures the detector only yields positives when there is strong evidence of the presence of events. Regarding the *CVparz* detector, the results suggest that the CV feature possesses certain ability to describe transient events without being

largely affected by background noise introduced by speed changes.

The *TIMENMFgmm* detector improved its consistency relative to the other detectors, but it was not a great improvement compared to its consistency result in test data set 1. Once again, consistency issues seem to be related to parameter d in the *COEFREQ* detector using an *or* detection strategy, where a larger value of d produces more consistent results.

An important observation is that the *TEO* detector decremented its consistency performance. However, its detection performance increased, moving to point (0.3771, 0.9167). This improvement could be as a product of an attenuation of high frequency background noise due to lower speeds, making it easier for the feature to reveal high frequency sudden changes of energy. Recall the *TEO* feature functions as non-linear high-pass filter.

The *COE* detector also decreased its consistency performance, however its detection performance is the best of all of the detectors, moving into point (0.2039, 0.9375) in ROC space.

For this test data set, most of the detectors presented low consistency ratings, however, that occurred because they showed improvements in its detection performance ratings. This suggests that, in general, it is easier to detect transient events in lower speed data. This is related to the increase of background noise with an increase of speed of the train.

It is difficult to define which of the detectors has the best overall performance. A higher $AUC(0.2)$ is desired to detect most of the transient events present in a noise track signal while keeping the false positives at minimum. However, consistency is also desired to be able to predict the performance of the detector for different data sets. Having a consistency rating of zero for different data sets would represent to have a strong certainty of the detection rates regardless of the data set. Thus, some detectors had high $AUC(0.2)$ but did not have good consistency results.

However, it could be said that the training data set belonged to a very noisy portion of the signal, representing the most hard condition in which a detector can operate. Thus, in terms of a detector's goal, the training data could be regarded as the limit detection performance condition and no worse true positive and false positive rates should be accepted for data sets with less noisy conditions. In that sense most of the simple threshold detectors yielded good results, improving their detection performance for a less noisy environment.

Conclusion

Transient event detection is a very broad subject, and a broad number of different approaches are applied from different points of view. The fields that apply transient event detection techniques range from medical applications, to seismology and machine monitoring, among others. Naturally, the relevant techniques to investigate were those which were applied to a similar field of interest. In this project, the application field was detection of transient events in train track noise. The kind of transient events to detect were impulsive events produced by uneven rail joints. Thus, the relevant fields in literature review were those in which the term transient event was associated with sudden injection of energy to a system. The found relevant fields of application were machine monitoring, sound surveillance, power quality, among others.

A detection task can be divided into feature extraction and application of a detection strategy. The feature extraction process is crucial for a correct characterization of transient events. Then, a detection strategy can analyse these extracted features to conclude on the detection of transient events. The relevant features for this project were mostly associated with the frequency information and energy in a train track noise signal. Moreover, 2 features and 1 detection strategy based on the energy of the signal and its frequency content were proposed.

The studied detection strategies can be divided in novelty detection and simple

threshold detection. The novelty detection strategy involves the characterization of the normal behaviour of a system in order to identify when the system is performing outside its normal operation state. The simple threshold strategy involves the direct evaluation of feature values, looking for those that exceed a threshold in order to detect events.

Once that the architecture of the detectors is defined, their performance can be evaluated in terms of a Receiver Operating Characteristic curve, showing the relation between True Positive Ratio and False Positive Ratio values as a function of a parameter k of the detector. This ROC curve provides a means for the selection of suitable detectors with high detection performance, that is, high True Positive Ratios at low False Positive Ratios. Moreover, a measure of consistency performance was developed to evaluate the consistency of detector results for different test data sets. The consistency of a detector is important to be able to predict its performance in different data sets. Providing, in that way, some confidence on the detection results regardless of the evaluated test data set.

Thus, a processing framework, implemented in Matlab, incorporating the detectors and the performance evaluation modules was created. The framework is flexible enough to allow for the creation of new detectors, based on the implemented features or new features, and the evaluation of them in terms of its detection and consistency performance. A description of the framework's structure in terms of the scripts generated can be found in section B, in the appendix.

A total of 17 detectors was designed, implemented and tested. That is, 12 novelty detection detectors and 5 simple threshold detectors. The characteristics of the training data represented a major issue for the novelty detectors. The training data was taken from a portion of the train track signal at fairly constant speed. However, this speed was the maximum of the train, thus, a high level of background noise was present. Different sources contribute for the characterization of background noise and all of them are accentuated at higher speeds. Thus, complex data models were needed to correctly model the data. A limiting factor was the number of samples to generate the model, affecting the quality of it. Moreover, it was found that the generated data models were not completely valid for test data sets obtained at different train speeds.

The major success of the simple threshold detectors was its simplicity. They tend to generate fairly consistent values in data at approximately the same speed. However, they are not very consistent in data at lower speeds where they tend to increase its detection performance. Moreover, this could be seen as an advantage if a training set at high speeds can be used to provide a limit in detector performance. Thus, the detectors should be expected to perform

better for data sets with less background noise.

A more profound knowledge of the training data set has to be obtained in order to improve the performance of novelty detectors. The large variations of the properties of the signal due to background noise and speed proved to greatly affect the proposed models. Hence, the noise track signal and its variation with speed has to be better understood for the improvement of any data models.

This project was delightful for the author as there are a lot of subjects to study on many interesting topics, in order to solve a complex detection problem. Transient event detection showed to be a very challenging topic, specially when applied to detection in train track noise, where a lot of factors contribute to the characterization of the noise signal. Through the literature review, it was observed that most of the fields under study assumed a very controlled application field, with simulated data or noise free signals for example. When, for this project, the application field was directly taken from a real life problem where no easy solutions seemed to be enough to solve the problem. There is still a lot of work to do, and some guidelines are now presented.

6.1 Future work

The future work ideas showed in this section are very concise and are mostly focused in the improvement of the detectors showed in the project. However, a continuation of the project should naturally lead to the classification problem of identifying different classes of transient events. Hopefully, the proposed features and methods will provide useful information for a classification task. Moreover, some other detection techniques based on, for example, adaptive filtering and wavelets could be studied.

Most of the detectors had a lot of parameters that affected its performance, thus, it is recommended to perform a thorough analysis on them to find the optimal parameters, understanding in that way its limitations.

NMF and auto-encoder neural networks perform dimensionality reduction, however, the characteristics of the generated basis components is different from each other. This differences affect the performance of the detectors that implement each of the approaches. Hence, it is recommended to find out how the basis components are created and what kind of information is discarded in order to perform dimensionality reduction. This knowledge might lead to a suitable number of basis components to set for a specific field of application.

The features used in the simple threshold detectors proved to be effective to characterize transient events, hence, new novelty detectors based on these features can be studied. Recalling that the novelty detectors make use of a data model, thus, the features used in the simple threshold detectors can be incorporated to a data model.

In this moment, the way to select a threshold suitable for the detection of transient events in different training sets is done through the ROC curve. However, a more convenient way to find this value of k could be found by applying Extreme Value Theory, where threshold k is chosen taking into consideration the distribution of data in the tails or extremes of a given model.

The speed information was not directly used in any of the detection strategies. Thus, it is recommended to analyse how to incorporate the it to any data models. An idea could be, for example, a data model as function of speed.

APPENDIX A

Results for ROC and Consistency tests

Table A.1: List of detectors, parameters and ROC relevant information, in AUC(0.2) descending order.

Name	Parameters	TPR_{ref}	k_{ref}	AUC(0.2)
<i>MFCCgmm</i>	14 Gaussian mixture components	0.8065	6.7481e-10	0.6867
<i>COEFREQ</i>	<i>or</i> detection strategy $d = 512$ samples	0.6613	2.2533	0.5852
<i>LPS</i>	$STFT_{window} = 512$ samples	0.7097	6.5301e-02	0.5778
<i>LPS</i>	$STFT_{window} = 2048$ samples	0.7581	4.9620e-02	0.5669
<i>COEFREQ</i>	<i>or</i> detection strategy $d = 2048$ samples	0.7258	1.0945	0.5580

<i>TEO</i>	N/A	0.6452	9.5118	0.5079
<i>MFCCgmm</i>	1 Gaussian component	0.6452	5.9075e-11	0.4995
<i>STECVMAXgmm</i>	30 Gaussian mixture components	0.6452	6.1313e+01	0.4928
<i>COE</i>	$d = 512$ samples	0.5806	1.3088	0.4630
<i>COEFREQ</i>	<i>or</i> detection strategy $d = 1$ sample	0.5968	1.2811e+01	0.4602
<i>COEFREQ</i>	<i>and</i> detection strategy $d = 1$ sample	0.6129	7.8695	0.4447
<i>TIMENMFgmm</i>	70 Gaussian mixture components 3 NMF basis components	0.7258	8.5503	0.4396
<i>COEFREQ</i>	<i>and</i> detection strategy $d = 512$ samples	0.5000	9.7112e-01	0.4314
<i>CVparz</i>	N/A	0.6290	1.2449e+01	0.4164
<i>CVuni</i>	N/A	0.5968	1.4205e+01	0.4074
<i>STECVMAXgmm</i>	1 Gaussian component	0.5806	5.5015e+01	0.4043
<i>MAXparz</i>	N/A	0.5323	5.0669e-01	0.3937
<i>FREQNMFgmm</i>	65 Gaussian mixture components 3 NMF basis components	0.6452	1.8655e+02	0.3927
<i>COE</i>	$d = 1$ sample	0.5484	1.0596e+01	0.3899
<i>TEOFREQ</i>	<i>and</i> detection strategy	0.5000	5.1708	0.3846
<i>TIMEnn</i>	12 neurons in hidden layers	0.6290	3.4091e+002	0.3824

<i>COE</i>	$d = 2048$ samples	0.4839	9.5621e-01	0.3767
<i>MAXuni</i>	N/A	0.4839	5.3631e-01	0.3756
<i>TIMENMFgmm</i>	37 Gaussian mixture components 12 NMF basis components	0.5000	2.7962e+04	0.3350
<i>TEOFREQ</i>	<i>or</i> detection strategy	0.4516	1.0480e+01	0.3087
<i>FREQnn</i>	4 neurons in hidden layer	0.3871	8.1927e-03	0.2493
<i>FREQnn</i>	3 neurons in hidden layer	0.3226	1.9506e-02	0.2342
<i>STeparz</i>	N/A	0.3871	5.0638	0.2327
<i>STEuni</i>	N/A	0.3710	6.3395	0.2202
<i>FREQNMFgmm</i>	51 Gaussian mixture components 4 NMF basis components	0.4194	4.5241	0.2201
<i>TIMEnn</i>	3 neurons in hidden layers	0.4194	2.9513e+002	0.2142
<i>TIMENMFgmm</i>	1 Gaussian component 3 NMF basis components	0.3710	2.4545	0.2089
<i>FREQNMFgmm</i>	1 Gaussian component 4 NMF basis components	0.2742	1.0349e+01	0.1819
<i>COEFREQ</i>	<i>and</i> detection strategy $d = 2048$ samples	0.2903	9.3425e-01	0.1812
<i>TIMENMFgmm</i>	1 Gaussian component	0.2581	4.0381e+03	0.1515

	12 NMF basis components			
<i>FREQNMFgmm</i>	1 Gaussian component 3 NMF basis components	0.2419	3.0567e+01	0.1105

Table A.2: List of 14 promising detectors, parameters and consistency test relevant information, in *score* descending order. Test data set 1.

No.	Name	Parameters	<i>FPR</i>	<i>TPR</i>	<i>score</i>
1	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 512$ samples	0.2297	0.4600	(0.0498, -0.9893)
2	<i>TEO</i>	N/A	0.2432	0.6200	(0.0500, -0.9668)
3	<i>CVparz</i>	N/A	0.2748	0.6800	(0.0905, -0.1860)
4	<i>LPS</i>	$STFT_{window} = 512$ samples	0.2297	0.6200	(0.0945, -0.8937)
5	<i>LPS</i>	$STFT_{window} = 2048$ samples	0.2387	0.6400	(0.1243, -0.8923)
6	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 2048$ samples	0.3288	0.7400	(0.1296, -0.6254)
7	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 1$ sample	0.2883	0.7200	(0.1516, 0.1630)
8	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 512$ samples	0.3514	0.7800	(0.1924, -0.1200)
9	<i>COE</i>	$d = 512$ samples	0.3694	0.7000	(0.2072, -0.1707)
10	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 1$ sample	0.3288	0.4400	(0.2156, -0.9895)
11	<i>STECVMAXgmm</i>	30 Gaussian mixture components	0.4820	0.7800	(0.3126, -0.3329)

12	<i>TIMENMFgmm</i>	70 Gaussian mixture components 3 NMF basis components	0.5450	0.6800	(0.3481, -0.7940)
13	<i>MFCCgmm</i>	1 Gaussian component	0.5766	0.8800	(0.4438, -0.2258)
14	<i>MFCCgmm</i>	14 Gaussian mixture components	0.8063	0.9600	(0.6254, -0.5119)

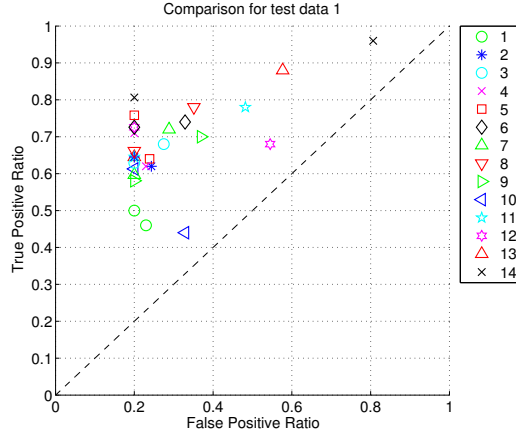


Figure A.1: (FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 1. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.

Table A.3: List of 14 promising detectors, parameters and consistency test relevant information, in *score* descending order. Test data set 2.

No.	Name	Parameters	FPR	TPR	$Score$
5	<i>LPS</i>	$STFT_{window} = 2048$ samples	0.1341	0.7083	(0.0826, 0.1386)
4	<i>LPS</i>	$STFT_{window} = 512$ samples	0.1536	0.8125	(0.1128, 0.9353)
1	<i>COEFREQ</i>	<i>and</i> detection strategy	0.2849	0.6875	(0.2058, 0.3524)

		$d = 512$ samples			
3	<i>CVparz</i>	N/A	0.2263	0.8542	(0.2267,0.6204)
12	<i>TIMENMFgmm</i>	70 Gaussian mixture components 3 NMF basis components	0.4413	0.7917	(0.2502,-0.4960)
6	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 2048$ samples	0.3408	0.9375	(0.2542,0.1972)
8	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 512$ samples	0.3128	0.9167	(0.2792,0.3610)
2	<i>TEO</i>	N/A	0.3771	0.9167	(0.3242,0.2059)
10	<i>COEFREQ</i>	<i>and</i> detection strategy $d = 1$ sample	0.4581	0.8333	(0.3394,-0.0785)
9	<i>COE</i>	$d = 512$ samples	0.2039	0.9375	(0.3569,0.6993)
11	<i>STECVMAXgmm</i>	30 Gaussian mixture components	0.3799	0.9583	(0.3612,0.2610)
7	<i>COEFREQ</i>	<i>or</i> detection strategy $d = 1$ sample	0.4218	0.8958	(0.3723,0.1468)
14	<i>MFCCgmm</i>	14 Gaussian mixture components	1.0000	1.0000	(0.8231,-0.5210)
13	<i>MFCCgmm</i>	1 Gaussian component	1.0000	1.0000	(0.8752,-0.3597)

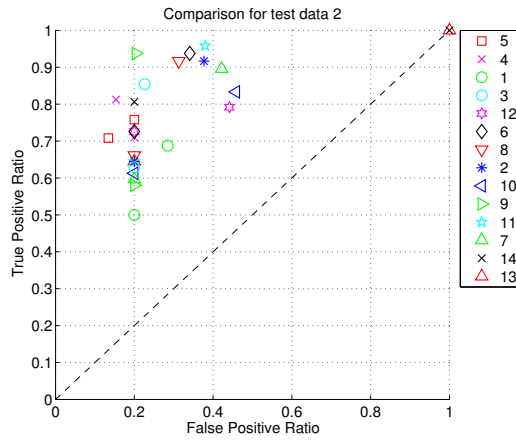


Figure A.2: (FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 2. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.

APPENDIX B

Detection Framework Structure

The framework structure is mostly organized as in figure B.1.

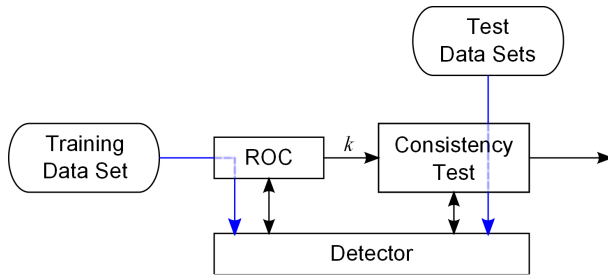


Figure B.1: Framework structure.

In the following sections the built framework is described in terms of scripts. The scripts are organized in packages or directories. Each of the packages and its contents is briefly described.

Table B.1 shows the list of packages in the framework.

Table B.1: List of packages.

Package
Consistency
Features
NoveltyDetectors
ROC
SimpleThresholdDetectors
ROC

B.1 Consistency package

`kTester.m`

Description:

- Definition of data sample indexes that belong to test data set 1.
- Obtention of location of manually detected transient events in test data set 1. This step is important for the calculation of FPR and TPR values.
- Creation of detector instance.
- Obtention of FPR and TPR values for the specific k_{ref} value.
- Display of comparison between detected and manually detected transient events.
- Storage of obtained FPR_{ref} , TPR_{ref} , FPR , TPR k_{ref} and $score$ values for test data set 1.
- Definition of data sample indexes that belong to test data set 1.
- Obtention of location of manually detected transient events in test data set 2. This step is important for the calculation of FPR and TPR values.
- Creation of detector instance.
- Obtention of FPR and TPR values for the specific k_{ref} value.
- Display of comparison between detected and manually detected transient

events.

- Storage of obtained FPR_{ref} , TPR_{ref} , FPR , TPR , k_{ref} and $score$ values for test data set 2.

Dependencies

getIdealEvents.m
NoveltyDetectors package
SimpleThresholdDetectors package
getScore.m

B.2 Features package

filterBankData.m

Description:

- Creation of filter bank.
- Data filtering.

Dependencies

createFilterBank.m

getCOE.m

Description:

- COE feature calculation.

`getLPS.m`

Description:

- LPS feature calculation.

`getRMS.m`

Description:

- RMS feature calculation.

`getSTE.m`

Description:

- STE filter calculation.

`melfiltermatrix.m`

Description:

- Calculation of Mel filter coefficients.

`mfcc.m`

Description:

- MFCC feature calculation.

Dependencies

<code>melfiltermatrix.m</code>

NMF.m

Description:

- NMF feature calculation

B.3 NoveltyDetectors package

CVparzModelGen.m

Description:

- Generation of *CVparz* detector's model.

Dependencies

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m

CVUniDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

hPFilterSignal.m

`CVuniModelGen.m`

Description:

- Generation of *CVuni* detector's model.

Dependencies

<code>getIdealEvents.m</code>
<code>getSignal.m</code>
<code>getTrainingMatrixFromSignal.m</code>

`FREQNMFgmmDetector.m`

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

<code>hPFilterSignal.m</code>
<code>createFilterBank.m</code>
<code>getRMS.m</code>

`FREQNMFgmmModelGen.m`

Description:

- Generation of *FREQNMFgmm* detector's model.

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
createFilterBank.m
getRMS.m
NMF.m

Dependencies

FREQnnDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction and evaluation in neural network.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
createFilterBank.m
getRMS.m

FREQnnModelGen.m

Description:

- Training of *FREQnn* neural network.

Dependencies

MAXparzModelGen.m

Description:

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
createFilterBank.m
getRMS.m

- Generation of *MAXparz* detector's model.

Dependencies

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m

MAXuniDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

hPFilterSignal.m

MAXuniModelGen.m

Description:

- Generation of *MAXuni* detector's model.

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m

Dependencies

MFCCgmmDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
mfcc.m

MFCCgmmModelGen.m

Description:

- Generation of *MFCCgmm* detector's model.

Dependencies

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
mfcc.m

STECVMAXgmmDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
getSTE.m

STECVMAXgmmModelGen.m

Description:

- Generation of *STECVMAXgmm* detector's model.

Dependencies

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
getSTE.m

STEparzModelGen.m

Description:

- Generation of *STEparz* detector's model.

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
getSTE.m

Dependencies

STEuniDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
getSTE.m

STEuniModelGen.m

Description:

- Generation of *STEuni* detector's model.

Dependencies

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
getSTE.m

`TIMENMFgmmDetector.m`

Description:

- Initialization: Obtention of signal to analyse, feature extraction and model evaluation.
- Application of detection strategy.

Dependencies

<code>hPFilterSignal.m</code>
<code>getRMS.m</code>

`TIMENMFgmmModelGen.m`

Description:

- Generation of *TIMENMFgmm* detector's model.

Dependencies

<code>getIdealEvents.m</code>
<code>getSignal.m</code>
<code>getTrainingMatrixFromSignal.m</code>
<code>getRMS.m</code>
<code>NMF.m</code>

`TIMEnnDetector.m`

Description:

- Initialization: Obtention of signal to analyse, feature extraction and evaluation in neural network.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
getRMS.m

TIMEnnModelGen.m

Description:

- Training of *TIMEnn* neural network.

Dependencies

getIdealEvents.m
getSignal.m
getTrainingMatrixFromSignal.m
getRMS.m
NMF.m

B.4 ROC package

ROCgenerator.m

Description:

- Definition of data sample indexes that belong to the training data set.
- Obtention of location of manually detected transient events in the training data set. This step is important for the calculation of *FPR* and *TPR* values.

getIdealEvents.m
NoveltyDetectors package
SimpleThresholdDetectors package

- Creation of detector instance.
- Calculation of k values to use for the generation of the ROC curve. This values are obtained from the detector properties.
- Obtention of FPR and TPR values for the defined k values.
- Calculation of $AUC(0.2)$.
- Display of ROC curve and storage of FPR_{ref} , TPR_{ref} , k_{ref} and $AUC(0.2)$ values.

Dependencies

B.5 SimpleThresholdDetectors package

COEDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
getCOE.m
getRMS.m

COEFREQDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
createFilterBank.m
getCOE.m
getRMS.m

LPSDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
getLPS.m
getRMS.m

TEODetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
getTEO.m
getRMS.m

TEOFREQDetector.m

Description:

- Initialization: Obtention of signal to analyse, feature extraction.
- Application of detection strategy.

Dependencies

hPFilterSignal.m
createFilterBank.m
getTEO.m
getRMS.m

B.6 Utilities package

createFilterBank.m

Description:

- Obtention of filter center frequencies.
- Design of filters.

Dependencies

getValidFrequencies.m

`getEvents.m`

Description:

- Reading of events file. The events file is a list of indexes where each value represents the sample where a transient event begins in a measurement signal.

`getIdealEvents.m`

Description:

- Obtention of indexes where all the transient events happen within a signal.
- Obtention of window number where transient events happen within a specified portion of a signal.

Dependencies

<code>getEvents.m</code>

`getScore.m`

Description:

- Obtention of consistency score.

`getSignal.m`

Description:

- Obtention of a portion of a measurement signal.

Dependencies

`hPFilterSignal.m`

`getTrainingMatrixFromSignal.m`**Description:**

- Obtention of matrix with transient-free data.

`getValidFrequencies.m`**Description:**

- Obtention of center frequencies for one octave and one-third octave band filters.

`hPFilterSignal.m`**Description:**

- Obtention of a portion of a measurement signal.

Bibliography

- [ABSC11] F. Al-Badour, M. Sunar, and L. Cheded. Vibration analysis of rotating machinery using time-frequency analysis and wavelet techniques. *Mechanical Systems and Signal Processing*, 2011.
- [ASSS07] A.R. Abdullah, A.Z. Sha'ameri, A.R.M. Sidek, and M.R. Shaari. Detection and classification of power quality disturbances using time-frequency analysis technique. In *Research and Development, 2007. SCORed 2007. 5th Student Conference on*, pages 1–6. IEEE, 2007.
- [BA02] K.P. Burnham and D.R. Anderson. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Verlag, 2002.
- [Bar07] SV Baranov. Application of the wavelet transform to automatic seismic signal detection. *Izvestiya Physics of the Solid Earth*, 43(2):177–188, 2007.
- [BBL⁺07] M.W. Berry, M. Browne, A.N. Langville, V.P. Pauca, and R.J. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational Statistics & Data Analysis*, 52(1):155–173, 2007.
- [Bel06] C.M. Bishop and SpringerLink (Service en ligne). *Pattern recognition and machine learning*, volume 4. springer New York, 2006.
- [BZ08] I. BEMKE and R. ZIELONKO. Improvement of glass break acoustic signal detection via application of wavelet packet decomposition. *Metrology and Measurement Systems*, 15(4):513–526, 2008.

- [Cam99] D.A. Campbell. Adaptive eeg transient event discrimination using dynamic lms filter weight leakage. In *Signal Processing and Its Applications, 1999. ISSPA'99. Proceedings of the Fifth International Symposium on*, volume 1, pages 359–362. IEEE, 1999.
- [CBT07] D.A. Clifton, P.R. Bannister, and L. Tarassenko. A framework for novelty detection in jet engine vibration data. *Key engineering materials*, 347:305–310, 2007.
- [CDS07] MI Chacon, JL Duran, and LA Santiesteban. A wavelet-fuzzy logic based system to detect and identify electric power disturbances. In *Computational Intelligence in Image and Signal Processing, 2007. CIISP 2007. IEEE Symposium on*, pages 52–57. IEEE, 2007.
- [CHT09] D.A. Clifton, S. Hugueny, and L. Tarassenko. A comparison of approaches to multivariate extreme value theory for novelty detection. In *Statistical Signal Processing, 2009. SSP'09. IEEE/SP 15th Workshop on*, pages 13–16. IEEE, 2009.
- [CHT11] D.A. Clifton, S. Hugueny, and L. Tarassenko. Novelty detection with multivariate extreme value statistics. *Journal of Signal Processing Systems*, 65(3):371–389, 2011.
- [CK10] U.K. Chandrika and J.H. Kim. Development of an algorithm for automatic detection and rating of squeak and rattle events. *Journal of Sound and Vibration*, 329(21):4567–4577, 2010.
- [CO10] G.G. Cabral and A.L.I. Oliveira. A hybrid method for novelty detection in time series based on states transitions and swarm intelligence. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–8. IEEE, 2010.
- [CTS10a] G. Costantini, M. Todisco, and G. Saggio. Musical onset detection by means of non-negative matrix factorization. In *Proceedings of the 14th WSEAS international conference on Systems: part of the 14th WSEAS CSCC multiconference-Volume I*, pages 206–209. World Scientific and Engineering Academy and Society (WSEAS), 2010.
- [CTS10b] G. Costantini, M. Todisco, and G. Saggio. A new method for musical onset detection in polyphonic piano music. In *ICCOMP'10 Proceedings of the 14th WSEAS international conference on computers: part of the 14th WSEAS CSCC multiconference*, 2010.
- [Das10] A. DasGupta. *Fundamentals of Probability: A First Course*. Springer Verlag, 2010.

- [DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.
- [dlRML07] J.J.G. de la Rosa, AM Mufioz, and A. Luque. Power transients characterization and classification using higher-order cumulants and neural networks. In *Instrumentation and Measurement Technology Conference Proceedings, 2007. IMTC 2007. IEEE*, pages 1–4. IEEE, 2007.
- [DSS07] UD Dwivedi, SN Singh, and SC Srivastava. Analysis of transient disturbances in distribution systems: A hybrid approach. In *Power Engineering Society General Meeting, 2007. IEEE*, pages 1–8. IEEE, 2007.
- [Faw04] T. Fawcett. Roc graphs: Notes and practical considerations for researchers. *Machine Learning*, 31:1–38, 2004.
- [FZ07] H. Fastl and E. Zwicker. *Psychoacoustics: facts and models*, volume 22. Springer-Verlag New York Inc, 2007.
- [GBR⁺12] F. Gritti, L. Bocchi, I. Romagnoli, F. Gigliotti, and C. Manfredi. Automatic detection of snore events from full night audio recordings. In *5th European Conference of the International Federation for Medical and Biological Engineering*, pages 183–186. Springer, 2012.
- [GE05] ON Gerek and D.G. Ece. An adaptive statistical method for power quality analysis. *Instrumentation and Measurement, IEEE Transactions on*, 54(1):184–191, 2005.
- [GMES⁺99] S.E. Guttormsson, RJ Marks, MA El-Sharkawi, I. Kerszenbaum, et al. Elliptical novelty grouping for on-line short-turn detection of excited running rotors. *Energy Conversion, IEEE Transactions on*, 14(1):16–22, 1999.
- [HCSZ06] E.B. Halim, M.A.A.S. Choudhury, S.L. Shah, and M.J. Zuo. Fault detection of rotating machinery from bicoherence analysis of vibration data. *power*, 200:400, 2006.
- [HS11] W. Härdle and L. Simar. *Applied multivariate statistical analysis*. Springer Verlag, 2011.
- [HUK⁺07] P. Hayton, S. Utete, D. King, S. King, P. Anuzis, and L. Tarassenko. Static and dynamic novelty detection methods for jet engine health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1851):493–514, 2007.

- [IPH09] H. Iba, T.K. Paul, and Y. Hasegawa. *Applied genetic programming and machine learning*. CRC, 2009.
- [JD96] G.P. John and MG Dimitris. *Digital Signal Processing: Principles, Algorithms and Applications*. Upper Saddle River, New Jersey: Prentice-Hall, 1996.
- [JF00] L.C. Jain and A.M. Fanelli. *Recent advances in artificial neural networks*. CRC Press, 2000.
- [JMG95] N. Japkowicz, C. Myers, and M. Gluck. A novelty detection approach to classification. In *International Joint Conference on Artificial Intelligence*, volume 14, pages 518–523. LAWRENCE ERLBAUM ASSOCIATES LTD, 1995.
- [JPR⁺11] F. Jacobsen, T. Poulsen, J.H. Rindel, A.C. Gade, and M. Ohlrich. *FUNDAMENTALS OF ACOUSTICS AND NOISE CONTROL*. November 2011. Note no 31200.
- [JVB⁺00] V. Jayashankar, R. VanaJaRanjan, K.L. Babu, K.N.P. Kumar, and P.P. Kumar. Time frequency analysis of transient signals. In *TENCON 2000. Proceedings*, volume 1, pages 261–264. IEEE, 2000.
- [Kal11] H.M. Kaltenbach. *A Concise Guide to Statistics*. Springer Verlag, 2011.
- [KOR64] LH Koopmans, DB Owen, and JI Rosenblatt. Confidence intervals for the coefficient of variation for the normal and log normal distributions. *Biometrika*, 51(1/2):25–32, 1964.
- [L⁺00] B. Logan et al. Mel frequency cepstral coefficients for music modeling. In *International Symposium on Music Information Retrieval*, volume 28, page 5, 2000.
- [LS01] D.D. Lee and H.S. Seung. Algorithms for non-negative matrix factorization. 2001.
- [Mal99] S. Mallat. *A wavelet tour of signal processing*. Academic Press, 1999.
- [MBE10] L. Muda, M. Begam, and I. Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques. *Arxiv preprint arXiv:1003.4083*, 2010.
- [Meh08] SK Meher. A novel power quality event classification using slantlet transform and fuzzy logic. In *Power System Technology and IEEE Power India Conference, 2008. POWERCON 2008. Joint International Conference on*, pages 1–4. IEEE, 2008.

- [Men91] J.M. Mendel. Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. *Proceedings of the IEEE*, 79(3):278–305, 1991.
- [Mil10] D. Miljkovic. Review of novelty detection methods. In *MIPRO, 2010 Proceedings of the 33rd International Convention*, pages 593–598. IEEE, 2010.
- [MJ51] F.J. Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, pages 68–78, 1951.
- [MJG10] MAS Masoum, S. Jamali, and N. Ghaffarzadeh. Detection and classification of power quality disturbances using discrete wavelet transform and wavelet networks. *Science, Measurement & Technology, IET*, 4(4):193–205, 2010.
- [MMB11] S.R. Mounce, R.B. Mounce, and J.B. Boxall. Novelty detection for time series data analysis in water distribution systems using support vector machines. *Journal of hydroinformatics*, 13(4):672–686, 2011.
- [MS03a] M. Markou and S. Singh. Novelty detection: a review—part 1: statistical approaches. *Signal Processing*, 83(12):2481–2497, 2003.
- [MS03b] M. Markou and S. Singh. Novelty detection: a review—part 2: neural network based approaches. *Signal Processing*, 83(12):2499–2521, 2003.
- [NPF11] S. Ntalampiras, I. Potamitis, and N. Fakotakis. Probabilistic novelty detection for acoustic surveillance under real-world conditions. *Multimedia, IEEE Transactions on*, (99):1–1, 2011.
- [OP06] P.D. O’grady and B.A. Pearlmutter. Convolutional non-negative matrix factorisation with a sparseness constraint. In *Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on*, pages 427–432. IEEE, 2006.
- [Ots75] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11:285–296, 1975.
- [Pon05] N. H. Pontoppidan. Condition monitoring and management from acoustic emissions. 2005. Supervised by Jan Larsen, IMM, and Torben Fog, MAN B&W Diesel A/S.
- [Rab89] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

- [RAMG10] E. Reyes-Archundia and E.L. Moreno-Goytia. Fault detection in electrical grid on power electronic controllers environment based on wavelet transform. In *Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2010*, pages 574–580. IEEE, 2010.
- [RDR03] MV Ribeiro, SM Deckmann, and JMT Romano. Adaptive filtering, wavelet and lapped transforms for power quality problem detection and identification. In *Industrial Electronics, 2003. ISIE'03. 2003 IEEE International Symposium on*, volume 1, pages 301–306. IEEE, 2003.
- [Sam07] S. Samarasinghe. *Neural networks for applied sciences and engineering: from fundamentals to complex pattern recognition*. Auerbach Publications, 2007.
- [Sil86] B.W. Silverman. *Density estimation for statistics and data analysis*, volume 26. Chapman & Hall/CRC, 1986.
- [Sta05] H.G. Stark. *Wavelets and signal processing: an application-based introduction*. Springer Verlag, 2005.
- [SYT11] A. Subasi, A.S. Yilmaz, and K. Tufan. Detection of generated and measured transient power quality events using teager energy operator. *Energy Conversion and Management*, 52(4):1959–1967, 2011.
- [Tho09] D. Thompson. *Railway noise and vibration: mechanisms, modelling and means of control*. Elsevier Science, 2009.
- [TK08] S. Theodoridis and K. Koutroumbas. *Pattern recognition 4th edn*, volume 1597492728. 2008.
- [TNTC99] L. Tarassenko, A. Nairac, N. Townsend, and P. Cowley. Novelty detection in jet engines. In *Condition Monitoring: Machinery, External Structures and Health (Ref. No. 1999/034), IEE Colloquium on*, pages 4–1. IET, 1999.
- [TR11] R.H.G. Tan and VK Ramachandaramurthy. Real time power quality event detection using continuous wavelet transform. In *Environment and Electrical Engineering (EEEIC), 2011 10th International Conference on*, pages 1–4. IEEE, 2011.
- [Wel86] C. Wellekens. Global connected digit recognition using baum-welch algorithm. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'86.*, volume 11, pages 1081–1084. IEEE, 1986.

- [WLCS06] W. Wang, Y. Luo, JA Chambers, and S. Sanei. Non-negative matrix factorization for note onset detection of audio signals. In *Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on*, pages 447–452. IEEE, 2006.
- [WZ08] C. Weiss and A. Zell. Novelty detection and online learning for vibration-based terrain classification. In *Proceedings of the 10th International Conference on Intelligent Autonomous Systems (IAS 2008)*, pages 16–25. Citeseer, 2008.
- [YC02] D.Y. Yeung and C. Chow. Parzen-window network intrusion detectors. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 4, pages 385–388. IEEE, 2002.
- [Zwi77] E. Zwicker. Procedure for calculating loudness of temporally variable sounds. *The Journal of the Acoustical Society of America*, 62(3):675–682, 1977.

List of Figures

3.1	Flow chart of a detector's internal process.	18
3.2	Example matrices V , W and H	21
3.3	2.5 seconds sample signal taken from a train track noise measurement. Sampling frequency 50000 Hz.	24
3.4	Energy of a signal as a function of samples.	25
3.5	Example of the change of energy feature obtained from a signal. $d =$ 500.	26
3.6	Spectrogram of a 2.5 seconds signal taken from a train track noise measurement.	27
3.7	Example of the LPS feature obtained from a signal.	27
3.8	3 Gaussians in \mathbb{R}^1 with $\mu = 1, 2, 4$ and $\sigma = 0.2, 0.5, 0.4$, respectively (red). GMM of the same Gaussians with mixing coefficients equal to 0.1, 0.4 and 0.5, respectively (blue).	30
3.9	GMM of two Gaussians in \mathbb{R}^1 with $\mu = 2, 4$ and $\sigma = 0.5, 0.4$, respec- tively. Mixing coefficients equal to 0.4 and 0.6, respectively. $k = 0.002$	31

3.10	Auto-encoder with 5 input neurons, 5 output neurons and 3 neurons in the hidden layer.	32
3.11	TEO feature obtained from the filtered signal in figure 3.3. $k = 15$. . .	33
4.1	Schematic of the methodology proposed to evaluate different detection techniques.	35
4.2	Example ROC curve.	36
4.3	Sample signal (upper) and detector output (below).	37
4.4	Process for univariate model fitting.	41
4.5	Process for STE, CV and Maximum features, GMM fitting.	41
4.6	Process for RMS filtered training data features and NMF, GMM fitting.	42
4.7	Process for RMS filtered transient-free training data features and NMF, GMM fitting.	43
4.8	Process for one octave band filtered transient-free training data features, neural network training.	44
4.9	Process for RMS filtered transient-free training data features, neural network training.	45
4.10	Detailed detection process. Novelty detectors.	46
4.11	Detailed detection process for detectors that use NMF as feature. Novelty detectors.	46
4.12	Detailed detection process for simple threshold detectors without frequency band analysis.	47
4.13	Detailed detection process for simple threshold detectors with frequency band analysis.	48
4.14	ROC space with 5 detectors at constant threshold. Redrawn from [Faw04].	49
4.15	Example ROC curve.	50

4.16	Comparison of two detectors.	52
4.17	Sample of two train track noise signals and respective spectrograms.	53
4.18	Speed profile of the train recorded for this project.	54
5.1	ROC curves of detectors with unimodal univariate models. Left column, Gaussian model. Right column, Parzen window estimate.	58
5.2	STE, CV and Maximum feature histograms for transient-free training data set. Gaussian and Parzen window models compared.	59
5.3	STECVMAXgmm detector ROC curves.	60
5.4	MFCCgmm detector ROC curves.	61
5.5	FREQNMFgmm detector ROC curves.	62
5.6	TIMENMFgmm detector ROC curves.	63
5.7	FREQnn detector ROC curves.	64
5.8	Reconstruction error of transient-free training data for NMF and neural network. Feature values as used in the <i>FREQnn</i> detector.	65
5.9	Reconstruction error of transient events in training data for NMF and neural network. Feature values as used in the <i>FREQnn</i> detector.	66
5.10	TIMEnn detector ROC curves.	67
5.11	Reconstruction error of transient-free training data for NMF and neural network. Feature values as used in the <i>TIMEnn</i> detector.	68
5.12	Reconstruction error of transient events in training data for NMF and neural network. Feature values as used in the <i>TIMEnn</i> detector.	69
5.13	<i>TEO</i> detector ROC curve.	70
5.14	<i>TEOFREQ</i> detector ROC curves.	71
5.15	COE detector ROC curves.	71

5.16	Noise track sample data containing transient events. COE feature at different d values.	72
5.17	<i>COEFREQ</i> detector ROC curves.	73
5.18	<i>LPS</i> detector ROC curves.	74
5.19	(FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 1 for the first 5 detectors according to table 5.2. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.	78
5.20	(FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 2 for the first 5 detectors according to table 5.3. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.	80
A.1	(FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 1. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.	91
A.2	(FPR_{ref}, TPR_{ref}) and (FPR, TPR) points for consistency test in test data set 2. All points on the $FPR_{ref} = 0.2$ represent (FPR_{ref}, TPR_{ref}) points.	93
B.1	Framework structure.	95

List of Tables

3.1	One-third octave and octave band (bold) center frequencies (Hz).	19
4.1	List of novelty detectors, features and models.	39
4.2	List of novelty detectors and chosen parameters.	40
4.3	List of simple threshold detectors, features and parameters. . . .	47
4.4	Confusion matrix. Redrawn from [Faw04].	49
5.1	List of promising detectors, parameters and $AUC(0.2)$, in $AUC(0.2)$ descending order.	75
5.2	List of promising detectors. Results of consistency test in test data set 1, in <i>score</i> descending order.	77
5.3	List of promising detectors. Results of consistency test in test data set 2, in <i>score</i> descending order.	79
A.1	List of detectors, parameters and ROC relevant information, in $AUC(0.2)$ descending order.	87

A.2 List of 14 promising detectors, parameters and consistency test
relevant information, in *score* descending order. Test data set 1. 90

A.3 List of 14 promising detectors, parameters and consistency test
relevant information, in *score* descending order. Test data set 2. 91

B.1 List of packages. 96

List of Abbreviations

AIC: Akaike's Information Criterion

AUC: Area Under the Curve

COE: Change Of Energy

COE: Detector using the change of energy feature

COEFREQ: Detector using the change of energy feature in frequency bands

CV: Coefficient of Variance

CVuni: Detector using Coefficient of Variance as feature, and a uni-modal uni-variate Gaussian as model

CVparz: Detector using Coefficient of Variance as feature, and Parzen window estimate as model

CWT: Continuous Wavelet Transform

DWT: Discrete Wavelet Transform

EGG: Electroencephalogram

EVT: Extreme Value Theory

FN: False Negative

FP: False Positive

FPR: False Positive Ratio

FREQnn: Detector using frequency information as features, with a neural network training

FREQNMFgmm: Detector using frequency information and Non-Negative Matrix factorization to generate features, with a Gaussian Mixture Model

GMM: Gaussian Mixture Model

HMM: Hidden Markov Models

KDE: Kernel Density Estimation

LMS: Least Mean Square

LPS: Logarithmic Power Spectrum

LPS: Detector using the Logarithmic Power Spectrum feature

MAXuni: Detector using the maximum as feature, and a uni-modal uni-variate Gaussian as model

MAXparz: Detector using the maximum as feature, and Parzen window estimate as model

MFCC: Mel Frequency Cepstral Coefficients

MFCCgmm: Detector using Mel Frequency Cepstral Coefficients as feature, with a Gaussian Mixture Model

MLT: Modulated Lapped Transform

NMF: Non-Negative Matrix Factorization

PQ: Power Quality

PTL: Perceived Transient Loudness

RLS: Recursive Least Square

RMS: Root Mean Square

ROC: Receiver Operating Characteristic

SPL: Sound Pressure Level

STA/LTA: Short Time Average to Long Time Average

STE: Short Term Energy

STECVMAXgmm: Detector using Short Term Energy, Coefficient of Variance and maximum as features, with a Gaussian Mixture Model

STEuni: Detector using Short Term Energy as feature, and a uni-modal univariate Gaussian as model

STeparz: Detector using Short Term Energy as feature, and Parzen window estimate as model.

STFT: Short Time Fourier Transform

SVM: Support Vector Machine

SVR: Support Vector Regression

TEO: Teager Energy Operator

TEO: Detector using Teager Energy Operator as feature

TEOFREQ: Detector using the Teager Energy Operator as feature in frequency bands

TIMENN: Detector using time series information as features, with a neural network training

TIMENMFgmm: Detector using time series information and Non-Negative Matrix factorization to generate features, with a Gaussian Mixture Model

TN: True Negative

TP: True Positive

TPR: True Positive Ratio

WPT: Wavelet Packet Transform