TIME DOMAIN CLASSIFICATION OF TRANSIENT RFI

Daniel Josef Czech

Thesis Presented for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical Engineering

UNIVERSITY OF CAPE TOWN

February 2019

Declaration

This is my own unaided work. All referenced work has been cited. I have not allowed, nor will I allow my work to be copied by anyone else with the intention of claiming it as their own. It has not been submitted previously for examination at any other university.

Signed by candidate

Cape Town

February 4, 2019

Abstract

Since the emergence of radio astronomy as a field, it has been afflicted by radio frequency interference (RFI). RFI continues to present a problem despite increasingly sophisticated countermeasures developed over the decades. Due to technological improvements, radio telescopes have become more sensitive (for example, MeerKAT's L-band receiver). Existing RFI has become more prominent as a result. At the same time, the prevalence of RFI-generating devices has increased as new technologies have been adopted by society.

Many approaches have been developed for mitigating RFI, which are typically used in concert. New telescope arrays are often built far from human habitation in radio-quiet reserves. In South Africa, a radio-quiet reserve has been established in which several world class instruments are under construction. Despite the remote location of the reserve, careful attention is paid to the possibility of RFI. For example, some instruments will begin observations while others are still under construction. The infrastructure and equipment related to the construction work may increase the risk of RFI, especially transient RFI.

A number of mitigation strategies have been employed, including the use of fixed and mobile RFI monitoring stations. Such stations operate independently of the main telescope arrays and continuously monitor a wide bandwidth in all directions. They are capable of recording spectra and high resolution time domain captures of transient RFI. Once detected, and if identified, an RFI source can be found and dealt with. The ability to identify the sources of detected RFI would be highly beneficial.

Continuous wave intentional transmissions (telecommunication signals for example) are easily identified as they are required to adhere to allocated frequency bands. Transient RFI signals, however, are significantly more challenging to

identify since they are generally broadband and highly intermittent. Transient RFI can be generated as a by-product of the normal operation of devices such as relays, AC machines and fluorescent lights, for example. Such devices may be present near radio telescope arrays as part of the infrastructure or equipment involved in the construction of new instruments. Other than contaminating observation data, transient RFI can also appear to have genuine astronomical origins. In one case, transient signals received from a microwave oven exhibited dispersion, suggesting a distant source. Therefore, the ability to identify transient RFI by source would be enormously valuable. Once identified, such sources may be removed or replaced where possible. Despite this need, there is a paucity of work on classifying transient RFI in the literature. This thesis focusses on the problem of identifying transient RFI by source in time domain data of the type captured by remote monitoring stations. Several novel approaches are explored in this thesis. If used with independent RFI monitoring stations, these approaches may aid in tracking down nearby RFI sources at a radio telescope array.

They may also be useful for improving RFI flagging in data from radio telescopes themselves. Distinguishing between transient RFI and natural astronomical signals is likely to be an easier prospect than classifying transient RFI by source. Furthermore, these approaches may be better able to avoid excising genuine astronomical transients that nevertheless share some characteristics with RFI signals. The radio telescopes themselves are significantly more sensitive than RFI monitoring stations, and would thus be able to detect RFI sources more easily. However, terrestrial RFI would likely enter via sidelobes, tempering this advantage somewhat.

In this thesis, transient RFI is first characterised, prior to classification by source. Labelled time-domain recordings of a number of transient RFI sources are acquired and statistically examined. Second, components analysis techniques are considered for feature selection. Cluster separation is analysed for principal components analysis (PCA) and kernel PCA, the latter proving most suitable. The effect of the supply voltage of certain RFI sources on cluster separation in the principal components domain is also explored. Several naïve classification algorithms are tested, using kernel PCA for feature selection.

A more sophisticated dictionary-based approach is developed next. While there are variations in repeated recordings of the same RFI source, the signals tend to adhere to a common overarching structure. Full RFI signals are observed to consist of sequences of individual transients. An algorithm is presented to extract individual transients from full recordings, after which they are labelled using unsupervised clustering methods. This procedure results in a dictionary of archetypal transients, from which any full RFI sequence may be represented. Some approaches in Automated Speech Recognition (ASR) are similar: spoken words are divided into individual labelled phonemes. Representing RFI signals as sequences enables the use of hidden Markov models (HMMs) for identification. HMMs are well suited to sequence identification problems, and are known for their robustness to variation. For example, in ASR, HMMs are able to handle the variations in repeated utterances of the same word. When classifying the recorded RFI signals, good accuracy is achieved, improving on the results obtained using the more naïve methods.

Finally, a strategy involving deep learning techniques is explored. Recurrent neural networks and convolutional neural networks (CNNs) have shown great promise in a wide variety of classification tasks. Here, a model is developed that includes a pre-trained CNN layer followed by a bidirectional long short-term memory (BLSTM) layer. Special attention is paid to mitigating class imbalance when the model is used with individual transients extracted from full recordings. High classification accuracy is achieved, improving on the dictionary-based approach and the other naïve methods.

Recommendations are made for future work on developing these approaches further for practical use with remote monitoring stations. Other possibilities for future research are also discussed, including testing the robustness of the proposed approaches. They may also prove useful for RFI excision in observation data from radio telescopes.

Acknowledgements

I would like to acknowledge the following people and organisations:

Prof. Mike Inggs, who agreed to supervise my proposed project and encouraged me from the start to pursue a PhD. Thank you for your expert help and guidance over the years, even after retiring.

Dr Amit Mishra for his co-supervision and for encouraging me to upgrade to a PhD; it is much appreciated.

Dr Jason Manley, for his valuable advice and insightful support, given freely since I was an undergraduate student. Thank you for meeting with me on numerous occasions to discuss my project.

The engineers at SARAO (formerly SKA SA) who have helped me: Mr Carel van der Merwe, Mr Christopher Schollar, Dr Gideon Wiid and Mr Simon Norval. Also Monika Obrocka and Maciej Serylak, for helping me find pulsar data.

Prof. Daniel O'Hagan for his advice, especially when writing my first conference paper, and Prof. Fred Nicolls, for always having time for a discussion.

Members (past and present) of the Radar Remote Sensing Group: Mr Jarryd Son, for answering my questions about Keras. Mr James Gowans, for lending me the ROACH board he was using for his masters project and helping me set it up to capture RFI. Mr Darryn Jordan, for helping me to record FMCW radar pulses from MiloSAR. Thanks also to Dr Craig Tong, Dr Francois Schonken, Mr Stephen Paine, Mr Po Kai Cheng and Dr Abhishek Bhatta.

SARAO for providing me with funding to support my postgraduate studies and for enabling me to attend and present my work at stimulating local and international conferences.

The University of Cape Town's ICTS High Performance Computing team for making their high-memory machines available to me.

The anonymous reviewers of my publications, for volunteering their time.

The many contributors to the free and open source projects I have used - the true democratisers of knowledge.

My parents, Anthea and Konrad, and my siblings, Josh and Sasha. The countless ways in which you have supported me during my time as a student would fill too many pages to include here.

My friends, for devising timeous distractions.

Contents

List of Figures

List of Tables

List of Abbreviations

ADC Analogue to Digital Converter. 21, 33, 34, 68

ANN Artificial Neural Network. 20

ASKAP Australian Square Kilometre Array Pathfinder. 21

ASR Automated Speech Recognition. iii, 66, 85, 88

BLSTM Bidirectional Long Short-Term Memory. iv, 14, 15, 88, 91, 92, 93

CASPER Collaboration for Astronomy Signal Processing and Electronics Research. 33

C-BASS C-Band All Sky Survey. 3

CDF Cumulative Distribution Function. 40

CFL Compact Fluorescent Lamp. xiv, 58, 66, 78, 80, 83, 86, 93, 98

CNN Convolutional Neural Network. iv, xiv, xvi, 15, 85, 86, 88, 90, 91, 92, 93, 94, 93, 97, 99

CUSUM Cumulative Sum. 19

CW Continuous Wave. 3, 4, 5, 18, 28, 99

DBSCAN Density-Based Spatial Clustering of Applications with Noise. 76, 82, 84

DM Dispersion Measure. 38

FMCW Frequency Modulated Continuous Wave. v, 38

FRB Fast Radio Burst. 2, 17, 20, 37, 100

GiB Gibibyte (2^{30} bytes). 91

GMM Gaussian Mixture Model. 10, 44

GPU Graphics Processing Unit. 91

GRU Gated Recurrent Unit. 84

GSa/s Gigasamples Per Second. 21, 29, 33

GSI Geometric Separability Index. 52, 53

HERA Hydrogen Epoch of Reionisation Array. 2, 3, 9, 34

HIRAX Hydrogen Intensity and Real-Time Analysis Experiment. 3

HMM Hidden Markov Model. iii, xiv, xxi, 14, 15, 65, 66, 78, 79, 80, 82, 84, 100

KAT-7 Karoo Array Telescope 7. 3

kNN k-Nearest Neighbours. 7, 15, 26, 53, 58, 63, 97

LNA Low Noise Amplifier. 36

LOFAR Low-Frequency Array. 20

LPDA Log-Periodic Dipole Array. 30, 34

LSTM Long Short-Term Memory. xvi, 84, 86, 88, 93, 94, 97

MeerKAT Meer Karoo Array Telescope ("Meer" translates to "More"). ii, xii, 1, 2, 3, 4, 8, 10, 21, 27, 28, 29, 31, 32, 47, 99

MERLIN Multi-Element Radio Linked Interferometer Network. 20

MWA Murchison Widefield Array. 20

NIALM Non-Intrusive Appliance Load Monitoring. 26

PAPER Precision Array for Probing the Epoch of Reionization. 2, 3, 34

PCA Principal Components Analysis. iii, xii, xiii, xiv, xvi, xx, 11, 12, 11, 12, 13, 26, 39, 49, 50, 51, 54, 55, 54, 55, 58, 59, 61, 63, 66, 74, 75, 84, 97, 99

PLC Power Line Communication. 25

PSU Power Supply Unit. xiii, 59, 78, 83, 93

RATTY Real Time Transient Analyser. xii, 4, 21

RBF Radial Basis Function. xiii, 54

RFF Radio Frequency Fingerprinting. 27

RNN Recurrent Neural Network. 85

ROACH Reconfigurable Open Computing Hardware. v, 9, 21, 33

SARAO South African Radio Astronomy Observatory, formerly SKA SA (Square Kilometre Array South Africa). v, xviii, 21, 29, 33

SCR Silicon Controlled Rectifier. 25

SERPent Scripted e-MERLIN RFI-Mitigation Pipeline for Interferometry. 20

SKA Square Kilometre Array. 2, 3

SKA SA Square Kilometre Array South Africa, now SARAO (the South African Radio Astronomy Observatory). v, xviii, 21

SPRT Sequential Probability Ratio Test. 19

SVM Support Vector Machine. xvi, xx, 7, 14, 15, 58, 59, 63, 88, 91, 92, 93, 97

List of Symbols

$\boldsymbol{A}$	State transition matrix. 50, 51, 79, 80
$\boldsymbol{A_r}$	A matrix of raw signals on which to perform PCA. 50
$\widetilde{\boldsymbol{A}}_{\boldsymbol{r}}$	The same as $\boldsymbol{A_r}$ but with the mean signal subtracted from each raw signal in $\boldsymbol{A_r}$. 50, 51
$\boldsymbol{B}$	The matrix of emission probabilities. 79
$\boldsymbol{C}$	The covariance matrix (typically for PCA and kernel PCA). 50, 51, 52
$\widetilde{\boldsymbol{C}}$	The alternative to $\boldsymbol{C}$, given as $\widetilde{\boldsymbol{C}} = \widetilde{\boldsymbol{A}}_{\boldsymbol{r}}{}^{T} \widetilde{\boldsymbol{A}}_{\boldsymbol{r}}$. 50
C_r	Regularisation parameter for a linear SVM. 92
D	Kolmogorov-Smirnov test statistic. 44, 47
$\boldsymbol{E}$	Eigenvectors of a matrix. 50, 51, 52
F_1	Combined F_1 score (see Section 6.4). 93
$\boldsymbol{G}$	A vector, each element of which is the number of instances (individual examples) in each class i. 91
G_i	The number of instances (individual examples) in class i. 91
H	The Hilbert transform. 70
$\boldsymbol{K}$	A kernel function. 51, 52

$\widetilde{\boldsymbol{K}}$	The Gram matrix. 52
L_1	The length of a moving window for smoothing a signal by convolution. xiv, 68, 70, 72, 74
M	The number of classes to be classified. 91, 92, 93
N	The number of instances (individual examples) in a particular training batch. 91
$\mathtt{P}_3$	3^{rd} order polynomial kernel function. 54
$\mathtt{P}_9$	9^{th} order polynomial kernel function. 54
$\boldsymbol{Q}$	A sequence of states. 80
$\boldsymbol{R}$	A set of recordings of full RFI events. 66, 67
s_s	The silhouette coefficient. 54
T_1	The first (more lenient) transient extraction threshold. 68, 70
T_2	The second (stricter) transient extraction threshold. 70
T_M	The merging threshold. Adjacent regions separated by fewer time steps are merged. xiv, 68, 70, 72, 74
V	Voltage. xiv, 61, 63, 97
V_{peak}	Peak voltage of the mains AC supply. xiv, 61, 63, 64, 97
$\boldsymbol{X}$	Principal components matrix, of shape (instances, components). 75
$\boldsymbol{X'}$	A matrix of principal components after scaling by eigenvalue as described in Section 5.2.2. 75
$\boldsymbol{\alpha}$	A vector of the individual coefficients α_i to α_l. 52
α	A single coefficient of a feature-mapped input vector (instance) to kernel PCA. 52

γ A parameter for different kernel functions (see Table 4.1). xiii, 54, 55, 58, 75

κ Cohen's Kappa. 53

$\mathcal{L}$ Loss function for use in backpropagation. 91, 92

$\boldsymbol{\lambda}$ The vector of eigenvalues corresponding with the amount of variance explained by each principal component. 50, 51, 52, 75

μ Mean. 67, 87

$\boldsymbol{\pi}$ The initial state distribution or elements thereof. xiv, 79, 80

σ Standard deviation. 67, 87

ϕ A feature map for kernel PCA. 51, 52

a The average distance from a particular instance to all the other instances of the same class. 54

a_{ij} The probability that state q_{i+1} will be s_j given that state q_i is s_i. 79, 80

b The average distance from the instance to all the instances belonging to the nearest different class. 54

b_{ij} The probability that state s_j will produce observable emission v_i. 79

$\boldsymbol{c}$ A vector of class weights (for mitigating class imbalance). 91

c_i Weight for class i (for mitigating class imbalance). 91, 92

d The dimension of an input vector. If the vector is an RFI recording $\boldsymbol{r}$, then d is the same as its length. 50, 51

$\texttt{fp}_i$ The number of false positives for class i. A false positive occurs when an instance is incorrectly labelled as belonging to the class in question. 93

$\texttt{f\_mask}$	The mask produced after the application of the first (lenient) threshold. 70
$\texttt{fn}_i$	The number of false negatives for class i. A false negative occurs when an instance is incorrectly labelled as not belonging to the class in question. 93
g	A region extracted from a full RFI event using T_1, the lenient threshold. 70
$g_{\texttt{sub}}$	A subregion in an extracted parent region g. 70
k	The number of nearest neighbours to a particular point. 11, 52, 53, 58, 79, 82
l	The number of input instances (individual examples). 50, 51, 52, 53
m	The number of observable emissions for each HMM. 79, 82
n	The number of hidden states in each HMM. 79
o	The weighted sum of activations in the hidden layers. 92
p	Probability value (asymptotic significance). 40, 47
p_0	The agreement between two raters (equivalent to accuracy). 53
p_e	The probability of agreement by chance between two raters. 53
q	A single state in a sequence of states. 80
$\boldsymbol{r}$	A single recording of a full RFI event, consisting of a sequence of transients. 66, 67, 70
$\boldsymbol{r'}$	A single recording $\boldsymbol{r}$ that has been standardised within its class. 67
$\boldsymbol{r_g}$	A region extracted from a full RFI event that has been smoothed using L_1. 70
$\boldsymbol{r_H}$	The result of applying the Hilbert transform to the absolute value of an RFI event. 70

`r_mask`	The mask (covering only an extracted region) produced after the application of the second (strict) threshold to $\boldsymbol{r_s}$. 70
$\boldsymbol{r_s}$	The resultant signal after smoothing. 70
s	A hidden state. 79, 80
$\boldsymbol{t}$	A class label. 52, 53
$\mathtt{tp}_i$	The number of true positives for class i. A true positive occurs when an instance is correctly labelled as belonging to the class in question. 93
$\boldsymbol{u}$	An individual time-domain transient signal, extracted from a recording $\boldsymbol{r}(t)$ containing a sequence of such transients. 70, 87
$\boldsymbol{u'}$	An individual transient signal scaled so that its amplitude ranges from -1 to 1. 87
v	An observable emission. 79
x	An individual input vector (an instance) of dimension d. 51, 52, 53
$\boldsymbol{x_u}$	A feature vector, each element of which is a single value from an instance $\boldsymbol{u}$ at a particular time step. 87
$\boldsymbol{x_u}'$	A standardised feature vector. 87
$\boldsymbol{y}$	1-hot encoded vector containing the ground truth. 91
y_i	Element i (corresponding with class i) of the 1-hot encoded ground truth vector. 92
y_{ij}	Element i (corresponding with class i) of the 1-hot encoded ground truth vector for the j^{th} instance. 91
$\boldsymbol{\hat{y}}$	The output (1-hot encoded) of the final layer in a neural network after the application of activation functions. 91

$\hat{y}_i$ The activation output at the i^{th} final nodes (one per class) of the neural network. 91, 92

$\hat{y}_{ij}$ Element i (corresponding with class i) of the output vector for the j^{th} input instance. 91

Chapter 1

Introduction

The very first observed astronomical radio source was only discovered because it was an inconvenient source of interference in early trans-Atlantic voice transmissions. Nowadays, the roles are reversed; more often than not, radio astronomy finds itself a victim of interference due to such telecommunications transmissions themselves. However, it was not until we had generated what would later be considered radio frequency interference (RFI), that the first successful radio astronomy observation was actually made. In a sense, the history of RFI in radio astronomy is as old as the field itself.

In recent decades, mitigating RFI has become increasingly important for two reasons. First, as technology used in the construction of radio astronomy instruments has improved, their sensitivity has increased. For example, the MeerKAT radio telescope is likely to achieve unprecedented sensitivity [1, 2]. This increase in sensitivity means that even very weak or distant sources of RFI may harm observations. Second, new technologies that actively use the RF spectrum are being developed and adopted in society (for example, collision avoidance radars in cars, or mobile broadband communications). Thus, the number of potential sources of RFI is generally increasing. As a result of these changes, efforts to mitigate all forms of RFI are becoming increasingly important.

Mitigating RFI has generally been accomplished by avoiding or excising unwanted signals. Such an approach is effective for powerful continuous transmissions, which have been among the most destructive of RFI sources. However, as transient radio astronomy has risen in prominence, so too has the problem of

transient RFI. Transient RFI signals are more difficult to classify than continuous transmissions, as they are typically broadband, intermittent and generated unintentionally (and thus do not adhere to allocated frequency bands). However, it has been historically less prevalent, less powerful and thus less damaging than other RFI. Nevertheless, it presents its own set of subtle problems.

For example, transient signals (dubbed Perytons) have been detected with significant frequency dispersion, as would be exhibited by an extragalactic source [3]. Such transient signals are of particular interest because of their similarity to Fast Radio Bursts (FRBs), a type of transient extragalactic signal with an as yet undetermined origin. While FRBs do seem to be legitimate astronomical sources, the source of the Perytons was unmasked as a nearby microwave oven by an enterprising graduate student in 2015 [4]. However, as a result of the Perytons, for a time some doubt was cast on the extragalactic origins of the original Lorimer burst (the first FRB to be detected)[3]. Incidents such as this one underscore the unmet need for an ability to identify the sources of transient RFI. The work in this thesis mounts an attempt to meet this need, at least in part.

1.1 Radio Astronomy in South Africa

Considerable resources have been invested in the establishment of radio astronomy facilities in South Africa. In 2012 it was announced that both South Africa and Australia would jointly host the Square Kilometre Array (SKA). Prior to this, a South African demonstrator array of 7 dishes had already been constructed, and work had begun on its successor, a 64 dish interferometer called MeerKAT [2, 5]. Ultimately, MeerKAT will be integrated into the SKA itself. In 2007 the Astronomy Geographic Advantage (AGA) Act [6] was passed, which provides legal protection for radio-quiet Astronomy Advantage Areas in South Africa. The MeerKAT core site is located in one such area, to which a number of other ongoing projects have been attracted. As of writing, these include:

1. MeerKAT [2, 5], South Africa's 64-dish SKA pathfinder that will eventually be integrated into the SKA proper.

2. The Hydrogen Epoch of Reionisation Array (HERA) [7], the successor to PAPER [8], a dense low-frequency array looking at the epoch of reionisation.

3. The southern component of the C-Band All Sky Survey (C-BASS) [9], which will observe the cosmic microwave background.

4. The Hydrogen Intensity and Real-Time Analysis Experiment (HIRAX) [10], a planned array observing at 400 - 800 MHz.

5. SKA-1 mid-frequency array (130 dishes, to be integrated with MeerKAT).

Prior instruments include KAT-7 (the demonstrator for MeerKAT) [11] and PAPER, the predecessor of HERA. The multitude of instruments is exciting and promises a fascinating future for radio astronomers and engineers in South Africa. Such rapid development brings potential side-effects with it, however. As the various radio telescopes are being constructed, others will be conducting scientific observations. The construction work itself and other related infrastructure will likely increase the risk of RFI to instruments that have already begun observing. One approach to mitigating this risk is to continuously monitor the RF environment for sources of RFI, and in the ideal case, identify them so that they may be removed or replaced.

At the MeerKAT site, wide-band monitoring systems have been installed, both fixed and mobile. They are capable of monitoring very wide frequency ranges simultaneously, and have both time-frequency and instantaneous time domain capturing modes. It will become possible to efficiently monitor extremely wide frequency ranges around the MeerKAT site. The challenge then lies in tracking down the sources of received RFI, so that they may be dealt with. Several avenues are available for source identification: physical location, high-level attributes for CW signals (frequency band, for example), and low-level features such as frequency or amplitude distribution. This thesis focusses on the use of low-level features (such as the time-domain shape of a transient signal) to identify the sources of transient RFI.

1.2 Transient RFI

Most prior RFI mitigation strategies have been applied in the time-frequency domain. In the time-frequency domain, it is often easy to identify continuous transmissions (such as telecommunication signals) because they adhere to government-allocated frequency bands. However, transient RFI poses more of a problem. In time-frequency plots, transient RFI typically presents as a brief, broadband streak. In Fig. 1.1, examples of both continuous transmissions and transient RFI signals are displayed. Besides the fact that such emissions (by their nature) are not limited to predefined bands, these other characteristics render them difficult to identify. First, information on their frequency distribution is lost due to channelisation. Second, some of these events are so short that they are averaged away and lost in the integration period of the receiver back-end.

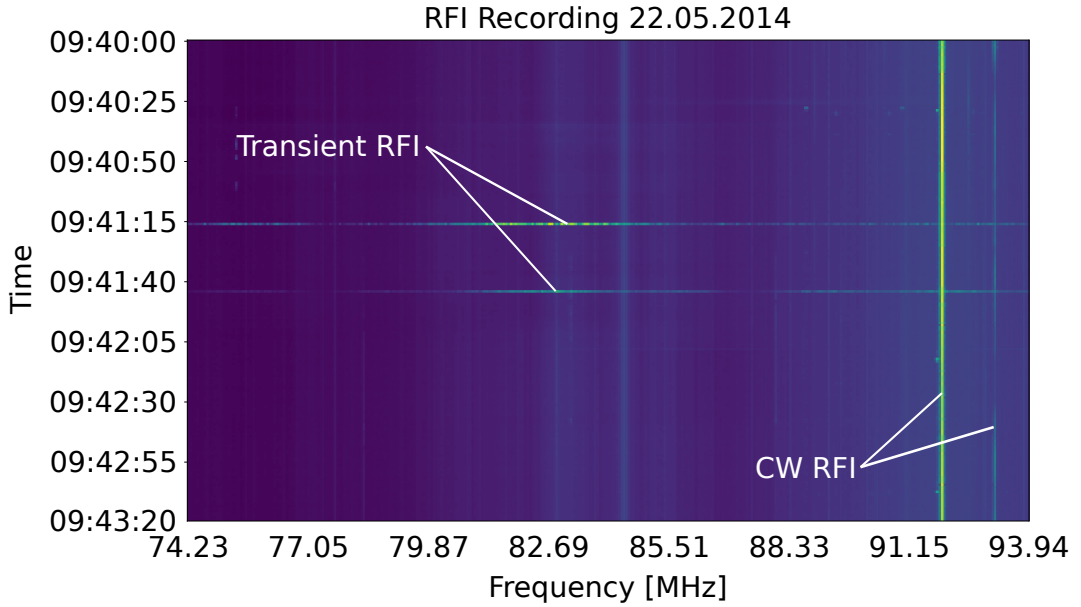

Figure 1.1: A waterfall plot generated from spectra recorded by an early RFI monitoring system (RATTY) installed at the MeerKAT site (see Section 2.1.7). Both CW and transient RFI signals are illustrated in this portion of a full recording. Each spectrum, given by a row of pixels, was recorded after integrating for a second.

Transient RFI has many sources, both human-generated and natural (atmospheric phenomena like lightning, for example). This thesis deals only with human-generated transient RFI. This RFI is most often the unintended release of

RF energy as a byproduct of the normal operation of devices such as mechanical relays, motors and fluorescent lighting. Work on identifying sources of this type of transient RFI is scarce in the open literature. As such, the aim of this thesis is to help fill this gap.

1.3 Problem Description

Nearly all prior work on RFI mitigation has dealt with time-frequency data directly from radio telescopes. CW signals (used for telecommunications, for example) have generally been considered the most important class of RFI, due to their power and near-continuous presence.

However, as discussed in prior sections, modern radio telescopes are becoming increasingly sensitive while astronomical radio transients (such as fast radio bursts) have become an important area of study. In addition, in radio-quiet reserves, it is likely that some telescopes will operate while others are under construction, heightening the risk of transient RFI. There is thus a pressing need for further investigation into transient RFI.

The ability to determine the sources of detected transient RFI would be highly valuable. Once an RFI source is known, it can be mitigated (by removal or replacement, for example). Furthermore, the risk of an RFI transient being mistaken for an astronomical source is diminished.

There is a paucity of work on the statistical characterisation and classification of transient RFI in the open literature. It is not a trivial problem, since such RFI is broadband, often extremely brief, and intermittent.

1.4 Objectives

The intent of this thesis is to help address the gap in research on RFI classification in a radio astronomy context. The aim is to characterise transient RFI and demonstrate methods of identifying it by its source. Transient RFI sources are identified using the type of data that is recorded by remote monitoring stations in radio-quiet reserves. The different approaches developed and evaluated in this

thesis form a strong basis for possible future RFI classification and identification systems.

1.5 Research Hypothesis

Sources of human-generated transient RFI can be identified by their RF emissions alone.

The following research questions are developed from this hypothesis:

1. What are the statistical characteristics (pertinent to source identification) of human-generated transient RFI?
2. Are components analysis methods good feature selection approaches for transient RFI classification? Do any extraneous factors affect cluster separation in the principal components domain?
3. Is it possible to build a canonical dictionary of transients, with which any full RFI event may be represented? Can hidden Markov models be used in conjunction with such a dictionary to identify transient RFI?
4. Are deep learning techniques, which have in recent years proven highly successful in many classification tasks, suitable for identifying transient RFI?

1.6 Statement of Originality

The following contributions are believed by the author to constitute original research:

1. The first published dataset of labelled time-domain recordings of transient RFI.
2. The first published analysis of this type of transient RFI at each time-step across recordings, including attempts at distribution fitting.

3. The first investigation into the application of components analysis techniques for feature selection on a dataset of transient RFI.

4. An evaluation of SVM and kNN classification techniques on full-length transient RFI events after feature selection using components analysis techniques.

5. An investigation into the relationship between the supply voltage phase of RFI sources and class separation in the principal components domain.

6. The development of a dictionary-based approach to transient RFI classification, including labelling and sequence representation and the application of hidden Markov models for source identification.

7. The first application of deep learning techniques (including convolutional neural networks and long short-term memory units) to the problem of classifying transient RFI by source.

1.7 List of Publications

Journal papers:

1. Czech, Daniel, Amit Mishra, and Michael Inggs. "A CNN and LSTM-Based Approach to Classifying Transient Radio Frequency Interference." Astronomy and Computing 25 (2018): 52-57. doi:0.1016/j.ascom.2018.07.002

2. Czech, Daniel, Amit Mishra, and Michael Inggs. "A dictionary approach to identifying transient RFI." Radio Science 53 (2018): 656-669. doi:10.1029/2018RS006538

3. Czech, Daniel, Amit Mishra, and Michael Inggs. "Characterizing transient radio-frequency interference." Radio Science 52 (2017): 841-851. doi:10.1002/2016RS006227

Conference papers:

1. Czech, Daniel, Amit Kumar Mishra, and Michael Inggs. "Distinguishing between pulsars and transient RFI in the time domain," 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 1236-1239. IEEE, 2017.

2. Czech, Daniel, Amit Kumar Mishra, and Michael Inggs. "A canonical interferencelet-based approach to RFI identification." Radio Frequency Interference (RFI), 16-20. IEEE, 2016.

3. Czech, Daniel, Amit Kumar Mishra, and Michael Inggs. "Identifying radio frequency interference with hidden Markov models." Radio Frequency Interference (RFI), 21-25. IEEE, 2016.

4. Czech, Daniel, Amit Kumar Mishra, and Michael Inggs. "Time domain classification of transient radio frequency interference." 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 302-305. IEEE, 2016.

1.8 Overview

The main findings of each chapter are discussed briefly in this section.

1.8.1 Literature

This chapter consists of a thorough review of the literature deemed pertinent to this thesis. In general, literature on the classification and identification of RFI, especially transient RFI, is relatively scarce. Fortunately, there are other fields that deal with the classification of similar transient signals. Therefore, selected work from these other fields is also reviewed since it provides valuable insights into the current problem. The chapter begins with an introduction to the potential dangers of transient RFI. The sections that follow deal with the set of important topics relevant to the problem at hand.

RFI detection can be considered a simple form of RFI classification - the binary decision whether a signal is RFI or not. General RFI detection techniques are covered, from simple statistical methods like the median absolute deviation to recent deep learning-based approaches. Next, remote monitoring stations are covered. Such systems continuously monitor a wide bandwidth independently of radio telescope arrays. A number of remote monitoring systems have been developed for MeerKAT in South Africa.

Work on characterising and modelling transient RFI is also reviewed. Much of the literature is concerned with modelling the amplitude distribution of transient signals, in a variety of related fields. A few papers assess high-level attributes like pulse inter-arrival time, while none attempt to model the statistics of repeated transient events from the same source. While not strictly RF in nature, the interfering signals studied in the field of power line communications are often very similar to typical transient RFI. Thus, some insights may also be drawn from such publications.

Finally, efforts to classify RFI by its source are considered. Due to the paucity of research in a radio-astronomy context, transient classification approaches in other fields are also reviewed. In a broader RF context, some relatively isolated experiments have been conducted on RF emission classification. Other related fields include bio-acoustics; a number of publications deal with classifying animals by species or even individually using their calls. These are particularly useful in Chapter 5, in which a dictionary approach to identification is proposed.

1.8.2 Characterising Transient RFI

This chapter examines human-generated transient RFI in close detail. The aim is to analyse the statistics (pertinent to source classification) of transient RFI. Importantly, the distribution at each time-step across many transient signals from the same source is investigated. The findings have implications for the choice of feature selection techniques in subsequent chapters.

Suitable datasets were needed before conducting any statistical analyses. The largest of these consists of nine common potential sources of RFI that, for example, could form part of the construction equipment or supporting infrastructure

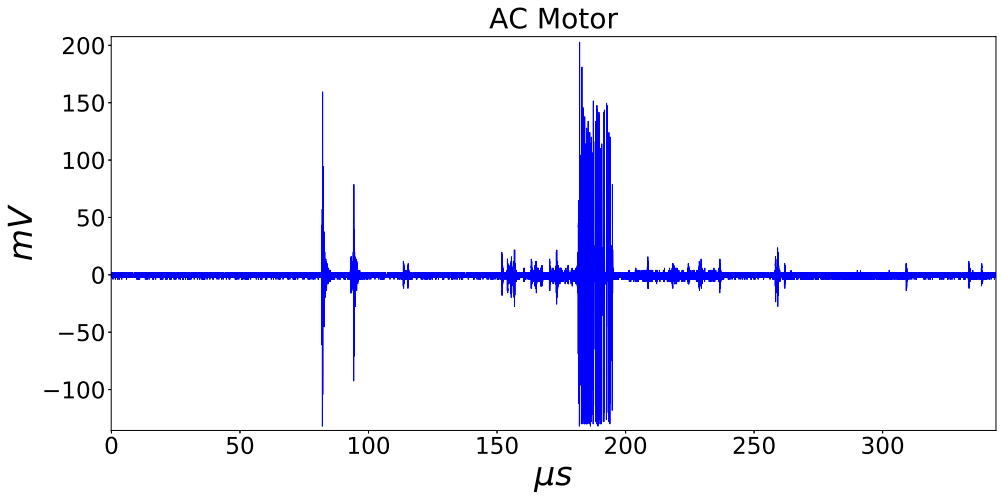

Figure 1.2: An example of an RFI event caused by switching on an AC motor.

at a radio telescope array. These devices were repeatedly switched on and off and the resultant RFI events recorded using a custom-built capturing system based on a Reconfigurable Open Computing Hardware (ROACH) board. Full sequences of RFI transients were captured in recordings up to a few milliseconds in length. The recording band was centred around 146 MHz (close to the centre frequency of instruments such as HERA). An example of a recorded RFI event is given in Fig. 1.2.

The statistical distribution at each time-step across RFI recordings is tested for adherence to a Gaussian distribution for each source. Two tests are applied, the Lilliefors and the Shapiro-Wilk tests. Both indicate that across all recordings of each source, the distributions are in essence non-Gaussian.

Next, distribution fitting is attempted at different time-steps. The Kolmogorov-Smirnov goodness of fit test is used as a figure of merit. Firstly, a number of standard parametric models (for example the Alpha, Power Normal and Laplace distributions) are fitted to the data, without promising results. Secondly, two-component Gaussian Mixture Models (GMMs) are attempted, also without convincing results. Finally, non-parametric kernel density estimation approaches are applied, with a variety of different kernels.

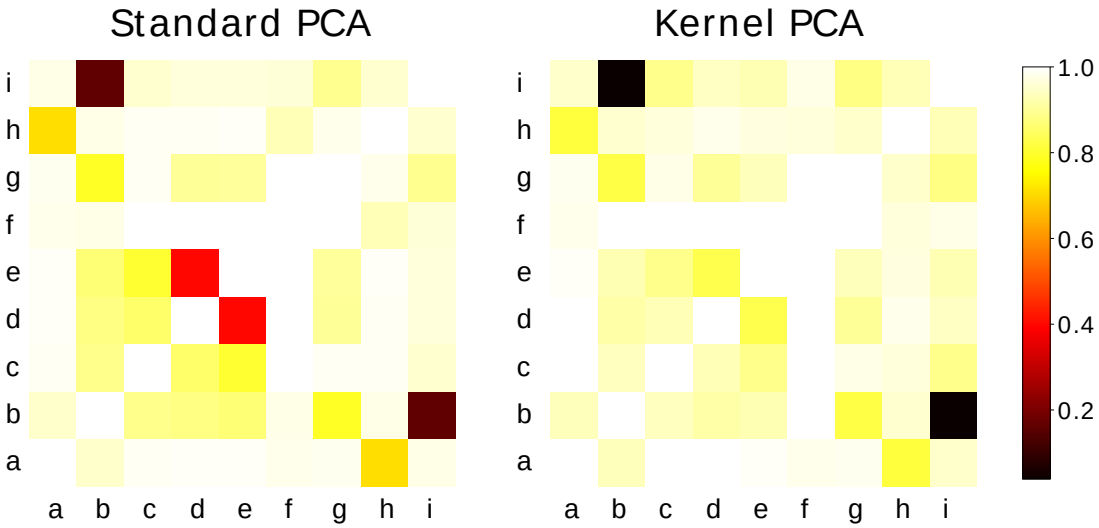

Figure 1.3: The separation between classes for standard PCA and kernel PCA. In this diagram, separation is evaluated using Cohen's Kappa after applying a k-nearest neighbours algorithm. Classes are perfectly separated for a score of 1. A description of each class is given in Chapter 3.

Besides the larger dataset used in the statistical analysis, several other, more limited datasets were also obtained. One of these involved recording transient interference from sources at the MeerKAT site itself.

1.8.3 Components Analysis Techniques

In Chapter 4, the use of components analysis techniques as feature selection steps is explored. Principal components analysis (PCA) is applied to the dataset of transient RFI and compared with a nonlinear method, kernel PCA.

In the initial approach, PCA and kernel PCA are applied directly to the data after some basic preprocessing steps. Results are evaluated by measuring the separation between classes in the principal components space. In Fig. 1.3, confusion matrices for both PCA and kernel PCA are shown, illustrating kernel PCA's advantage over standard PCA. This is expected given the statistical findings in the previous chapter. Kernel PCA has a number of parameters, such as the type of kernel function and its own associated parameters. Their effects on class separability are also investigated.

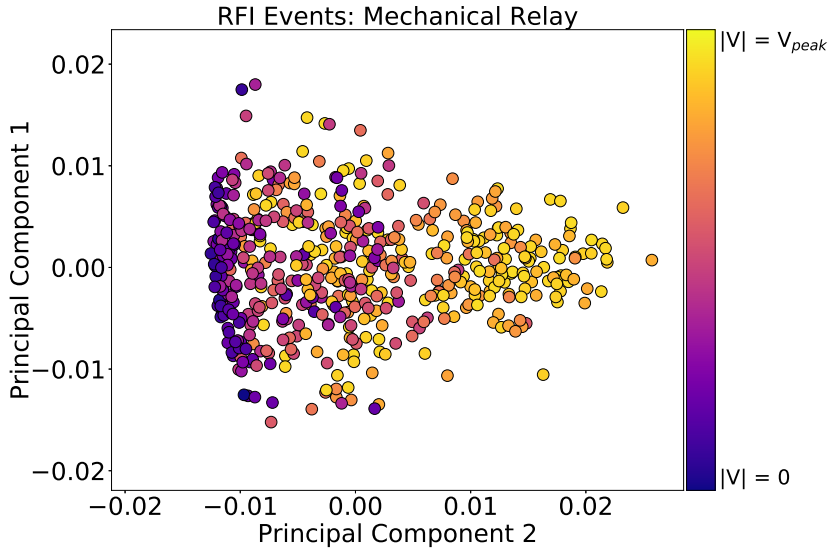

Figure 1.4: Kernel PCA applied to the RFI events recorded when switching a mechanical relay on and off. Each marker represents a particular RFI event. The absolute instantaneous mains supply voltage is indicated by the colour of each marker. It is clear that the instantaneous supply voltage influences the shape of the cluster.

In addition, the suitability of kernel PCA as a feature selection step is evaluated. The data are separated into five sets, four of which are used for cross-validated parameter selection, while the fifth is used for testing. For classification, well-known techniques such as support vector machines are used.

The classification results obtained in this direct approach are, in general, fairly poor. A potential reason for this was suspected during data collection. The instantaneous mains supply voltage appears to affect the shape of the recorded signal (and by implication, cluster separation in the principal components domain). To investigate this, the instantaneous mains supply voltage was recorded along with the time domain RFI emissions themselves for a subset of the RFI sources. An example of the effect of the mains supply voltage in the principal components domain is illustrated in Fig. 1.4. Since kernel PCA is to be used for feature selection, this effect has ramifications for class separability and is analysed further in the chapter itself.

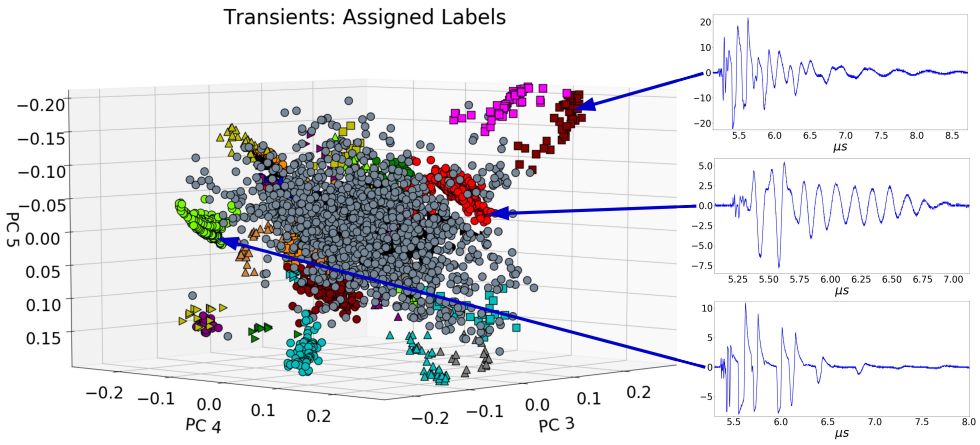

Figure 1.5: The result of applying unsupervised clustering to extracted transients. Transients and their labels are represented by the markers and their shapes and colours. For legibility, transients labelled as noise are not included. Furthermore, only 3 of many possible components are visualised, so some clusters will appear less separated than they actually are. Several examples of transient signals that correspond with different labels are also illustrated.

1.8.4 A Dictionary-Based Approach to Identification

Given the relatively lacklustre results obtained using the more direct classification methods attempted in Chapter 4, a more sophisticated approach is sought. Chapter 5 deals with the concept of a canonical dictionary. Most of the RFI events studied consist of a sequence of many individual transients. A hypothesis is proposed that a dictionary of basic transients could be used to reconstruct any full RFI event. That is, each full RFI event would be represented only as a sequence of labels, one label per transient.

In this approach, each full RFI event must be separated into its constituent transients. Manual separation by a human is not practical, since there are many thousands of transients across all events. Thus, an automated extraction algorithm is developed and evaluated with a special set of human-annotated RFI events. Each transient's metadata is also stored: the event from which it was extracted, the class of the event, and its sequence number.

Next, a dictionary of transients is created. This is achieved by applying kernel PCA to the transients from a training set, and labelling them using density-based unsupervised clustering algorithms in the principal components space. Any tran-

sient can be represented by its label and every full RFI event by a sequence of such labels. An example of a result of the labelling process is given in Fig. 1.5. Having established the procedure for creating a dictionary of transient sub-events, details are given on how to use it to represent hitherto unseen RFI events as sequences of such labels.

Hidden Markov models (HMMs) are chosen for the identification step, given their success in a number of other fields (albeit with entirely different data). Unconstrained HMMs are trained for each class, using the extracted sequences of labels for each event. New, unseen events are then converted to sequences and evaluated under the models. Using the same data splits as before in Chapter 4, significant improvements in accuracy are obtained; see Table 1.1.

1.8.5 Deep Learning Techniques for Transient RFI Classification

Deep learning techniques have risen to prominence in recent years, offering the best performance across a wide variety of machine learning tasks [12]. Chapter 6 is dedicated to investigating the use of deep learning techniques for classifying transient RFI.

The dataset described in previous chapters is again used so that comparisons can be drawn between the different techniques. However, in this chapter a different preprocessing approach is taken. In prior chapters, full RFI events are used, which consist of sequences of individual transients. In this approach, the full RFI events are subdivided into their constituent transients, each of which is considered an instance in its own right. Certain classes contain many more individual transient signals than others, resulting in significant class imbalance. When training models, special consideration is given to addressing this class imbalance.

A number of configurations are investigated, but ultimately the model illustrated in Fig. 1.6 is selected. The convolutional layer is pre-trained and followed with either a bidirectional long short-term memory (BLSTM) layer or a linear support vector machine (SVM). Keras [13] and Tensorflow [14] are used for model training and Hyperopt [15] for parameter tuning.

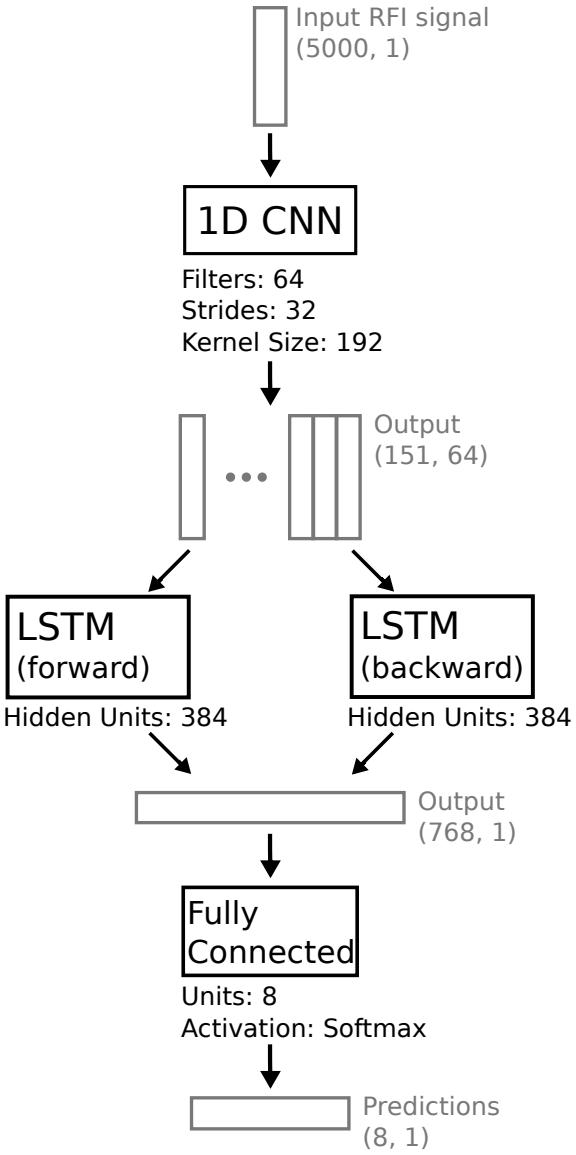

Figure 1.6: The architecture of the model used to classify transient RFI signals (see Chapter 6 for full details).

Classification results are very good, surpassing all other attempts in previous chapters (see Table 1.1; a full evaluation is provided in Chapter 6). However, caution is recommended, as it remains to be seen how well such an approach would fare with new, entirely unseen data. Further work is recommended to investigate the potential of turning this approach into a tool for use with RFI monitoring systems.

Table 1.1: A comparison of the overall accuracy of different classification approaches. Further details are given in Chapter 7.

Classification Method	**Overall Acc.** [%]
Dictionary Approach + HMMs	69.47
Direct Approach + SVM	61.58
Direct Approach + kNN classifier	18.95
CNN + BLSTM (individual transients)	96.36
CNN + SVM (individual transients)	94.03

1.8.6 Summary

The final chapter concludes the thesis by discussing the main findings and making recommendations for future research. To answer the hypothesis: it is indeed possible in principle to classify transient RFI signals by their source, based on their RF emissions alone. Important findings are discussed in the preceding paragraphs.

In a practical system, operating in a new environment, an appropriate initial goal might be to identify entire classes of devices, rather than specific examples. There is scope for further work in identifying the defining characteristics of these classes and their influence in the principal components space or on neural networks. These investigations would constitute a natural extension for the work exposing the influence of the mains supply voltage on class separability.

Another area which is likely to yield interesting results is a more in-depth investigation into the application of this work to natural astronomical transients. This could include distinguishing between natural transients and RFI transients, recognising types of natural transient, or searching for hitherto unknown transients.

Chapter 2

Literature Review

This chapter reviews the literature pertinent to transient RFI classification. RFI flagging, modelling and classification are each dealt with to provide context for the work in this thesis. It is no surprise that RFI is detrimental to radio astronomy observations, as discussed in Chapter 1. Continuous, intentional transmissions tend to cause more widespread harm than transient RFI. However, transient RFI is associated with particular problems and is in some ways more difficult to mitigate than other types.

For example, transient RFI has the potential to be confused with real astronomical sources. At the Parkes observatory in Australia, a number of millisecond transients were recorded, each exhibiting frequency dispersion suggestive of an extragalactic origin [3]. It was determined that these transient signals (named "Perytons") were entering a sidelobe of the telescope and were thus of terrestrial origin. A number of potential causes were proposed, even including ball lightning [16]. Finally, in 2015 their origin was unmasked as a microwave oven [4]. The associated transient signal was caused by opening the microwave oven prematurely. This particular case illustrates how even rare, intermittent transient RFI can have a damaging effect on research. As a result of the Perytons, for a time doubt was cast on the interpretation of Fast Radio Bursts (FRBs) [3].

There have also been other cases where the similarity between some forms of transient RFI and natural astronomical transients has caused problems. It has been shown that a popular RFI removal pipeline removes some astronomical transients along with RFI transients [17]. Some astronomical observations

are particularly vulnerable to transient RFI. For example, observations of giant pulses (from radio pulsars) can be obscured by transient RFI. These natural pulses could also be confused with transient RFI signals, especially if they exhibit frequency dispersion [18].

Single-dish radio telescopes suffer a particular vulnerability to transient RFI. In their multi-antenna array counterparts, if a transient signal is detected in one dish but not the others, it is most probably terrestrial transient RFI.

2.1 RFI Detection

Automated RFI detection algorithms and pipelines deal almost exclusively with time-frequency data from radio telescopes themselves, rather than remote monitoring stations.

2.1.1 Mean and Median-Based Detectors

Some of the simplest techniques involve applying mean and median absolute deviation filters to the channelised data. The statistic is calculated over a moving window and signals are flagged as RFI when they exceed a threshold. The mean is still used at times due to its simplicity; for example such an approach was used to detect RFI in data from the Parkes Multibeam Pulsar Survey in a 2012 publication [19]. However, RFI often manifests as outliers [20], which may be missed when the mean of a window is calculated. The median is a robust estimator and is not affected by outliers as is the case with the mean [20]. Calculating the median does however require sorting, which is computationally more intensive than ratio based methods. RFI has been flagged in radiometer earth observations by applying a threshold to the median absolute deviation calculated over a moving window [21].

Setting a threshold for RFI is challenging if naturally varying signals are to be ignored. If the mean is calculated over a moving window, then the mean is liable to drift for slowly varying signals. In addition, impulsive RFI is usually non-Gaussian [22]. Thus, a method to detect non-Gaussian RFI against a background of Gaussian-distributed, natural signals, would be very useful.

2.1.2 Kurtosis

Kurtosis estimation [23] indicates to what extent a given density function differs from a normal Gaussian distribution in flatness or peakedness, so it differs from the measure of skewness. More specifically, positive (leptokurtic) kurtosis is indicative of thicker tails and a higher peak, while negative (platykurtic) kurtosis signifies thinner tails and a lower peak [24]. Temporal kurtosis is useful for detecting transient RFI signals or low duty-cycle RFI [25, 26]. However, it is far less effective at higher duty cycles [25]. Furthermore, kurtosis will not detect RFI with a duty cycle of 50% for a given channel [25, 26]. Spectral kurtosis, on the other hand, is well suited to detecting CW interference and RFI with a high duty cycle [25]. Kurtosis-based RFI detection systems have also been proposed for airborne and spaceborne microwave radiometry [27, 26].

2.1.3 Sequential Probability Ratio Tests

Sequential probability ratio tests (SPRTs), originally applied to problems of quality control [28], have also been used to flag RFI [29, 28]. The cumulative sum (CUSUM) algorithm, an SPRT with a threshold, has been used to detect RFI [29]. Offringa et al suggested using the total detected power or the power received at a single frequency by a single dish as the incoming data on which to apply the CUSUM algorithm [28]. A drawback of SPRTs is that RFI detection will be delayed to some extent by necessity [28].

2.1.4 Combinatorial Thresholding

Since most RFI detection systems deal with time-frequency plots, a variety of 2D flagging approaches have been developed. One such approach is combinatorial thresholding, introduced by Offringa et al [28]. Single RFI events often take place over several samples. Instead of using individual thresholds for each sample, the combinatorial threshold considers adjacent samples. If neighbouring samples fail to exceed a threshold individually, they may still be marked as RFI if they both exceed a more lenient threshold. The required threshold is reduced as the number of connected samples increases. It is claimed that false positives are reduced since

individual thresholds can be made more stringent [28].

2.1.5 Transient RFI and Astronomical Radio Transients

Some work has been devoted specifically to distinguishing between transient RFI and astronomical sources of transient signals (such as fast radio bursts and pulsars). One approach discards signals as RFI if their dispersion measure is too low [30]. However, some transient RFI signals have been known to exhibit dispersion similar to distant astronomical sources [4]. Therefore, it is potentially risky to rely on dispersion measure alone. Other work has used basic artificial neural networks (ANNs) to detect pulsar signals and distinguish them from RFI [31]. More recently, machine learning approaches have been used to search specifically for pulsars [32, 33] and fast radio bursts [34, 35], discarding all other unwanted signals (including RFI).

2.1.6 RFI Flagging Pipelines

A variety of software packages have been developed to automatically flag RFI in radio astronomy observations. AOFlagger [36, 37] is one example which has been used with Low-Frequency Array (LOFAR) and Murchison Widefield Array (MWA) data. Among other techniques, it makes use of combinatorial thresholding (described previously). The effect of the AOFlagger pipeline on legitimate astronomical transients has also been investigated [17]. As part of the work, a modification was proposed to mitigate the flagging effect on astronomical transients (as opposed to RFI). Another example of software for RFI flagging is the scripted e-MERLIN RFI-mitigation pipeline for interferometry (SERPent) [38]. It was designed for the multi-element radio linked interferometer network (MERLIN). Some of the techniques described in this work were successfully applied to detecting RFI in radiometer earth observation data [21].

Besides searching for transients like FRBs, deep learning techniques have also been employed for standard RFI flagging and excision. In recent work, deep convolutional neural networks have been used to flag RFI in 2D time-frequency data [39]. The implementation that was selected (U-Net) was compared to the

SumThreshold combinatorial thresholding algorithm used in the AOFlagger software. Results were comparable when applied to a simulated dataset.

2.1.7 Remote Monitoring Stations for RFI Mitigation

Another avenue for RFI mitigation is to install independent RFI monitoring systems at radio telescope arrays. Such systems operate independently of radio telescopes, and continuously monitor a wide frequency range. Since all bands are continuously monitored in all directions, it is easier to deduce the sources of RFI. Once RFI sources have been identified, they may be removed or replaced where possible. It can even be possible to forecast the occurrence of certain types of RFI [40]. At the MeerKAT array in the radio-quiet Astronomy Advantage Area in South Africa, independent monitoring stations have been installed. Several related research projects have been undertaken at universities in South Africa.

One of the early RFI monitoring systems at MeerKAT was called the Real Time Transient Analyser, or RATTY. It was built around a ROACH 1 board and installed in a trailer at the MeerKAT site, shown in Fig. 2.1. A directional log-periodic array antenna was used. Automated monitoring software was developed for RATTY by C. Schollar [41]. Among other functions, the software collects spectra from the monitoring station and stores them in a database. Some RFI detection algorithms are applied and statistics like channel occupancy are provided. A web interface was also written.

The successor to RATTY is a ROACH 2-based platform, which is configurable to capture spectra or time-domain recordings of transient signals [42]. It uses a 10-bit ADC with a maximum sampling rate of 1.8 GSa/s. The system and its development are described in greater detail in A.R Botha's PhD thesis [43]. Currently, a newer, more versatile system is being developed for both fixed and mobile monitoring stations at MeerKAT [44]. Examples of such monitoring stations are provided in Fig. 2.2. Independent monitoring stations have also been built at other radio telescopes, for example the Medicina and Sardinia radio telescopes [45] and the Australian Square Kilometre Array Pathfinder (ASKAP) [46].

Various antennas have been specifically developed with RFI monitoring at

Figure 2.1: The early RFI monitoring trailer housing RATTY (2014).

MeerKAT in mind. Ideally, such antennas would be wide-band, omni-directional and non-dispersive. A number of projects have been undertaken by Dr Gideon Wiid and colleagues, including the exotic-looking Protea antenna [47] and low-cost, 3D-printed versions [48, 49, 50, 51].

Finally, some work has dealt with transient RFI direction-finding systems using techniques such as multilateration. A masters project (in progress) by J Gowans has shown promise (for more details, see his GitHub profile[1]). In a practical scenario, basic transient sources were located using an array of four folded dipole antennas. A different masters project [49] also offers good results. An array of 3D-printed asymptotic conical dipole antennas (which perform across a wider band than folded dipole antennas) was used with a phase correlation algorithm. In separate work [52] unrelated to radio astronomy, transient RFI sources were located using a portable wideband system. Three separate asynchronous receivers were used to locate sources of impulsive noise in a substation.

[1] A preprint of his thesis and the related code are available at https://github.com/jgowans

Figure 2.2: Examples of fixed and mobile RFI monitoring systems at MeerKAT. The left-hand image depicts a vehicle equipped with an RFI monitoring system along with a telescopic antenna mast. A fixed RFI monitoring station installed at the top of Losberg (a small peak close to the MeerKAT core) is illustrated on the right. Images courtesy of the RFI team, SARAO (formerly SKA SA).

2.2 Characterising Transient RFI

Past work on modelling impulsive RFI has generally focussed on amplitude distribution. For the work in later chapters of this thesis, amplitude information is less useful than other measures. Nevertheless, past work on modelling amplitude distributions is included here for context. Impulsive RFI has been shown to be non-Gaussian in distribution [53, 22]. Therefore, Gaussian noise models are inadequate, and new models, such as stable distributions, are required [53].

2.2.1 Middleton's Models

In the field of telecommunications, various different attempts have been made to model impulsive RFI. Foremost are the Middleton models for RFI [54]. Middleton proposed three classes of impulsive interference: Class A, for which the spectrum of the interference is narrower than that of the receiver; Class B, for which its spectrum is broader than that of the receiver, and Class C, which is a

sum of Class A and Class B noise [54]. These models are validated with empirical data [54], looking at the distribution of amplitude in signals emanating from ore crushing machinery, fluorescent lights, powerlines and automotive ignition noise.

2.2.2 Symmetric Alpha Stable Models

While proven accurate [22, 54] Middleton's models can be difficult to apply, especially the Class B model, which has 7 parameters [22]. Consequently, the Class B model is often approximated with a Symmetric Alpha Stable model [55, 22]. The Symmetric Alpha Stable model can be likened to a Gaussian distribution with much thicker tails, the thickness of which is controlled by the eponymous alpha. One formulation of the probability density function of a Symmetric Alpha Stable model is as follows [56]:

$$f_\alpha(x;\gamma,\delta) = \frac{1}{\gamma} h\left(\frac{x-\delta}{\gamma};\alpha\right) \tag{2.1}$$

In which

$$h(x;\alpha) = \frac{1}{\pi}\int_0^\infty \cos(xt)e^{-t^\alpha}dt \tag{2.2}$$

A small, positive value of α indicates high impulsivity and thicker tails [56] while a value of 2 yields a Gaussian distribution [53, 56]. In 2.1, δ is the location parameter and γ is the dispersion. γ is similar to the standard deviation in a Gaussian distribution [56].

Parameter estimation for the Symmetric Alpha Stable model has been explored in the literature, with many methods proposed. However, the prevailing computationally efficient method is that which was developed in [22].

2.2.3 Attempts at Modelling Impulsive RFI

Radio telescopes often observe in the same bands used for telecommunications. Thus, analyses of impulsive noise in the field of telecommunications may also be useful for characterising RFI affecting astronomical observations. In [57], recordings were taken of impulsive noise in a 10 MHz band centred at 762 MHz. The

distributions of the amplitude, pulse duration and inter-arrival time of the impulses were modelled. The authors attempted to fit Power Rayleigh, lognormal, exponential, Poisson and Gamma distributions to cumulative distribution plots of the experimentally obtained data. While the Gamma distribution seemed to fit well in most cases, it only passed a Kolmogorov-Smirnov test in the case of impulse duration [57].

There have also been various studies on the emissions caused by microwave ovens [58, 59, 60]. In [58], Middleton's Class A model is applied to recorded interference from a microwave oven driven by a switching power supply and another by a transformer. However, in [59], it is claimed that an altered Gaussian mixture model fits the data better than both a basic Gaussian mixture model and Middleton's Class A model.

2.2.4 Modelling RFI in Power Line Communications

Power Line Communication (PLC) is another field that deals extensively with interference. PLC involves transmitting data along AC power lines. One potential disadvantage of this approach is the presence of impulsive noise, caused by appliances and machines which draw power from the AC supply. Consequently, there is a body of work that deals with modelling the statistics of this impulsive noise, which is non-Gaussian in nature [61]. In the time domain, some work has focussed on modelling the amplitude, length, and inter-arrival times of impulses [62, 63, 61] without separating them according to their sources. Others, however, have looked at the emissions of individual devices, for example an electric drill and SCR light dimmer [64]. [65] gives a set of examples of signals produced by everyday electrical appliances including a vacuum cleaner, kettle, refrigerator, incandescent lamp and several others. Middleton's models have been applied to impulsive noise in PLC networks [66, 62]. However, several other statistical models have also been suggested, including partitioned Markov chains [61], Gaussian mixture models [64], Beta-like distributions and Gamma distributions [64]. In [62], it is explained that the Middleton Class A model using only the first three terms of the cumulative sum, is very similar to a two-term Gaussian mixture model.

2.3 Classifying RFI By Source

There are few attempts in the literature to classify transient RFI in a radio astronomy context. Consequently, work from other fields that is relevant to RFI classification is also included here.

Doran [19] classified transient RFI in the time-frequency domain using data from the Parkes radio telescope in Australia. Two high-level attributes are used, time of day and pointing direction, in addition to the average intensity by frequency channel. Clustering was achieved using the mini-batch k-means algorithm, an unsupervised technique. However, the practical utility of this approach is limited due to the use of high-level attributes and unsupervised clustering techniques. It would be difficult to infer the sources of specific sources without additional investigation. Temporal averaging and the limited number of frequency channels also restricts the number of possible features. Therefore, information about the structure (which may have discriminatory value) of RFI events is lost.

In other work [67] different vehicles were classified by their RF emissions when running. Time-domain captures of transients were taken and used for classification. Time-frequency spectrograms were calculated from which a variety of parameters were extracted and reduced using PCA. Classification was performed using a neural network with accurate results; however, only five different vehicles were classified.

In another investigation [68], wireless doorbell receivers and a radio controlled toy truck were identified by classifying their unintended radio transmissions. Again, several high-level attributes, calculated for a number of chosen frequency bands, were used as an input to a multilayer perceptron neural network, as was done in [67]. However, for this experiment, cross-correlation with previously obtained recordings was used to aid source identification [68].

In recent work [69], household appliances were classified by their RF emissions for non-intrusive appliance load monitoring (NIALM). 6 appliances were classified using frequency domain information and a kNN-based approach. However, the applicability of their work to radio astronomy-related RFI monitoring is limited by the choice of recording band. For classification, recordings were taken

in the range from DC to 1 MHz, which is well below the observing bands of terrestrial radio telescopes. Furthermore, they only classified steady-state signals, whereas this thesis is concerned with transient RFI.

In [70], transient RFI is classified in the time domain. Several RFI sources were recorded at the MeerKAT radio telescope site in the Northern Cape of South Africa and classified according to their source. This work is relevant to this thesis for a few reasons. It used data recorded with hardware similar to that which will be used in planned remote monitoring stations, it recorded actual RFI sources at a radio telescope array, and used supervised learning algorithms. However, the time domain captures were limited to 8 microseconds, which is too short to record full RFI events (as demonstrated later in this thesis). The sets of labelled recordings of each source were also relatively small, occasionally containing fewer than 10 examples.

2.4 Classification of Transients in Other Fields

The need for classifying transient signals is present in a variety of other fields and is not limited to the radio spectrum. Transient classification is applied in automated speech recognition [71], bio-acoustics [72] and sonar target classification [73]. An interesting approach is taken in work classifying cricket songs by species. Audio pulses produced by the crickets are extracted and classified using a variety of features. These features include pulse length and frequency, calculated over moving windows [72].

Radio frequency fingerprinting (RFF) is another field that involves the processing of transient portions of transmission signals. RFF is the identification of specific electronic devices based on their individual physical characteristics, which render their turn-on or transmission signals unique. The main application of radio-frequency fingerprinting is in security, to prevent device cloning for example. Feature selection is usually necessary to reduce the dimensionality or complexity of the signals to be classified. Various methods have been tested, among them principal components analysis [74], dynamic wavelet fingerprinting, wavelet packet decomposition and higher order statistics [75]. Likewise, a variety of classification or recognition methods have been employed to identify

unique devices. These include probabilistic neural networks [76, 74] and multiple discriminant analysis [77].

2.5 Conclusion

RFI is a significant problem which has been studied extensively over the preceding decades. Many different tactics for dealing with RFI have been explored. However, the majority of the work undertaken thus far has dealt with RFI flagging in 2D time-frequency plots, rather than determining specific RFI sources. These methods range from basic statistical approaches (such as median absolute deviation) to relatively complex processing pipelines. CW RFI is easier to identify in time-frequency plots than transient RFI. Another strategy employed at MeerKAT and other radio telescopes is to install independent RFI monitoring stations, which have the capability to detect and record time-domain transient RFI signals. In general, very little research has been dedicated to the classification of transient RFI by source. Some basic machine learning techniques have been used in a handful of studies. Almost no work has investigated the use of contemporary deep neural networks. Some work from related fields (like non-intrusive appliance load monitoring) is partially relevant. Inspiration can be drawn from some of these adjacent research areas when designing new approaches to RFI classification.

Chapter 3

Characterising Transient RFI[1]

To inform the development of the classification approaches described in Chapters 4, 5 and 6, attempts are first made to characterise transient RFI statistically. To accomplish this and obtain training and testing data for classification approaches, a number of datasets of transient RFI were recorded. These datasets are described in detail in Section 3.1. Statistical tests and modelling attempts (on a subset of the data) are provided in Section 3.3.

3.1 Data Collection

This section describes how each of the different datasets referenced in this thesis were obtained.

3.1.1 Dataset 1

This exploratory dataset was obtained in September 2014 at MeerKAT (under construction at the time). The dataset was recorded together with CJ Wolfaardt, a master's student, and C Schollar of SARAO. CJ Wolfaardt published a conference paper [70] dealing with this data.

[1]This chapter is based in part upon the following publications:
Czech, D, A Mishra, and M Inggs. "Characterizing transient radio-frequency interference." Radio Science 52 (2017): 841-851.
Czech, D, A Mishra, and M Inggs. "A dictionary approach to identifying transient RFI." Radio Science 53 (2018): 656-669.

Figure 3.1: An early version of the Real Time Analyser in use, capturing transient RFI signals at the MeerKAT site for Dataset 1.

Recordings were taken with an early version of the Real Time Analyser (see Fig. 3.1), an instrument specifically designed for RFI detection. For further information on the development of this instrument, refer to Botha et al [43, 42] and Millenaar et al [44]. Recordings were taken in separate frequency bands between 50 MHz and 2.55 GHz, and a sampling rate of 1.8 GSa/s was used. Time-domain captures were only 8192 samples in length. Two different log-periodic dipole array antennas were used. Examples of recordings from several sources are provided in Fig. 3.2.

This dataset suffers from a number of notable limitations, restricting its utility. Foremost is the low number of recordings per class; many classes contain fewer than 10 usable examples. Therefore, it would be difficult to train and test classification algorithms with this dataset, and performance results would likely be unreliable. It would also be very difficult to perform statistical analyses across repeated recordings in each class. Another shortcoming of this dataset is the length of the time domain recordings. As indicated in Fig. 3.2, too few samples are captured to record the full length of RFI events. This issue was also noted in CJ Wolfaardt's work [70]. Additionally, since log-periodic dipole array (LPDA) antennas suffer from frequency dispersion, it is possible that the antennas themselves altered the signals that were recorded.

Due to these limitations, it was decided that this dataset would not be suitable for many of the planned investigations. Thus, a new dataset was required, one suitable for statistical analysis and the training and testing of certain classification algorithms. This new dataset is discussed in Section 3.1.2.

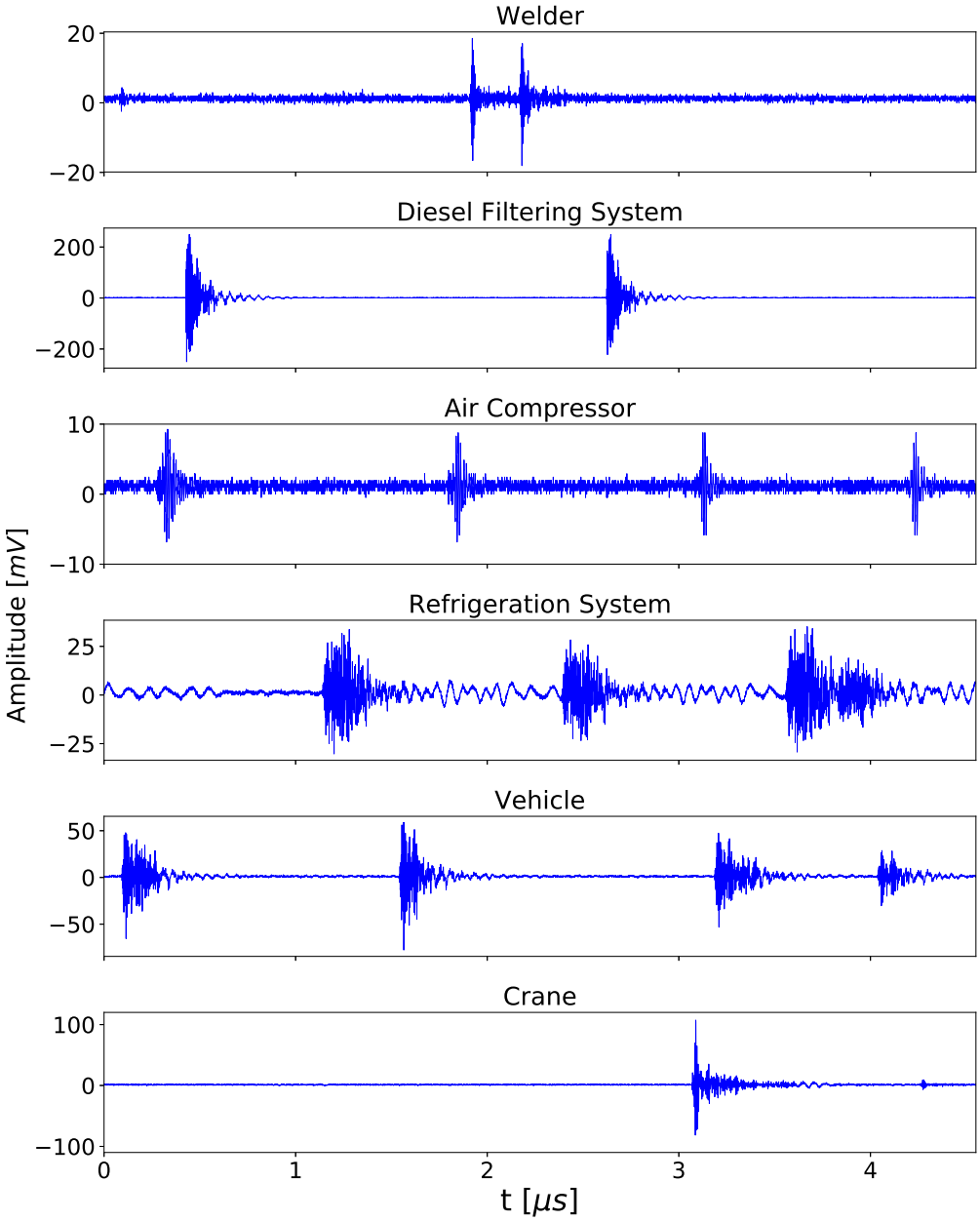

Figure 3.2: Examples of some of the transient RFI signals recorded at the MeerKAT construction site in 2014, using the first recording band (50 - 828 MHz). The crane is a small JLG Toucan 10E (a 'cherry picker') and the refrigeration system belongs to a Carrier Transicold Thinline refrigerated shipping container. The signal from the vehicle was recorded when starting its engine. The full extent of many of the signals was likely not recorded. The recording time (of 8192 samples) is insufficient to capture entire RFI events for many sources.

Despite these disadvantages, this dataset still has value. It shows that it is possible for this type of transient RFI to be present at a radio telescope array. It demonstrates that independent RFI monitoring systems such as those under development for MeerKAT [44] will be capable of detecting such transient RFI generated by nearby sources. In addition, it illustrates that longer recording lengths are needed. This dataset provides a useful starting point for the different analyses undertaken in this work.

3.1.2 Dataset 2

The second dataset is used predominantly in this thesis. It was recorded to address the main limitations of Dataset 1. The recording length was increased so that RFI events could be recorded in their entirety. Many repeated recordings were obtained of each source (see Table 3.1). Since full events were recorded, and each full event consists of a sequence of transients, even more individual transients were recorded (see Table 6.1 in Chapter 6). In addition, a non-dispersive antenna was used.

Table 3.1: The different RFI sources in Dataset 2.

Label	Source Description	No. Events
a	Compact fluorescent lamp 1	64
b	Sander (power tool)	135
c	Small step-down transformer	142
d	5m length of cable	102
e	Mechanical relay with 700W load	128
f	Mechanical relay without load	141
g	AC motor ($\approx$ 1 kW)	63
h	Compact fluorescent lamp 2	64
i	Small switching power supply	105

The dataset consists of 944 recordings from 9 different classes (see Table 3.1). These particular devices were selected for several reasons. Firstly, they are com-

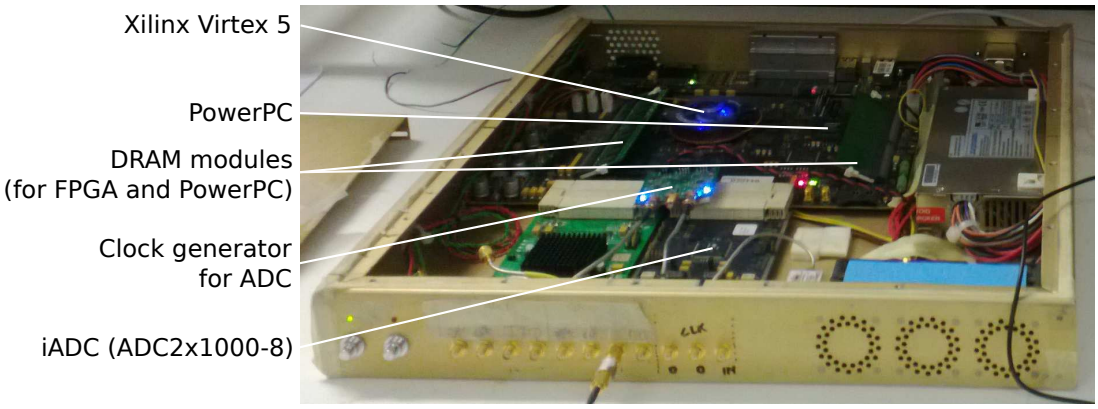

Figure 3.3: The Reconfigurable Open Architecture Computing Hardware (ROACH) board used to obtain the dataset, loaned from J Gowans/SARAO.

mon devices likely to be present as part of the infrastructure or maintenance and construction equipment surrounding a new radio telescope array. In the case of MeerKAT, new radio telescopes will be under construction while other existing arrays are still operating. Secondly, these devices are cheap and easily acquired.

Before taking recordings, an area as radio-quiet as possible was sought since a shielded anechoic chamber was not available. Several locations were evaluated using a portable spectrum analyser (a Keysight Technologies N9912A FieldFox). The large basement of the university's library was selected as a recording location. There, no other RFI transients could be detected, even at much higher sensitivity levels than were used for actual recordings.

Each device was switched on and off multiple times. When a device was switched on or off, transient RFI signals were generated. These signals were detected and recorded with the capturing system described in Fig. 3.4. Since the switching of each device was controlled, it was possible to know with certainty that a transient signal belonged to a particular source. Furthermore, the power of each RFI event was kept much greater than the noise by adjusting a variable attenuator.

These recordings were taken using a Reconfigurable Open Architecture Computing Hardware (ROACH) [78] board, displayed in Fig. 3.3 with attached ADCs. It was programmed using the CASPER toolflow [79] to record triggered time-domain captures of transient signals. An equal number of samples were recorded

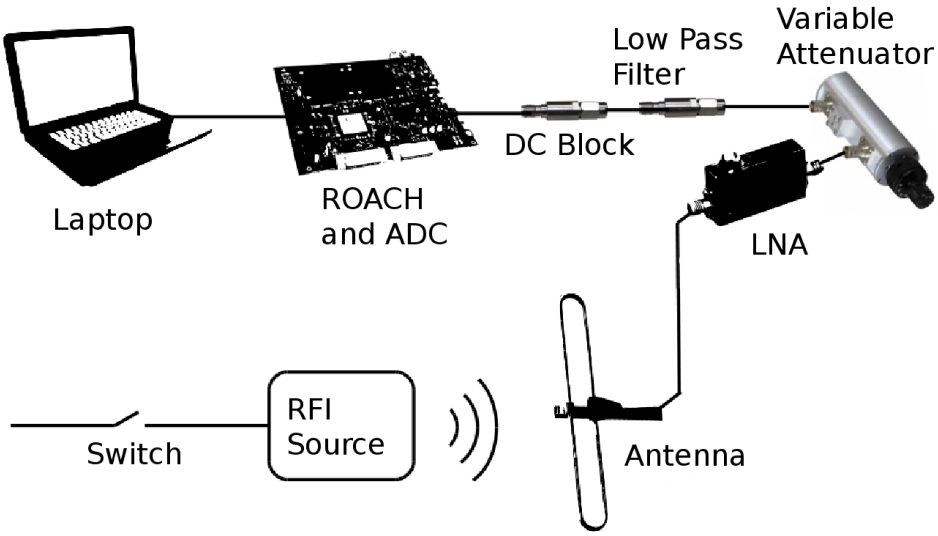

Figure 3.4: The transient RFI capturing system for Dataset 2. Each RFI source was switched on and off multiple times.

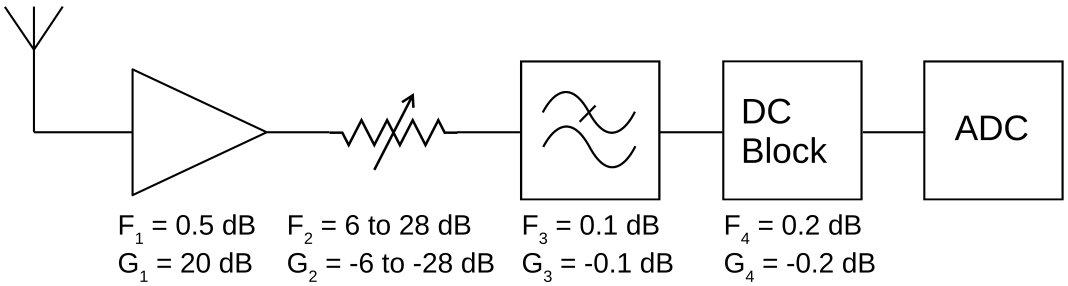

Figure 3.5: The front-end chain for the capturing system depicted in Fig. 3.4. In this figure only, F_i is the noise figure and G_i the gain of the i^{th} stage.

before and after the triggering instant to ensure that full signals were recorded. An ADC2x1000-8 ADC [80] was used with a sampling rate of 1.6 GSa/s. The experiment is illustrated in Fig. 3.4 and Fig. 3.5 depicts the RF front-end.

A folded dipole antenna was selected since such designs are non-dispersive (compared with LPDA antennas, for example) and inexpensive. A centre frequency of 146 MHz was chosen, since it is close to the centre of the operating bands of radio telescopes such as PAPER and HERA [7]. RFI signals in this band therefore have the potential to interfere with observations by such telescopes. The front-end also includes a low-pass filter (to prevent aliasing) and a DC block to protect the ADC from any accidental DC signals.

Figures 1.2 and 3.6 provide examples of the RFI signals recorded in this dataset. Each full-length RFI signal in Figures 1.2 and 3.6 consist of sequences of individual transients. These transients also differ from one-another; Fig. 3.7 displays an example from each source. Chapter 6 deals with the classification of RFI sources based on individual transients (rather than full-length RFI signals).

In Fig. 3.6, it can be seen that while repeated full-length signals from the same source vary from one to the next, there is a common underlying structure to them. This implies that existing classification techniques designed to handle such variations may offer good results. For example, hidden Markov models have been used in speech processing to classify spoken words. A word consists of a sequence of phonemes and is produced slightly differently each time it is spoken. In Chapter 5, this approach to classification is investigated further.

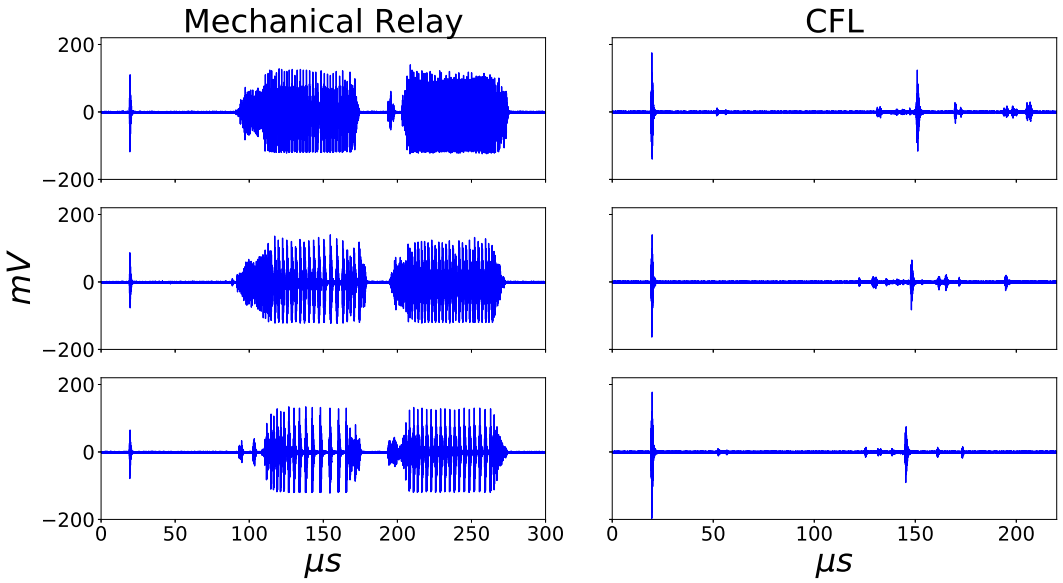

Figure 3.6: Further examples of full RFI events. These are caused by switching a mechanical relay with a resistive load and switching off a compact fluorescent lamp. While there are variations between successive examples, there is likely to be a common underlying structure to them.

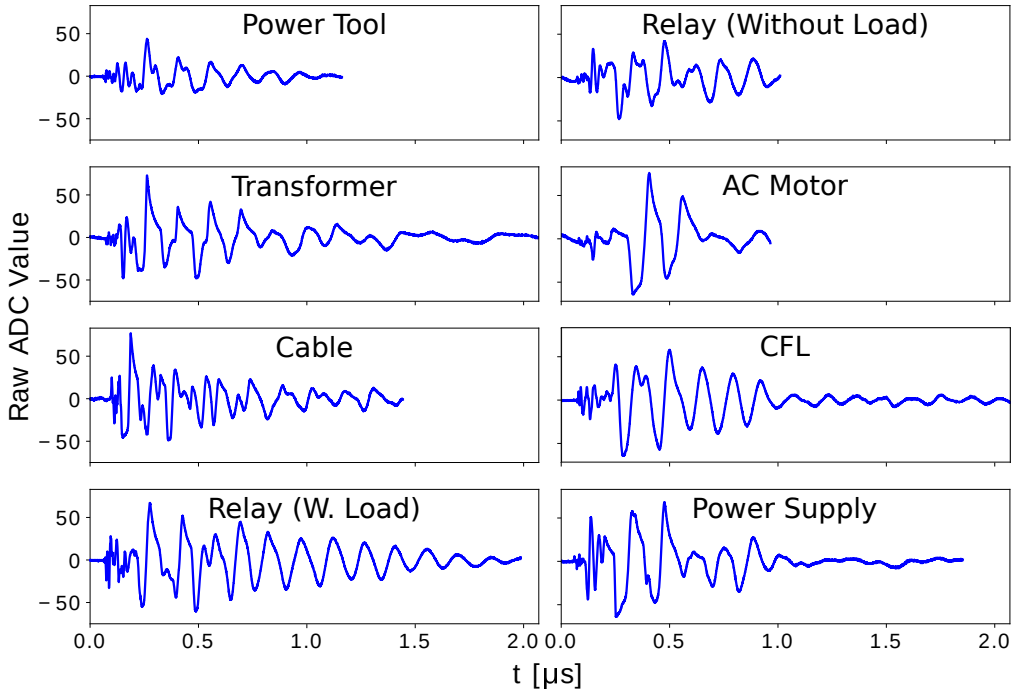

Figure 3.7: Individual transients extracted from full-length recordings of each RFI source. Full-length recordings consist of sequences of such transients. The lengths of these transients are similar to those present in the signals displayed in Fig. 3.2. This figure is modified from the publication: Czech, D, A Mishra, and M Inggs. "A CNN and LSTM-Based Approach to Classifying Transient Radio Frequency Interference." Astronomy and Computing 25 (2018): 52-57.

3.1.3 Dataset 3

The third dataset consists of recordings of RFI events taken with an Agilent MSO9104A mixed signal oscilloscope. In this dataset, the AC supply voltage was recorded simultaneously with the transient RFI signals, as illustrated in Fig. 3.8. In Section 4.6, this dataset is used to investigate the effect of the phase of the AC supply on cluster separability in the principal components domain. Three of the devices recorded in Dataset 2, the relay, switching power supply unit and transformer were used in this experiment. Each device was switched on and off between 200 and 300 times each, and the corresponding RF signals recorded (along with the instantaneous supply voltage).

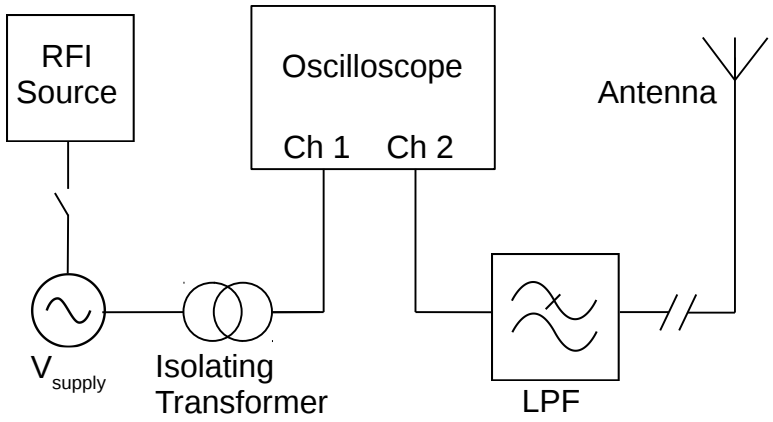

Figure 3.8: The experiment to record the third dataset. The AC supply voltage is recorded simultaneously with the transient RFI signals.

The same folded dipole antenna from Section 3.1.2 was used again. Note that no LNA was used in this case, since the oscilloscope itself contains its own front-end components. However, an external low-pass filter was included to prevent aliasing (should it have occurred). The supply voltage was measured via a small isolating transformer for reasons of safety and convenience.

3.2 Comparison with Astronomical Transients

This section briefly compares the RFI transients in Datasets 1, 2 and 3 with natural transient RF phenomena that would be recorded by radio telescopes. It should be noted that the techniques proposed in this thesis were developed primarily to classify time domain transient RFI signals as recorded by independent RFI monitoring stations (see Section 2.1.7). The intent is not to distinguish between astronomical radio sources and transient RFI, since independent remote monitoring stations do not detect astronomical sources by design. Nevertheless, it may be possible to adapt these techniques for use with data from radio telescopes, an interesting area for potential future research (see Section 7.2).

Some of the most important transient astronomical phenomena include fast radio bursts (FRBs) and pulsars. It has been shown that some transient RFI signals look very similar to FRBs [4]. However, of the few known FRBs de-

tected at the time of writing, only a handful are comparable in length to the full RFI signals presented in this chapter. Furthermore, those short FRBs are significantly longer than the individual transients of which the RFI signals consist. For example, the transients in Fig. 3.7 are around $1.5\mu s$ in length, while the shortest FRB detected (as of the time of writing) is about $350\mu s$ in length [81]. However, if time resolution is reduced (for example via integration), then the individual transients may merge into a single signal comparable in length to one of the shorter FRBs.

Pulsar signals are repeated radio transients produced by the radio beams of rotating neutron stars or white dwarf stars. Rotation periods of neutron star pulsars vary from $1.4\mu s$ [82] to several seconds [83], so the shorter millisecond class pulsars produce signals similar in length to some of the full RFI signals. The signals of most known pulsars are too weak for individual pulses to be detected. However, some powerful pulsars exist for which certain individual pulses have been studied. [84].

One method for distinguishing between transient RFI and authentic astronomical radio transients is to consider dispersion measure (DM). Distant radio transients will appear dispersed by frequency due to the interstellar medium. Since terrestrial sources of transient RFI are close to the receiving antenna, no such dispersion can occur. However, some transient RFI signals have been detected which look as if they are dispersed [4]. In addition, some frequency modulated continuous wave FMCW radar pulses (which generally consist of frequency chirps) may appear dispersed. Therefore, it would be unwise to rely on the DM of signals alone.

The RFI signals illustrated in this chapter do not appear dispersed in such a convincing manner, as illustrated in Fig 3.9. For example, if the signals in Fig. 3.9 exhibited significant dispersion, the broadband streaks in the spectrogram on the right-hand side would be expected to follow a curved path. The higher frequency portions would be expected to arrive before the lower frequency portions, which is not the case here. Nevertheless, the algorithms presented in subsequent chapters may be trained to accommodate such sources of RFI that do in fact exhibit dispersion.

It may also be possible to adapt some of the techniques proposed in later

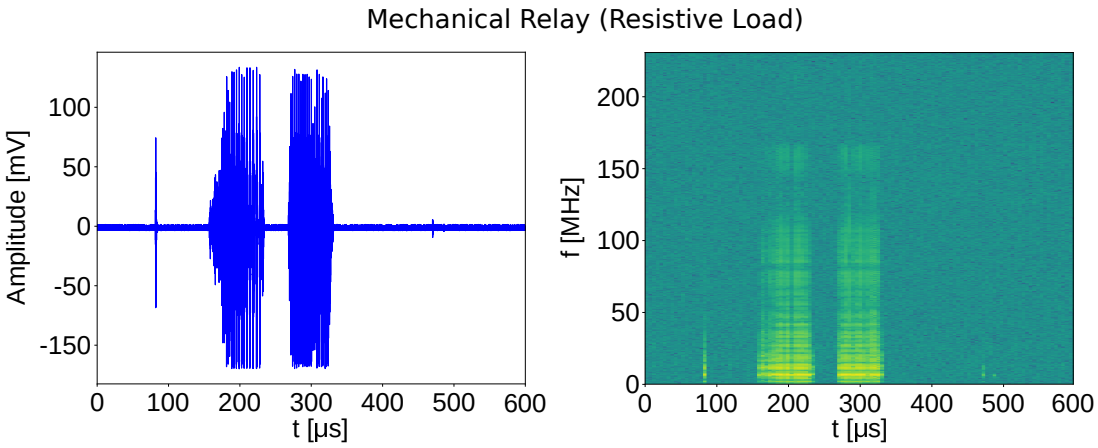

Figure 3.9: A spectrogram of one of the recorded signals. The transients are not dispersed as they do not exhibit a convincing sweep in frequency over time. The higher frequency portions of each transient would be expected to arrive before the lower, which is not the case here (see especially the first transient in the sequence).

chapters to identify different types of astronomical radio transients as well as different sources of transient RFI. This research is left for future work, since the investigations in later chapters deal entirely with time-domain signals as would be recorded by independent remote monitoring stations.

3.3 Statistical Characterisation

Before attempting to apply any classification techniques to the data, a statistical analysis is carried out on the largest of the datasets (Dataset 2). First, the data are tested for normality across all repeated recordings at each time-step. This information is important, because it has implications for feature selection and classification techniques. For example, if the data do not conform to a Gaussian distribution at each time-step, using PCA as a feature selection step may not suffice. This is because ordinary PCA expects the input data to have a multivariate Gaussian distribution. In addition, there are no other attempts in the literature to model transient RFI across repeated recordings at each time-step. That is, there are no attempts to model the distributions of values taken (one each) from the same specific temporal locations in each recording. Some

previous researchers have assumed Gaussian distributions for various reasons, such as a lack of empirical data [85].

Two separate tests for normality are applied, the Lilliefors [86] and Shapiro-Wilk [87] tests. The Lilliefors test is sensitive to both the shape and location of distributions, as it compares the cumulative distribution function (CDF) of the data to a Gaussian CDF.

The Lilliefors test statistic, similar to the the Kolmogorov-Smirnov test statistic, is the maximum difference between the CDF of a Gaussian distribution and the CDF of the data in question. Unlike the Kolmogorov-Smirnov test, the mean and variance of the Gaussian distribution are estimated from the data. The smaller the Lilliefors test statistic, the closer the data conform to a Gaussian distribution. The Shapiro-Wilk test statistic is calculated differently, however (please see the reference [87]). The closer the Shapiro-Wilk test statistic to a value of 1, the closer the data conform to a Gaussian distribution.

The p-value (also known as the asymptotic significance) is used in statistical hypothesis testing and is the probability of obtaining a test statistic result at least as extreme (as the one in question) if the null hypothesis (claim under test) is correct. A smaller p-value signifies stronger evidence for rejecting the null hypothesis, while a larger p-value signifies poorer evidence for rejecting the null hypothesis. For the normality tests in this section, the null hypothesis is that the data conform to a Gaussian distribution. Therefore, if a significance threshold is set as $p > 0.05$, the p-value would have to be less than or equal to 0.05 for the null hypothesis to be considered rejected.

These two tests are applied to three of the nine sources from Dataset 2. Each test is applied to all the instances (recordings) at each time-step. That is, for each test at each time-step, a single value is supplied from each recording. The results are displayed for three of the sources in Figures 3.10, 3.11 and 3.12. It is evident that the data essentially do not conform to a Gaussian distribution across all the recordings at different time-steps.

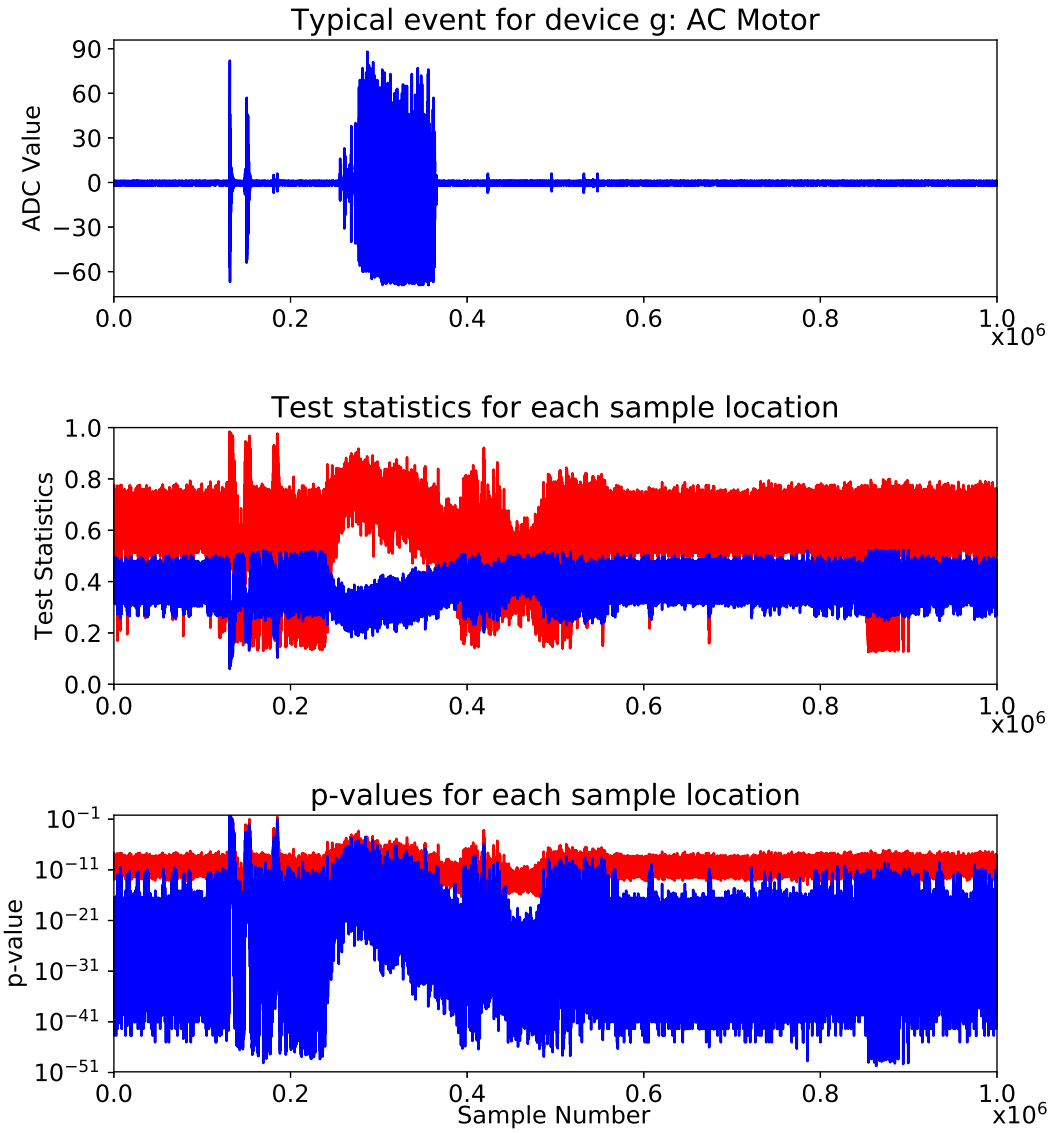

Figure 3.10: The results of normality testing applied across recordings at each time-step for device g (AC motor). The top plot provides examples of a single raw recording, while the test statistics for the Lilliefors and Shapiro-Wilk tests are given in the middle plot. The red (upper) curve shows the Shapiro-Wilk statistic, while the blue (lower) curve shows the Lilliefors statistic. The bottom plot illustrates the corresponding p-values; again the red curve belongs to the Shapiro-Wilk test and the blue curve to the Lilliefors test.

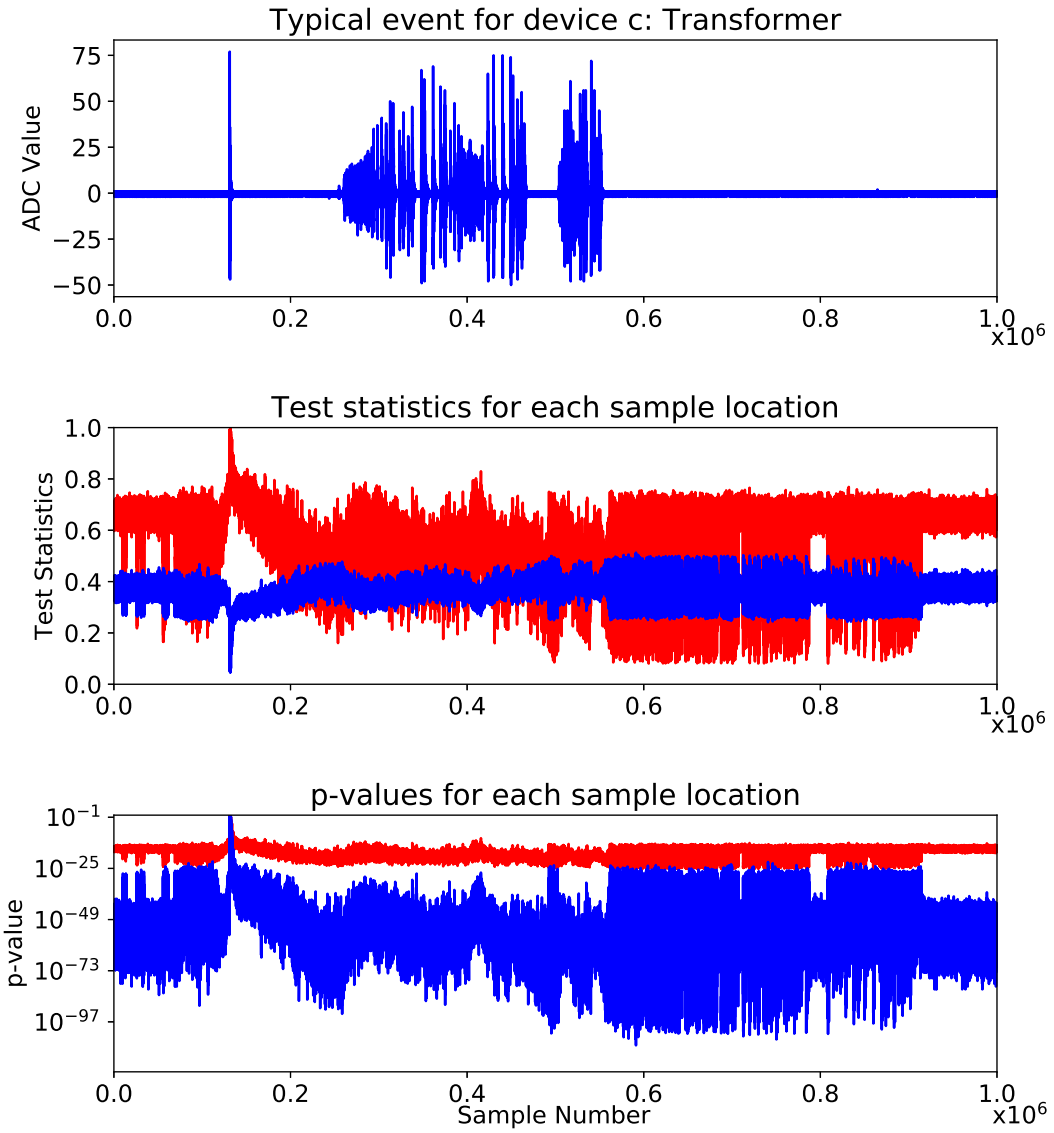

Figure 3.11: The results of normality testing applied across recordings at each time-step for device c (transformer). The top plot provides examples of a single raw recording, while the test statistics for the Lilliefors and Shapiro-Wilk tests are given in the middle plot. The red (upper) curve shows the Shapiro-Wilk statistic, while the blue (lower) curve shows the Lilliefors statistic. The bottom plot illustrates the corresponding p-values; again the red curve belongs to the Shapiro-Wilk test and the blue curve to the Lilliefors test.

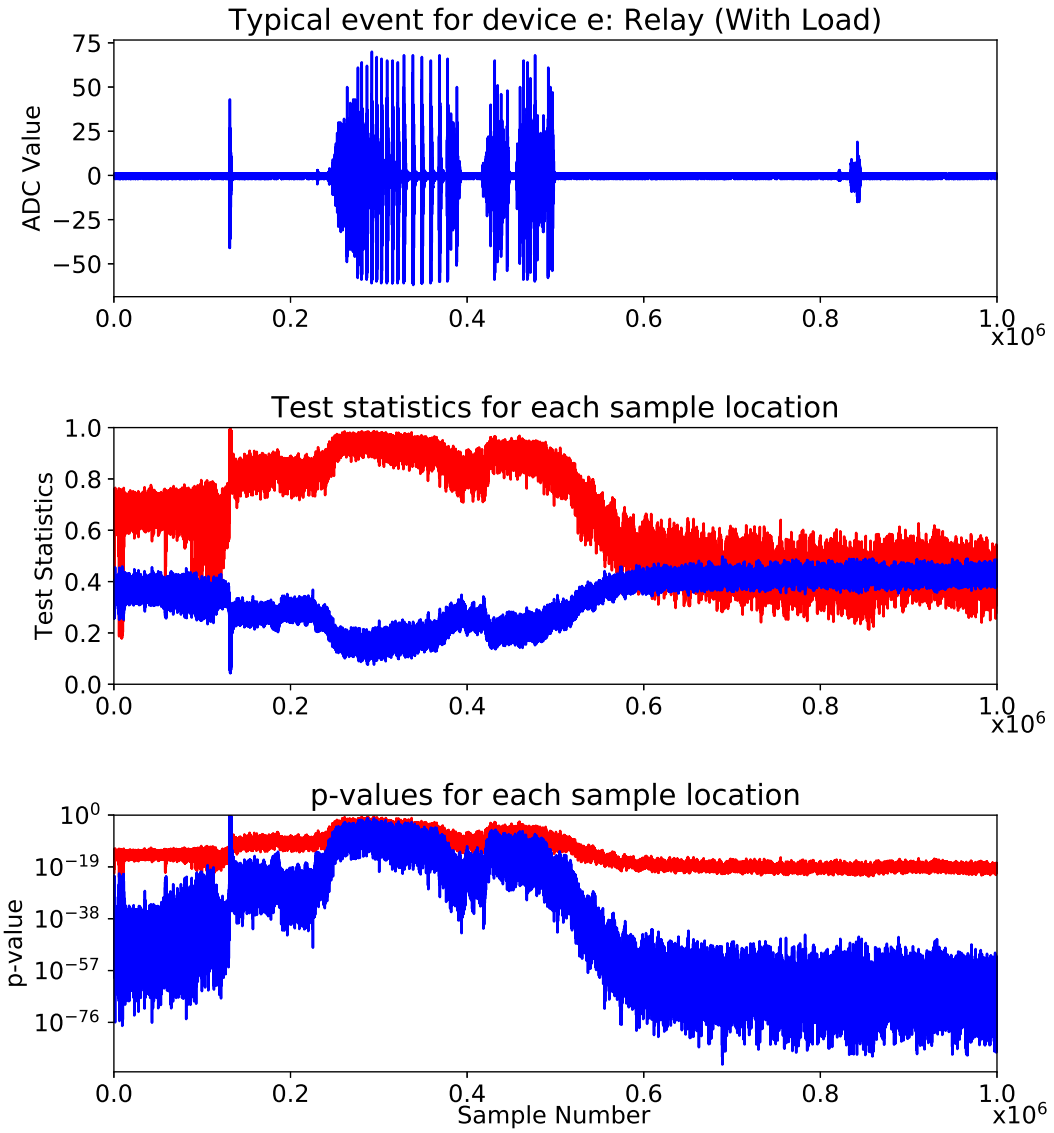

Figure 3.12: The results of normality testing applied across recordings at each time-step for device e (mechanical relay with load). The top plot provides examples of a single raw recording, while the test statistics for the Lilliefors and Shapiro-Wilk tests are given in the middle plot. The red (upper) curve shows the Shapiro-Wilk statistic, while the blue (lower) curve shows the Lilliefors statistic. The bottom plot illustrates the corresponding p-values; again the red curve belongs to the Shapiro-Wilk test and the blue curve to the Lilliefors test.

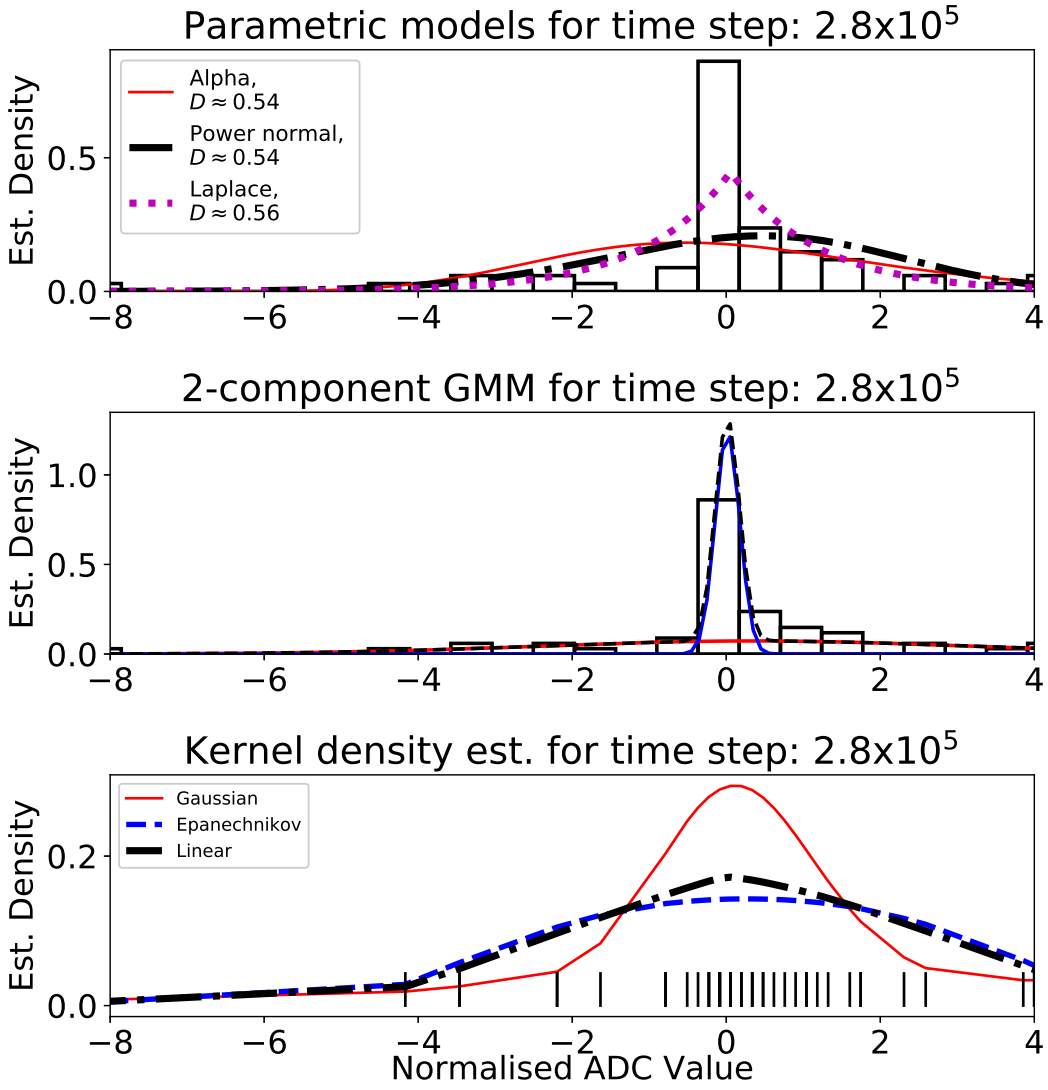

Figure 3.13: Modelling attempts for several time-steps across recordings for device g (AC motor). The top plot illustrates parametric modelling attempts, and provides the Kolmogorov-Smirnov test statistic, D, for each. Two-component Gaussian mixture models (GMMs) are shown in the middle plot, and kernel density estimation results in the bottom plot. The vertical black lines on the x-axes indicate recorded ADC values (each value corresponding to a different recording) for the time-step in question.

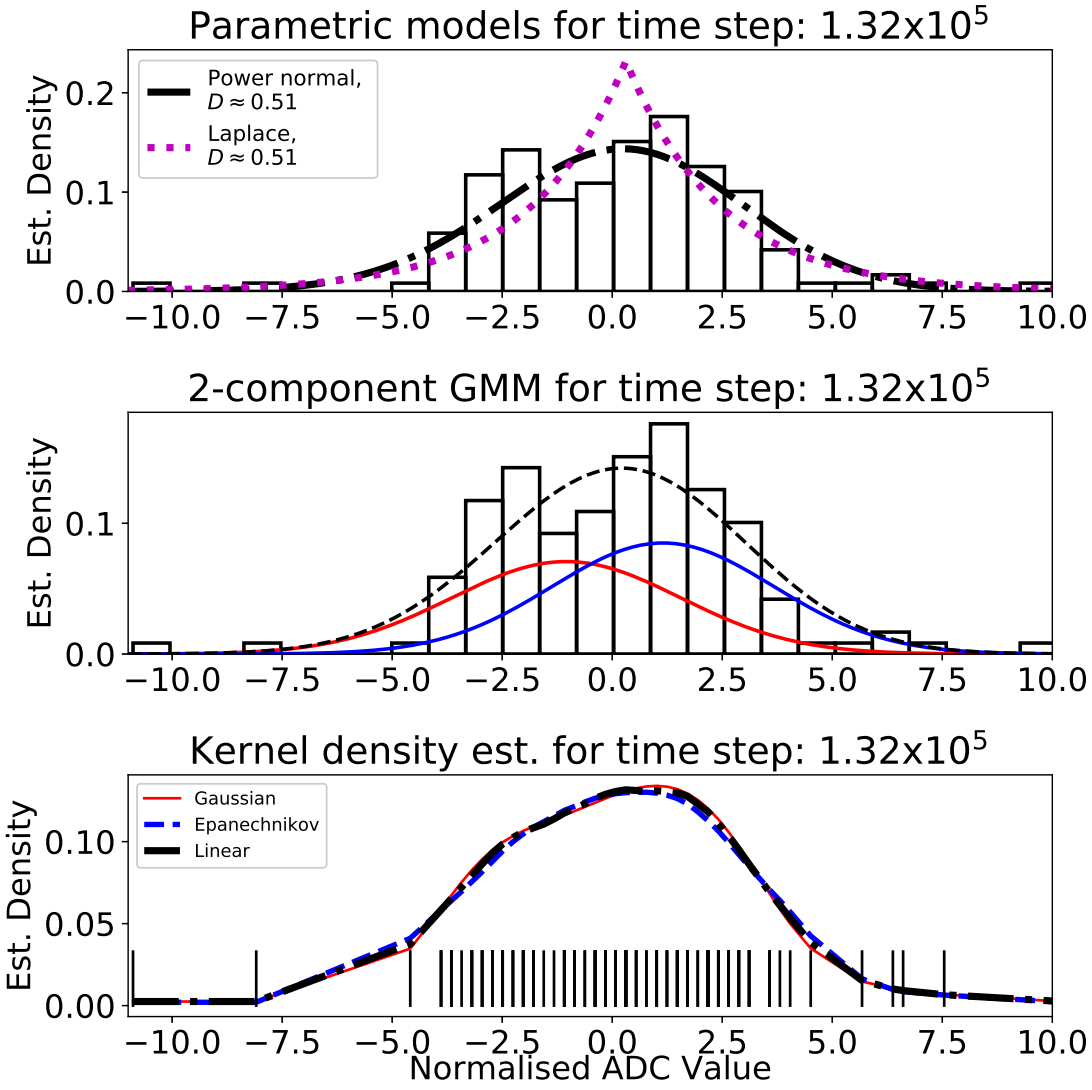

Figure 3.14: Modelling attempts for several time-steps across recordings for device c (transformer). The top plot illustrates parametric modelling attempts, and provides the Kolmogorov-Smirnov test statistic, D, for each. Two-component Gaussian mixture models (GMMs) are shown in the middle plot, and kernel density estimation results in the bottom plot. The vertical black lines on the x-axes indicate recorded ADC values (each value corresponding to a different recording) for the time-step in question.

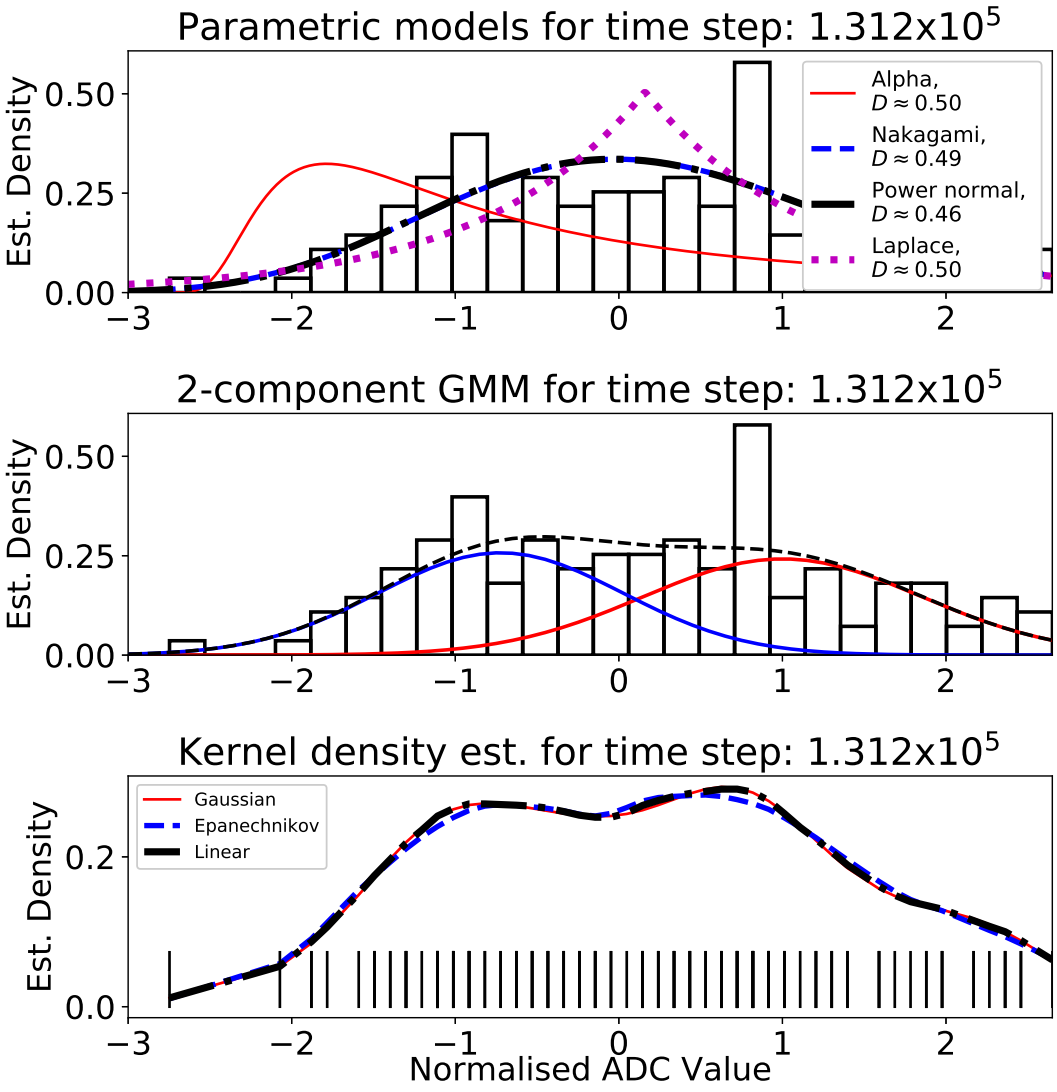

Figure 3.15: Modelling attempts for several time-steps across recordings for device e (mechanical relay with load). The top plot illustrates parametric modelling attempts, and provides the Kolmogorov-Smirnov test statistic, D, for each. Two-component Gaussian mixture models (GMMs) are shown in the middle plot, and kernel density estimation results in the bottom plot. The vertical black lines on the x-axes indicate recorded ADC values (each value corresponding to a different recording) for the time-step in question.

As the recordings effectively do not conform to a Gaussian distribution at every time-step, several attempts are made to find a suitable distribution. A number of parametric models are tested, along with several non-parametric methods. The Nakagami, Alpha, Laplace and Power Normal distributions are tested at each time-step across instances (recordings). Goodness of fit is evaluated using the Kolmogorov-Smirnov two-sample test. As illustrated in the results in Figures 3.13, 3.14 and 3.15, none of these distributions pass for $p > 0.05$. Nevertheless, the Kolmogorov-Smirnov test statistic, D, is included for each distribution that was tested. It provides an indication of goodness of fit: If D is close to 0, then the tested distribution is a close fit for the distribution of the data. If D is close to 1, then the tested distribution is a poor fit.

Next, two-component Gaussian mixture models are attempted. As with the parametric distributions, they do not pass the Kolmogorov-Smirnov two-sample test. Finally, kernel density estimation is applied. Gaussian, linear and Epanechnikov [88] kernels are tested. Their bandwidths are selected using thirtyfold cross validation in a parametric search.

3.4 Conclusion

In this chapter, several useful datasets are presented. Each consists of labelled time domain recordings of transient RFI signals and in each case, a wide recording bandwidth was used.

The first of the datasets was recorded at the MeerKAT site itself and provides insight into the nature of such transient RFI signals. However, for a number of reasons it is of limited use for training and evaluating classification algorithms. The second dataset is much more extensive and was recorded specifically to address the shortcomings of the first. In the third dataset, transient RFI signals were recorded simultaneously with the mains supply voltage to enable a particular investigation to be carried out (detailed in Chapter 4).

Finally, a statistical analysis of the signals recorded in Dataset 2 is presented. It is determined that the data do not conform to a Gaussian distribution at every time-step across repeated recordings of the same source. Attempts are also made to fit a distribution to the data at each time-step across recordings. A variety of

parametric distributions are unsuccessful, so non-parametric techniques such as kernel density estimation are applied as well.

The finding that the data in essence do not conform to a Gaussian distribution (across recordings) has implications for the choice of components analysis technique for feature selection. This is investigated further in the next chapter.

Chapter 4

Components Analysis Techniques[1]

The recordings of transient RFI events obtained in Chapter 3 are in excess of 10^6 time-steps in length. Given the number of recordings involved, such large lengths render many classification techniques computationally impractical. Thus, a feature selection stage is sought, to elicit the most useful features with which to discriminate between classes.

In this chapter, components analysis techniques are considered as a potential feature selection step in the classification of transient RFI. Ordinary PCA would be a standard place to start. PCA reduces the number of dimensions by projecting the data onto a set of orthogonal axes. The first of these axes points in the direction of greatest variance in the data. The next axis points in the direction (orthogonal to the first axis) accounting for the next-greatest variance in the data. Subsequent axes follow in the same manner. Typically, a relatively small number of axes accounts for most of the variance in the data. Many of the remaining axes, which account for comparatively small amounts of variance, may then be discarded. The most important discriminating features (which account for most of the variance) are retained. A typical rule of thumb would be to retain

[1]This chapter is based in part upon the following publications:
Czech, D, A Mishra, and M Inggs. "Characterizing transient radio-frequency interference." Radio Science 52 (2017): 841-851.
Czech, D, A Mishra, and M Inggs. "A dictionary approach to identifying transient RFI." Radio Science 53 (2018): 656-669.

the first axes (also called components) which together account for the first 80% of the variance.

One caveat is that standard PCA assumes the data conform to a multivariate Gaussian distribution. As determined in the previous chapter, this is not the case for the transient RFI events analysed here. Nonlinear variants of PCA have been developed which do not assume such a distribution, and thus would be expected to perform better than standard PCA. In this chapter, one such variant, kernel PCA, is compared with standard PCA.

4.1 Standard PCA

Consider $\boldsymbol{A_r}$, the d by l matrix consisting of l instances (individual recordings) of d time-steps in length. To calculate the full covariance matrix $\boldsymbol{C}$ of $\boldsymbol{A_r}$ directly, at least as many instances as there are time-steps in each instance would be required. Since there are 10^6 time-steps in each instance, obtaining such a large dataset would be impractical in this case. Fortunately, PCA can be computed using a different method which sidesteps this problem. This well-established approach is described in work by Turk and Pentland [89]. In standard PCA, the eigenvectors $\boldsymbol{E}$ of the covariance matrix $\boldsymbol{C}$ are needed. In Turk and Pentland's approach, a different matrix, $\widetilde{\boldsymbol{C}}$, is used instead. It is calculated as follows:

$$\widetilde{\boldsymbol{C}} = \widetilde{\boldsymbol{A}}_{\boldsymbol{r}}{}^{T} \widetilde{\boldsymbol{A}}_{\boldsymbol{r}} \tag{4.1}$$

Here, $\widetilde{\boldsymbol{A}}_{\boldsymbol{r}}$ is calculated by subtracting the average instance of shape $(d, 1)$ from each and every instance in $\boldsymbol{A}$. Next, the eigenvectors of $\widetilde{\boldsymbol{C}}$ are obtained:

$$\widetilde{\boldsymbol{C}}\boldsymbol{E}' = \boldsymbol{\lambda}'\boldsymbol{E}' \tag{4.2}$$

$\widetilde{\boldsymbol{C}}$ can be substituted for $\widetilde{\boldsymbol{A}}_{\boldsymbol{r}}{}^{T} \widetilde{\boldsymbol{A}}_{\boldsymbol{r}}$ (equation 4.1):

$$\widetilde{\boldsymbol{A}}_{\boldsymbol{r}}{}^{T} \widetilde{\boldsymbol{A}}_{\boldsymbol{r}}\boldsymbol{E}' = \boldsymbol{\lambda}'\boldsymbol{E}' \tag{4.3}$$

Multiplying each side by $\widetilde{\boldsymbol{A}}_{\boldsymbol{r}}$,

$$\widetilde{\boldsymbol{A}}_r \widetilde{\boldsymbol{A}}_r{}^T \widetilde{\boldsymbol{A}}_r \boldsymbol{E}' = \boldsymbol{\lambda}' \widetilde{\boldsymbol{A}}_r \boldsymbol{E}' \tag{4.4}$$

from which the following is obtained:

$$\boldsymbol{C} \widetilde{\boldsymbol{A}}_r \boldsymbol{E}' = \boldsymbol{\lambda}' \widetilde{\boldsymbol{A}}_r \boldsymbol{E}' \tag{4.5}$$

The eigenvectors of $\boldsymbol{C}$ are then given by $\widetilde{\boldsymbol{A}}_r \boldsymbol{E}'$, and it was not necessary to compute the covariance matrix $\boldsymbol{C}$ directly from $\boldsymbol{A}$.

4.2 Kernel PCA

Kernel PCA [90] is a nonlinear version of PCA that improves the prospect of cluster separation by creating a nonlinear mapping of the input space. The nonlinear mapping allows the principal components to be computed in a higher dimensional space, in which it is likely easier to linearly separate clusters. A kernel function is used:

$$\boldsymbol{K}(x_i, x_i) = \phi(x_i)\phi(x_i)^T \tag{4.6}$$

Here x_i is an input vector (instance) of dimension d. In this chapter, several different kernel functions are applied (see Table 4.1). The goal is to compute PCA in the higher dimensional space without ever needing to evaluate $\phi(x_i)$ explicitly. In standard PCA, components are calculated by computing the eigenvectors and eigenvalues of the covariance matrix. Due to the kernel function mapping in kernel PCA, a different covariance matrix is calculated [90] as follows:

$$\boldsymbol{C} = \frac{1}{l} \sum_{i=1}^{l} \phi(x_i)\phi(x_i)^T \tag{4.7}$$

x_i is a single individual instance of length d. The eigenvectors and eigenvalues are again computed:

$$\boldsymbol{\lambda E} = \boldsymbol{CE} \tag{4.8}$$

As described by Schölkopf et al [90], (4.8) is equvalent to:

$$\boldsymbol{\lambda}(\phi(x_i) \cdot \boldsymbol{E}) = (\phi(x_i) \cdot \boldsymbol{CE}) \tag{4.9}$$

for $i = 1$ to l. The matrix of eigenvectors $\boldsymbol{E}$ can be represented as follows:

$$\boldsymbol{E} = \sum_{i=1}^{l} \alpha_i \phi(x_i) \tag{4.10}$$

By using equations (4.6), (4.7), (4.9) and (4.10), the following is obtained:

$$l\boldsymbol{\lambda K\alpha} = \boldsymbol{K}^2\boldsymbol{\alpha} \tag{4.11}$$

To find $\boldsymbol{\alpha}$, $l\boldsymbol{\lambda\alpha} = \boldsymbol{K\alpha}$ is solved. Then, the principal components may be computed for an instance x:

$$\phi(x)^T \boldsymbol{E} = \sum_{i=1}^{l} \alpha_i \boldsymbol{K}(x_i, x) \tag{4.12}$$

One final consideration is that the matrix may not be centred (that is, the projected dataset may not have a mean of zero). As described in [90] and [91], this can be achieved using the Gram matrix:

$$\widetilde{\boldsymbol{K}} = \boldsymbol{K} - 1_l\boldsymbol{K} - \boldsymbol{K}1_l + 1_l\boldsymbol{K}1_l \tag{4.13}$$

The centred matrix is given by $\widetilde{\boldsymbol{K}}$. 1_l is a matrix of shape (l, l) and each of its elements is $\frac{1}{l}$.

4.3 Cluster Separability Measures

Several separability measures are used to evaluate and compare the different components analysis techniques investigated in this chapter. Well formed and separated clusters are desired, each of which should contain instances from only one of the different classes.

4.3.1 Geometric Separability Index

The geometric separability index (GSI) takes a Euclidean approach in evaluating class separation. In the binary case, Thornton's separability index [92] is calculated as the fraction of instances which share their class with the majority of their k nearest neighbours [93]. Here, Thornton's separability index is modified to work with multiple classes as follows: Each instance x_i belongs to a class $\boldsymbol{t}$. x_j is the jth nearest instance to x_i. Defining

$$f\left(x_i, x_j\right) = \begin{cases} 1 & x_i,\ x_j \in \boldsymbol{t}_p \\ 0 & x_i \in \boldsymbol{t}_p,\ x_j \in \boldsymbol{t}_q,\ p \neq q \end{cases} \tag{4.14}$$

then

$$GSI = \frac{1}{l} \sum_{i=1}^{l} \texttt{max} \left\{ h \in \mathbb{Z} \mid h \leq \frac{1}{2} + \frac{1}{k} \sum_{j=1}^{k} f\left(x_i, x_j\right) \right\} \tag{4.15}$$

The GSI varies between 0 and 1, where 0 indicates overlapping clusters and 1 indicates separated clusters.

4.3.2 Cohen's Kappa

Class separation may also be evaluated indirectly using Cohen's Kappa [94] by first applying a kNN classifier to each instance and comparing the results to the ground truth. The advantage of using Cohen's Kappa is that it takes chance agreement between two raters into account. Cohen's Kappa is calculated as follows:

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \tag{4.16}$$

Here, p_0 is the agreement between two raters and p_e is the probability of agreement between them by chance. If κ is close to 0, classes are poorly separated, no better than would be expected by chance. Conversely, if κ is close to 1, then classes are well separated.

4.3.3 Silhouette Score

The silhouette coefficient [95] is calculated as follows:

$$s_s = \frac{b - a}{max(a, b)} \tag{4.17}$$

a is the average distance from a particular instance to all the other instances in its class, while b is the average distance from the instance to all the instances belonging to the nearest different class. Overlap and poorly formed clusters are indicated by a coefficient of -1 , whereas a coefficient near 1 indicates good, well-separated clusters.

4.4 Application to Transient RFI

In this section, standard PCA and kernel PCA are applied directly to the signals in Dataset 2 (See Chapter 3). Fig. 4.1 illustrates the results for three components. Next, kernel PCA is applied to the data. It is established in Chapter 3 that the distribution of the data is principally non-Gaussian (across instances in each class). Therefore, nonlinear variants of PCA (such as kernel PCA) should perform better than standard PCA. A number of variables affecting cluster separation in PCA and kernel PCA are investigated. The effect of varying the number of components (retained for classification) is investigated in Fig. 4.2. The peak accuracy of standard PCA is inferior to that of kernel PCA with RBF, sigmoid and third-order polynomial kernel functions.

Table 4.1: The different kernel functions used with kernel PCA.

Kernel	**Equation**
Radial basis function	$K(x, y) = e^{-\gamma\|x-y\|^2}$
Sigmoid	$K(x, y) = tanh(\gamma x^T y + u_0)$
Polynomial	$K(x, y) = (\gamma x^T y + u_0)^z$

In Fig. 4.3, the separability index is plotted as a function of γ for kernel PCA applied to the data using different kernel functions. Using components

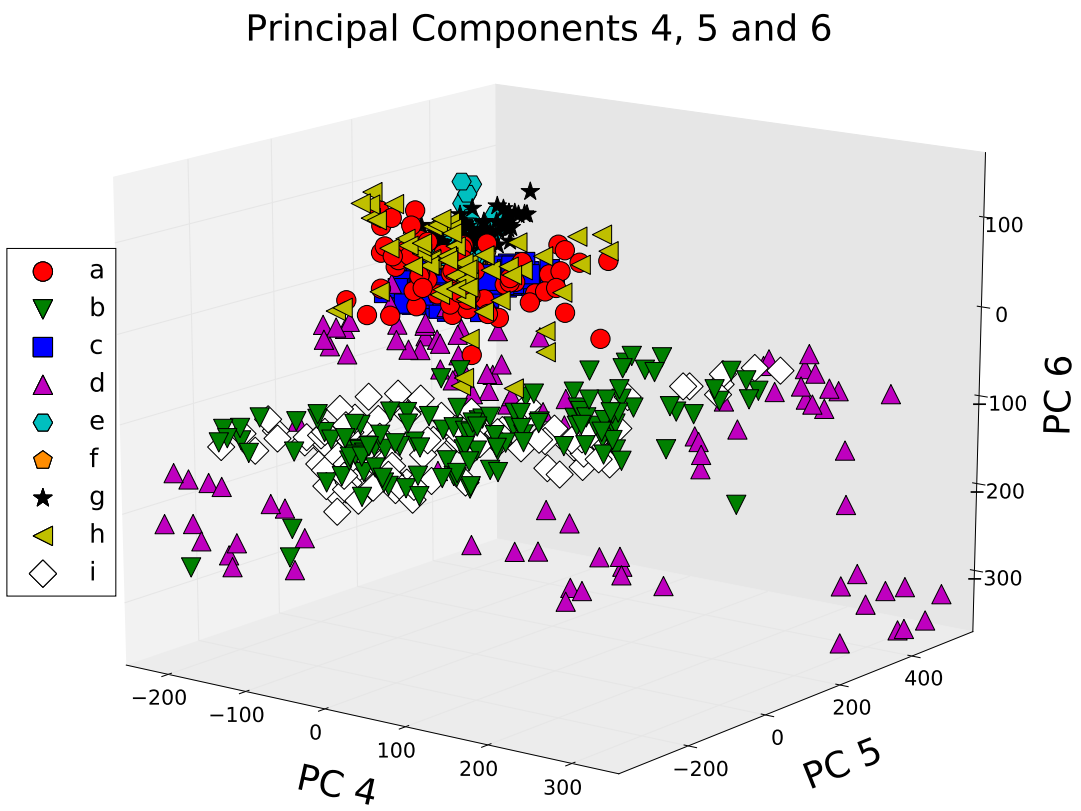

Figure 4.1: Standard PCA applied to Dataset 2. Each marker represents a single transient RFI signal. The colour and shape of the marker indicate the source of the signal.

3 to 20 (a reasonable choice given Fig. 4.2), an accuracy of 0.83 is obtained for a radial basis kernel function with $\gamma \approx 4.92 \times 10^{-7}$. This is better than that which is obtained using standard PCA (see Fig. 4.2). As predicted, kernel PCA outperforms standard PCA in this case. Fig. 4.4 illustrates three of the components obtained via kernel PCA using these parameters.

A deeper look at the separation between classes is provided in Fig. 4.5. Cohen's Kappa (Section 4.3.2) is calculated between each pair of classes for standard PCA and kernel PCA, after the application of a basic k-nearest neighbours algorithm.

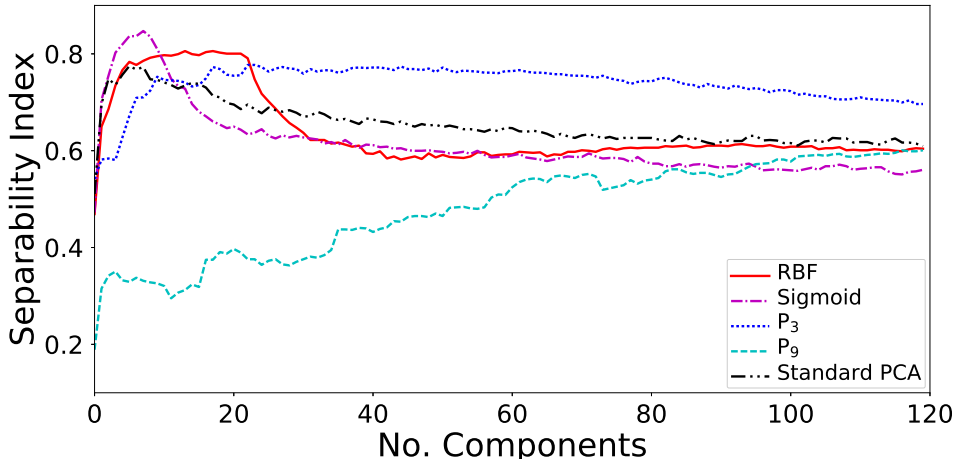

Figure 4.2: The geometric separability index calculated when applying PCA and kernel PCA with various different kernel functions. The number of retained components is varied for each technique. P_3 and P_9 are third and ninth order polynomial kernel functions respectively and RBF denotes a radial basis function. The equations for these kernel functions are provided in Table 4.1.

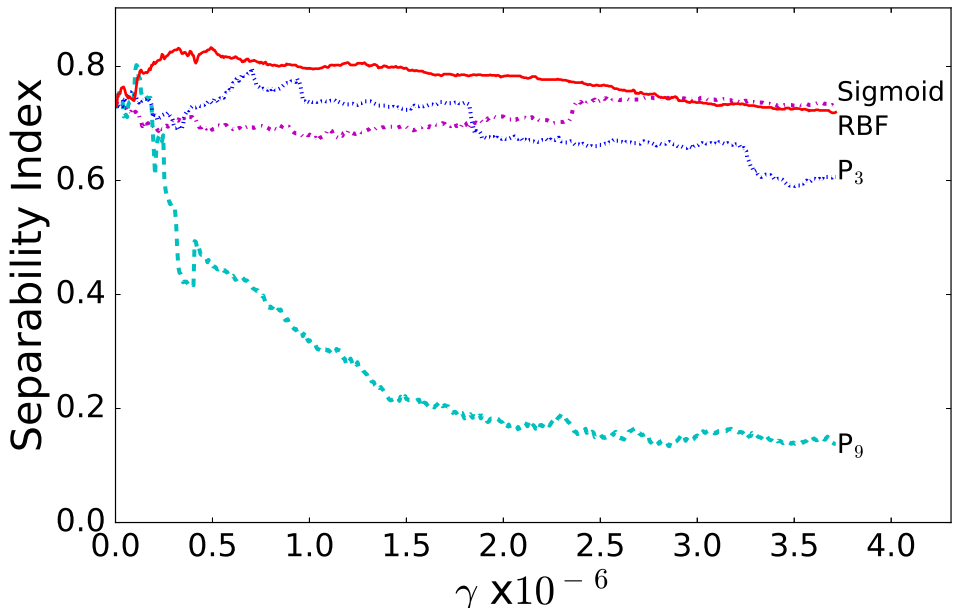

Figure 4.3: The geometric separability index calculated when using the different kernel functions, plotted as a function of γ (a parameter for each; see Table 4.1). RBF = radial basis function. P_3 and P_9 are third and ninth order polynomial functions, respectively.

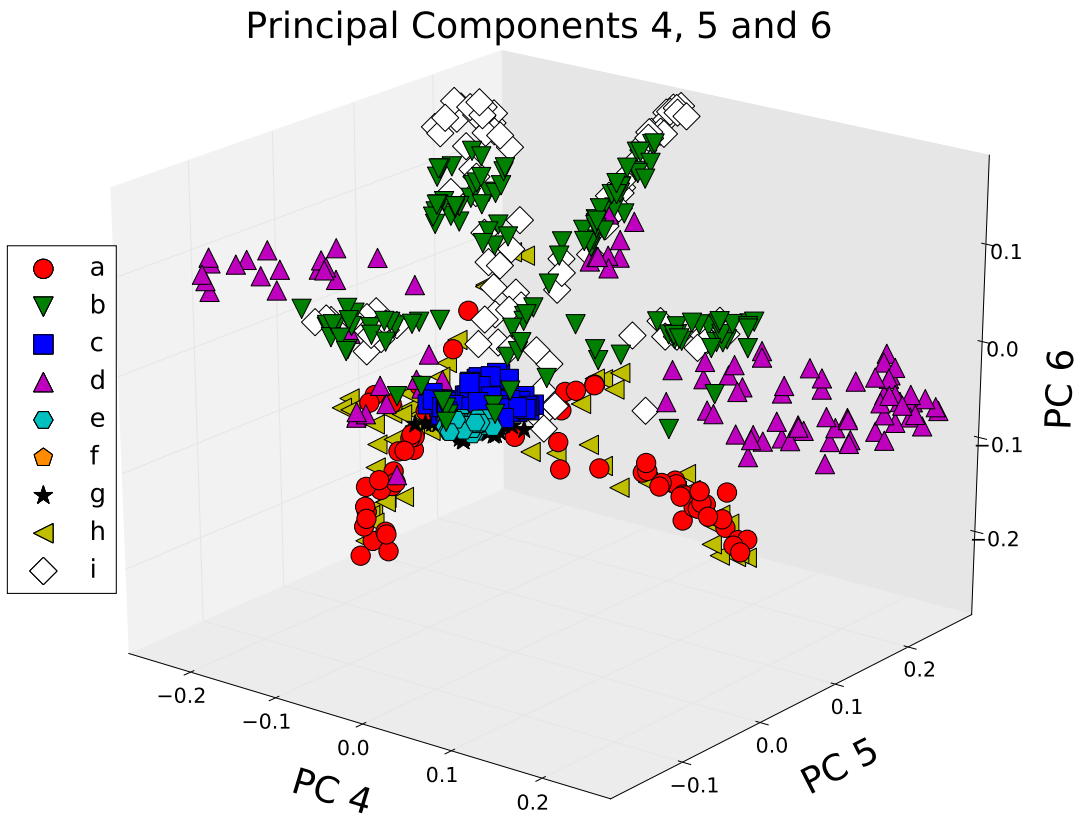

Figure 4.4: Kernel PCA applied to Dataset 2 using a radial basis kernel function with parameter $\gamma \approx 4.92 \times 10^{-7}$. The colour and shape of each marker represents the class to which it belongs. Each marker represents a single recording of a transient RFI signal.

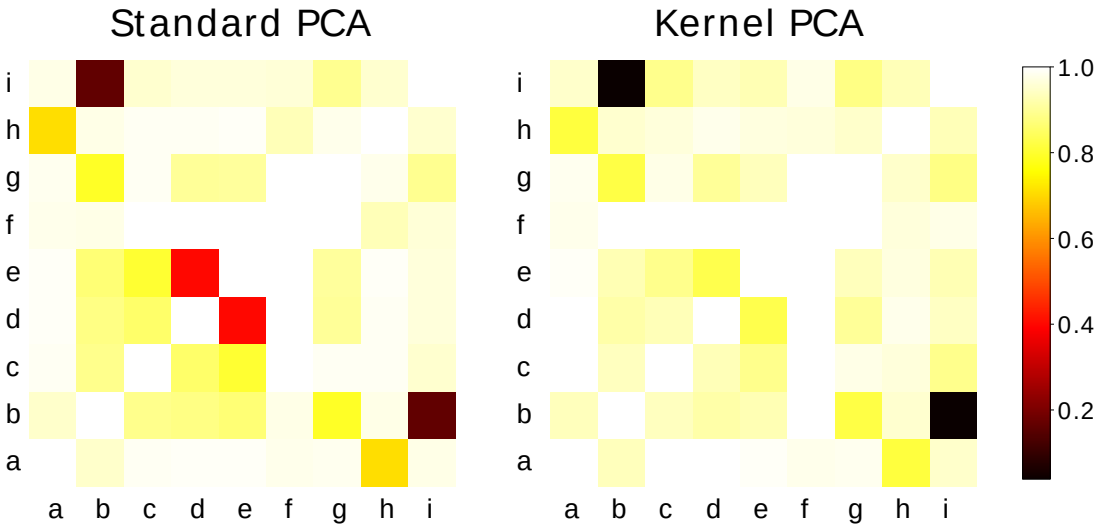

Figure 4.5: Confusion matrices illustrating class separation for standard PCA and kernel PCA (with a radial basis kernel function and $\gamma \approx 4.92 \times 10^{-7}$). The confusion matrices indicate Cohen's Kappa, calculated for each pair of sources after applying a k-nearest neighbours approach. In most cases, kernel PCA offers better class separation than standard PCA. Devices a and h are the least separated classes, which might be expected since they are each a different brand of compact fluorescent lamp (see Table 4.1).

4.5 Basic Source Classification

Next, an attempt is made to classify the RFI signals by source using two basic approaches. After feature selection using kernel PCA, a linear support vector machine (SVM) and a kNN classifier are applied and compared. For this test, both the CFL classes are combined, since they exhibit extremely low class separation in Fig 4.5. Determining the brand of CFL is of secondary importance to accurately detecting signals from CFLs in general.

Prior to classification, the data are divided into a training set, comprising 80% of the data, and a testing set comprised of the remaining 20%. The datasets are also stratified, so that each class is randomly divided according to the same ratio. After the kernel PCA step, the top components which together explain 80% of the variance in the data are retained (excepting the top three) for classification.

The value of k for the kNN classifier is selected via four-fold cross validation using only the training dataset. The best classification result (on the unseen test

set) is obtained for $k = 1$. It is perhaps not surprising that the accuracy (for $k = 1$) is only 18.95%, since classification decisions are made based only on one neighbouring point.

The linear SVM classifier offers better results, with an overall classification accuracy of 61.58%. These results obtained via this basic, direct approach are compared with more sophisticated methods in Chapters 5 and 6.

4.6 Effect of AC Supply Phase on Cluster Separation

In this section, a potential reason behind the lacklustre results obtained in Section 4.5 is investigated. When recording Dataset 2 (Section 3.1.2 in Chapter 3), a link was suspected between the phase of a source's electrical supply at the instant of switching, and the characteristics of the resultant RFI signal. Prior work has shown that the supply voltage influences the RFI signals generated by certain sources. In [52] the RFI signals generated by a defective thermostat are shown to correlate with supply voltage peaks.

To investigate the influence of a source's supply phase, an experiment was set up to record both the transient RFI signal and the instantaneous supply voltage (see Dataset 3 in Section 3.1.3). One of the aims was to determine if the instantaneous supply voltage corresponded with particular principal components. The switching transients of three devices were recorded, a switching power supply unit, a mechanical relay and a step-down transformer.

Figures 4.6, 4.7 and 4.8 illustrate how the changing supply voltage influences RFI recordings in the principal components domain. It is clear that there is an effect, as events recorded when the supply voltage was at similar levels tend to cluster together. In Fig. 4.6, the supply voltage displays a strong correspondence with the first principal component. In Fig. 4.7, a correspondence is observed with the third principal component. The effect of the supply voltage is less apparent in Fig 4.8, however. The observed spreads of data points have implications for cluster separability if the spreads from different sources overlap.

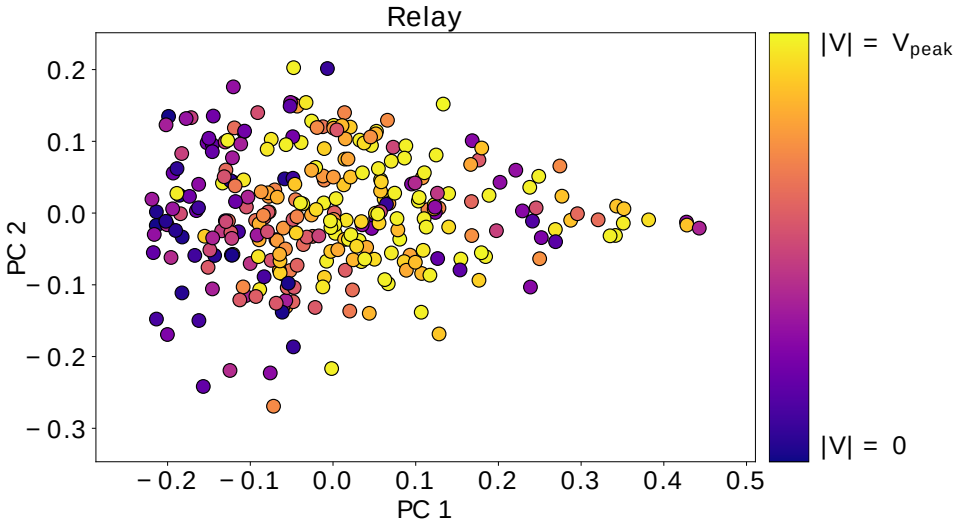

Figure 4.6: Kernel PCA applied to the RFI signals generated when switching an AC mechanical relay. Recorded signals from Dataset 3 are analysed here. The colour of each marker represents the magnitude of the instantaneous AC supply voltage (to the relay) upon signal capture. Each marker itself represents a single recording. The first principal component displays a correspondence with the instantaneous voltage.

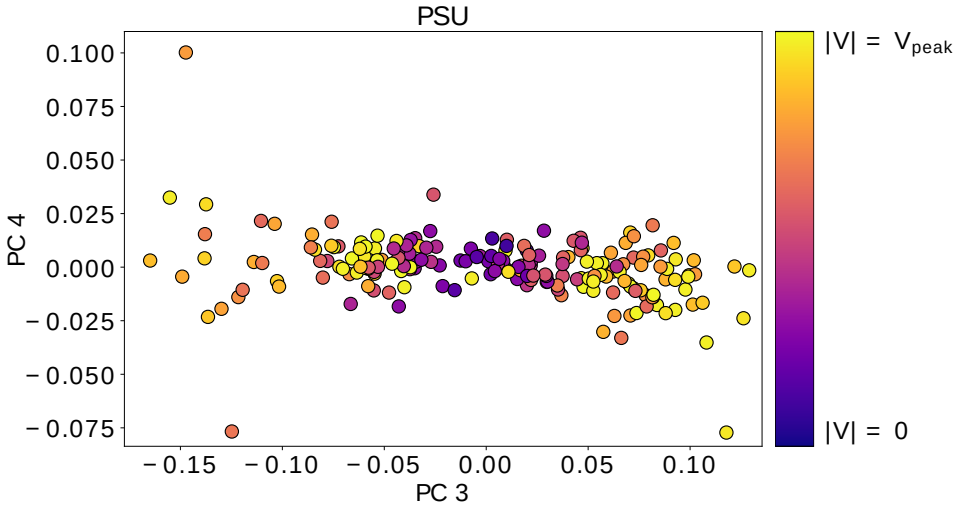

Figure 4.7: Kernel PCA applied to the RFI signals generated when switching a PSU (from Dataset 3). As in Fig. 4.6, each marker represents a single recording and the colour of each marker represents the instantaneous supply voltage magnitude.

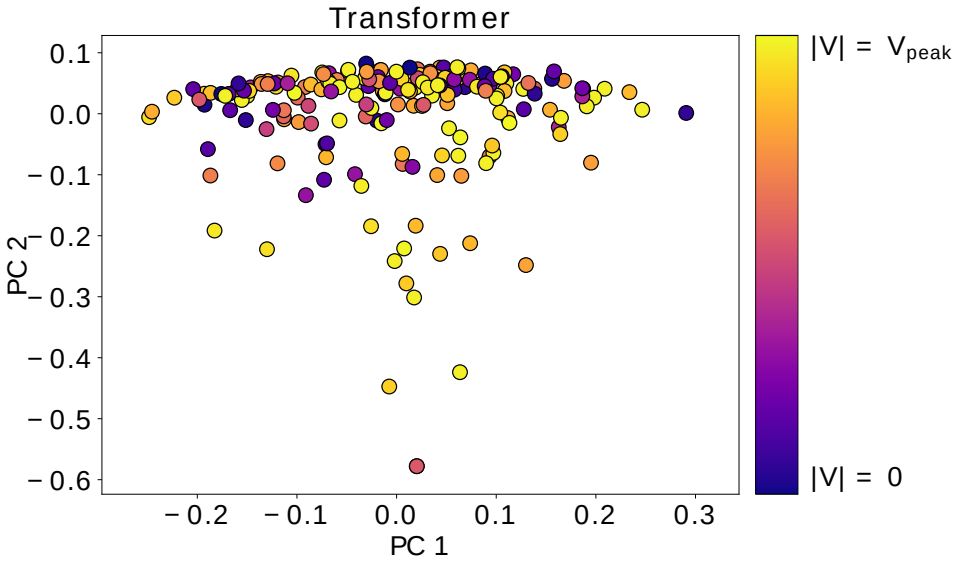

Figure 4.8: Kernel PCA applied to the RFI signals recorded when switching a small step-down transformer (also from Dataset 3). As in Fig. 4.6, each marker represents a single recording. The colour of each marker represents the magnitude of the instantaneous supply voltage.

In Figures 4.9, 4.10 and 4.11, the effect of the instantaneous supply voltage on cluster separation is illustrated for two sources. The data are split into two sets, each containing recordings from both sources. The first set only includes recordings taken for $|V| < 0.4V_{peak}$, and the second for $|V| > 0.6V_{peak}$. These cut-off values are somewhat arbitrary and can be varied. The purpose of dividing the data into these two sets is to illustrate how the RFI signals cluster differently depending on whether the supply voltage is near a zero crossing or a peak (when they are recorded).

In each case, the silhouette score is calculated for the top ten principal components to provide an indication of cluster separation. When the supply voltage is near V_{peak}, the silhouette score is 0.24. When it is near the zero-crossing, the silhouette score drops to 0.07. Therefore, cluster separation is influenced by the magnitude of the instantaneous supply voltage in at least some sources. Accounting for the phase of the supply voltage may aid in developing better classification systems for this type of RFI.

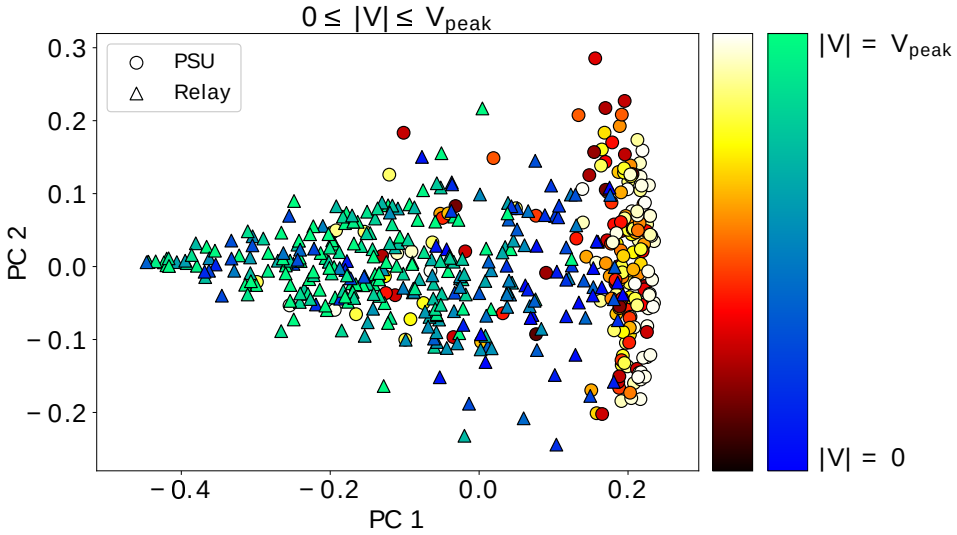

Figure 4.9: The results of applying kernel PCA to RFI signals from both the power supply unit and the relay. Each marker represents a single recording, with the shape indicating the source from which it was recorded. The colour of each marker indicates the magnitude of the supply voltage at the instant the signal was captured. All the recordings are included in this plot irrespective of the supply voltage when captured.

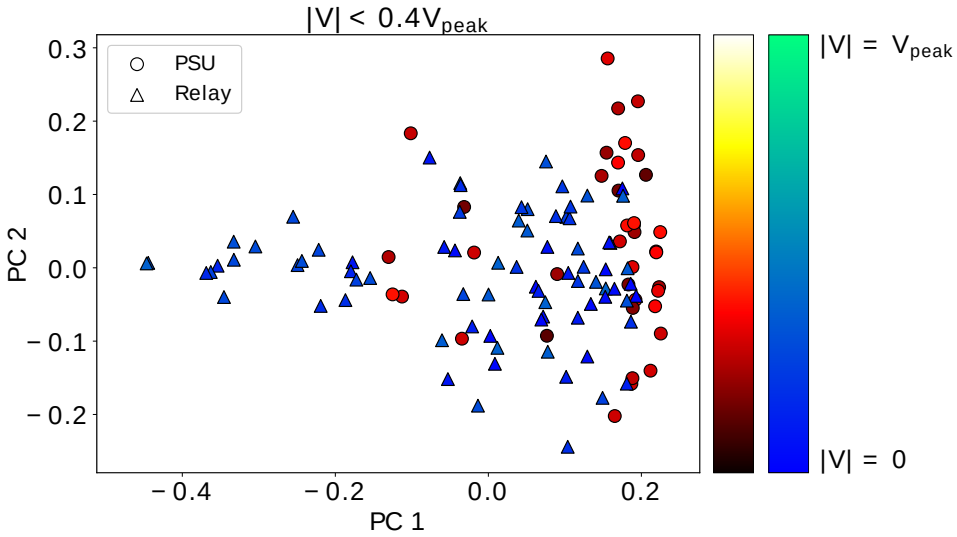

Figure 4.10: Of the recordings (projected using kernel PCA) in Fig. 4.9, only those for which $|V| < 0.4V_{peak}$ are plotted here. Cluster overlap is visible and the silhouette score (computed using the top 10 principal components) is only 0.07.

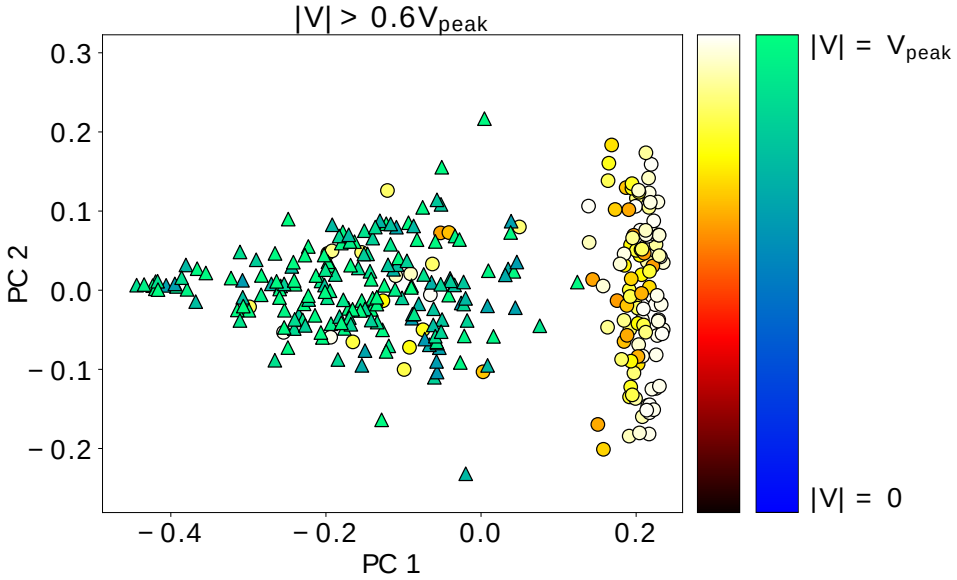

Figure 4.11: Only the recordings (from Fig. 4.9) for which $|V| > 0.6V_{peak}$ are plotted here. Cluster overlap is less than in Fig. 4.10. The silhouette score (computed using the top 10 principal components) is 0.24.

4.7 Conclusion

In this chapter, standard PCA and a nonlinear variant, kernel PCA, are compared for use as a feature selection step in the classification of transient RFI. Several different kernel functions are compared when applying kernel PCA. In Chapter 3, the transient RFI recordings in Dataset 2 are found to be essentially non-Gaussian across recordings. Given that standard PCA assumes a Gaussian distribution, it is therefore expected that kernel PCA would be more suitable. This proved to be the case, as kernel PCA with a radial basis kernel function (see Table 4.1) offers better results.

Next, kernel PCA is used as a feature selection step for the classification of transient RFI. Following kernel PCA, both a linear SVM classifier and a kNN classifier are applied to the data. For evaluation, the data are split into separate training and testing sets. Accuracies of 61.58% and only 18.95% are obtained for the SVM and kNN classifiers respectively. This approach of directly applying kernel PCA followed by a basic classification technique is relatively naïve,

however. These results serve as a benchmark for more sophisticated approaches taken in Chapters 5 and 6.

Finally, the effect of the mains supply voltage phase on cluster separation in the principal components domain is investigated. It is shown that for certain devices, the mains supply voltage accounts for the variance in at least one of the principal components. In addition, for two particular sources, cluster separation improves when the supply voltage nears V_{peak}.

Chapter 5

A Dictionary-Based Approach to Classifying Transient RFI[1]

A dictionary-based approach to identifying the sources of RFI events is proposed in this chapter. The idea of treating the problem in this manner arose when recording Datasets 1 and 2 (see Chapter 3). Both Datasets 1 and 2 show that full transient RFI signals tend to consist of sequences of short individual transients, as illustrated in Figures 1.2, 3.2 and 3.6. However, the recordings in Dataset 1 are too short to capture full sequences. This limitation and others were addressed when recording Dataset 2. Consequently, this chapter deals only with the recordings in Dataset 2.

Repeated recordings of the full RFI signals from a particular source vary from one to the next, but adhere to a common underlying structure. Fig. 3.6 in Chapter 3 provides examples of this phenomenon for two of the sources recorded in Dataset 2. The overarching structure in the sequence of transients from a particular source is in itself an identifying feature. This perspective is supported by the classification results obtained via direct approaches in Chapter 4. In addition, the individual transients within each full sequence differ for each source, potentially providing further identifying features. Examples of individual transients from the different classes are given in Fig. 3.7.

[1]This chapter is based in part upon the following publication:
Czech, D, A Mishra, and M Inggs. "A dictionary approach to identifying transient RFI." Radio Science 53 (2018): 656-669.

The approach to source identification demonstrated in this chapter considers both the sequences and the individual transients themselves, while the strategy proposed in Chapter 6 considers individual transients only. In this chapter, individual transients are assigned labels, and full sequences of such transients are represented as sequences of labels. The aim is to construct a dictionary of individual transients with which any full transient RFI signal may be represented. Representing RFI signals in this manner paves the way for the use of powerful classification techniques such as hidden Markov models (HMMs).

Similar approaches have been taken in other fields, for example in automated speech recognition (ASR). Many ASR methods consider spoken words as sequences of phonemes or other features [71, 96, 97]. While the sequence of phonemes in a spoken word may vary each time it is uttered, the underlying meaning remains the same. HMMs are well-suited to this type of classification problem [96] because they can learn to recognise words despite the inevitable differences between a recorded sequence and the underlying 'true' sequence. Due to their robustness to such variations, HMMs have been applied in a variety of diverse classification tasks. For example, they have been used for handwriting recognition by representing characters from a lexicon of subunits [98]. Such an approach has also been taken for sign language recognition [99] and even the classification of Humpback Whale songs [100].

To analyse full RFI signals as sequences, the transients (of which they consist) must first be extracted. In work from the field of bio-acoustics, a similar task has been faced when classifying cricket songs by species [72]. Like the RFI signals analysed in this chapter, the audio recordings of the cricket songs consist of sequences of individual transients. In their work, Dietrich et al. present an algorithm for extracting individual transients from the cricket songs using a dual-threshold approach. Since the RFI transients studied in this chapter often overlap one another, a different extraction algorithm is proposed in Section 5.2.1. To classify the cricket songs, Dietrich et al. used moving windows to obtain a variety of features. In this chapter however, instead of moving windows, features are extracted using kernel PCA and density based unsupervised clustering techniques. Details on this strategy are provided in Section 5.2.2.

5.1 Preprocessing

The approach proposed in this chapter is developed and tested using Dataset 2. The two CFL classes are combined as before in Chapter 4. Full RFI events are limited to 950000 samples (approximately $593\mu s$) to reduce computational requirements, as nearly all the activity across recordings occurs within this time period. RFI events are also normalised by class. For each recording $\boldsymbol{r}(t)$ in the set of recordings $\boldsymbol{R}$ of a particular source:

$$\boldsymbol{r'}(t) = \frac{\boldsymbol{r}(t) - \mu}{\sigma} \tag{5.1}$$

Here, μ and σ are the mean and standard deviation respectively of all $\boldsymbol{r}(t) \in \boldsymbol{R}$ and $\boldsymbol{r'}(t)$ is the normalised RFI signal. For new, unseen RFI recordings, it will not be possible to pre-emptively normalise signals in this manner. This will not be necessary, however. The only reason for taking this step is to eliminate amplitude differences between classes as a discriminating feature. The variation in amplitude due to a variation in distance from a monitoring receiver should not affect the identification of a source.

5.2 Dictionary Creation

This section delineates the concept of a canonical dictionary that can be used to represent full transient RFI events and identify them. This approach is motivated by the similarities between full RFI events, which consist of sequences of transients, and spoken words, which consist of sequences of phonemes. Like the spoken representation of a word that varies each time it is uttered, so too does a particular RFI signal vary from recording to recording. In Fig. 3.6 in Chapter 3, small variations are apparent in repeated recordings of RFI signals from the same source. However, an underlying structure is apparent. If such a dictionary approach makes sense as described, then some transient labels should be shared by RFI events from different sources. This is in fact the case, as indicated in Table 5.1. Considering RFI events as sequences of transients drawn from a dictionary permits the use of powerful approaches like hidden Markov models.

5.2.1 Automated Transient Extraction

To enable the creation of a dictionary of transients, it is necessary to extract the individual transients from full RFI recordings. As there are more than 900 full recordings, many of which contain several hundred individual transients each, manual extraction is unfeasible. In this section, an automated extraction algorithm is proposed. A small, representative subset of the transients in the full dataset are extracted manually and used as a ground truth for parameter selection.

There are likely many possible approaches to transient extraction, such as the previously developed method (designed for bio-acoustic transients) discussed in the introduction to this chapter [72]. However, a new algorithm (given below) is developed to suit the task of extracting RFI transients. For example, the proposed algorithm is capable of handling the presence of overlapping transients and sequences of immediately adjacent transients (Fig. 5.1). This is necessary since sequences of adjacent and overlapping transients (apparent in Figures 3.2, 3.6, 3.9 and others) are common in the full RFI recordings considered in this thesis.

Extraction Algorithm

The basic procedure is outlined in pseudocode in Algorithm 1 and illustrated in Fig. 5.1. First, the Hilbert transform is applied to the raw ADC output to obtain the envelope of the signal. The absolute value of the result is subsequently smoothed by convolution with a moving window of length L_1. L_1 is empirically selected as described later in this section. The first extraction pass is applied by setting a lenient threshold (T_1) at 1.5% of the maximum range of values in the signal. Since the threshold is lenient, the chance of overlooking any transients is minimised. Where the threshold is exceeded, portions of the signal are retained. Such portions that are closer together than T_M time-steps are merged together. T_M is also selected empirically as described later in this section.

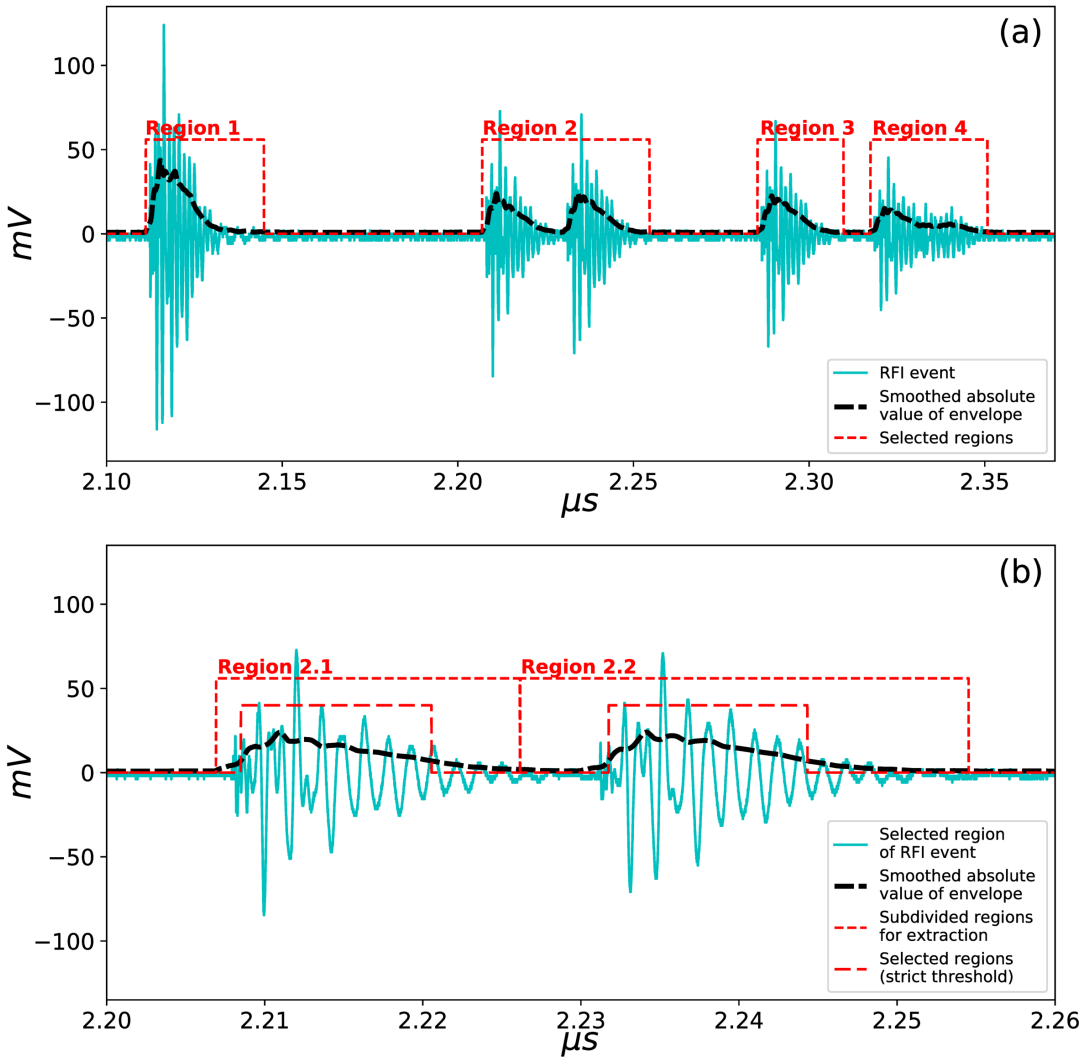

Figure 5.1: The procedure for extracting RFI transients, visualised. In plot (a), a lenient threshold is applied to ensure that entire transients are extracted. However, such a lenient threshold also results in merged transients (for example, in Region 2). To mitigate this, a second, stricter threshold is applied to the extracted regions, shown in (b). The original regions are finally subdivided at the midpoints between subregions selected using the stricter threshold.

Next, a stricter threshold is applied to each of these portions separately, in order to prevent unwanted merging of transients that are close together or overlapping. An adaptive threshold, $T_2(t)$, is applied to each portion. It is obtained by applying a scaled median filter to the smoothed portion that was extracted previously. The choice of window length is not critical as long as it is set a few times larger than L_1; a value of 8000 time-steps is selected here. New subregions are selected from within each original portion where $T_2(t)$ is exceeded (see Fig. 5.1). If the subregions are separated by notable gaps, the original portions are subdivided at the midpoints of these gaps. The original unaltered time-domain signal is then segmented according to these subdivisions, and the regions containing transients extracted.

Parameter Selection

In order to select parameters for the extraction algorithm given above, a ground truth is obtained by extracting a representative subset of the main dataset by hand. The different parameters are then varied against one another and the results evaluated on the manually extracted dataset. Three evaluation metrics are used:

1. The fraction of ground truth transients that are correctly extracted. A ground truth transient is regarded as correctly extracted if more than 60% of the time-steps it occupies are assigned to any extracted region.

2. The number of erroneously extracted regions as a fraction of the number of ground truth transients. An erroneous extraction is deemed to have occurred when less than 30% of an extracted region is occupied by any ground truth transients. This value can exceed 1, since it is possible for the number of extracted regions to exceed the number of transients in the ground truth. This can occur (for example) when many poorly extracted regions do not contain ground truth transients.

3. The fraction of ground truth transients that are erroneously merged together.

Algorithm 1

Input: an RFI event, $\boldsymbol{r}(t)$

Output: Transients extracted from $\boldsymbol{r}(t)$

$\boldsymbol{r_H}(t) \leftarrow |H(\boldsymbol{r})(t)|$

$\boldsymbol{r_s}(t) \leftarrow \boldsymbol{r_H}(t) * \frac{1}{L_1} \times ones(L_1)$

f_mask$(t) \leftarrow$where $\boldsymbol{r_s}(t) > T_1 \times (\max(\boldsymbol{r_s}(t)) - \min(\boldsymbol{r_s}(t))) + \min(\boldsymbol{r_s}(t))$

for each region g_i in f_mask(t) **do**

  while g_i closer than T_M samples to g_{i+1} **do**

    $g_i \leftarrow$merge g_i and g_{i+1}

  end while

  if length $g_i < T_M$ **then**

    delete g_i

  end if

end for

for each region g_i in f_mask(t) **do**

  $\boldsymbol{r_{gi}}(t) \leftarrow$ region g_i extracted from $\boldsymbol{r_s}(\mathrm{t})$

  $T_2(t) \leftarrow$ scaled moving median filter applied to $\boldsymbol{r_{gi}}(t)$

  r_mask$_i(t) \leftarrow$ where $\boldsymbol{r_{gi}}(t) > T_2(t)$

  for each sub-region $g_{\mathrm{sub}j}$ in r_mask$_i(t)$ **do**

    $g_{\mathrm{sub}j} \leftarrow$ subdivide g_i at midpoints between $g_{\mathrm{sub}j}$ and $g_{\mathrm{sub}j+1}$

    $\boldsymbol{u}(t) \leftarrow g_{\mathrm{sub}j}$ extracted from original $\boldsymbol{r}(t)$

    return $\boldsymbol{u}(t)$

  end for

end for

end

Figures 5.2, 5.3, 5.4 and 5.5 display the results of the parameter selection procedure for these three metrics. In particular, Fig. 5.5 illustrates how L_1 and T_M are selected by considering the three metrics together. Three requirements are set when selecting L_1 and T_M, in order of importance:

1. The fraction of extracted transients should exceed 90%.
2. The fraction of erroneously extracted regions should remain less than 5%.
3. The fraction of incorrectly merged transients should also be less than 5%.

In Fig. 5.5, requirement 1 is not satisfied in region **a**. In region **b**, requirement 1 is satisfied, but not requirement 2. Both requirements 1 and 2 (but not requirement 3) are satisfied in region **c**, while all three are simultaneously satisfied only in region **d**. Therefore, values for L_1 and T_M may only be selected from within region **d** in order to satisfy all three requirements.

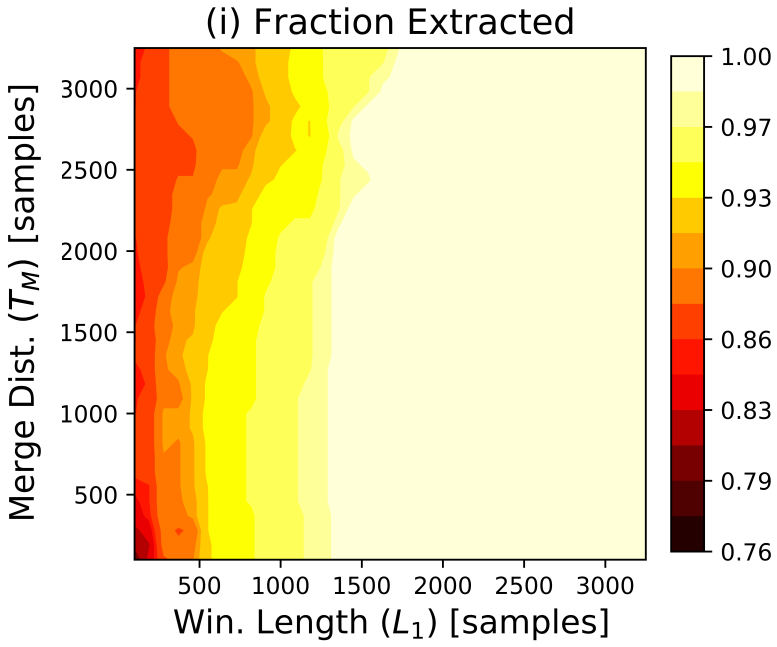

Figure 5.2: The fraction of ground truth transients that are accurately extracted for different values of the two parameters T_M and L_1.

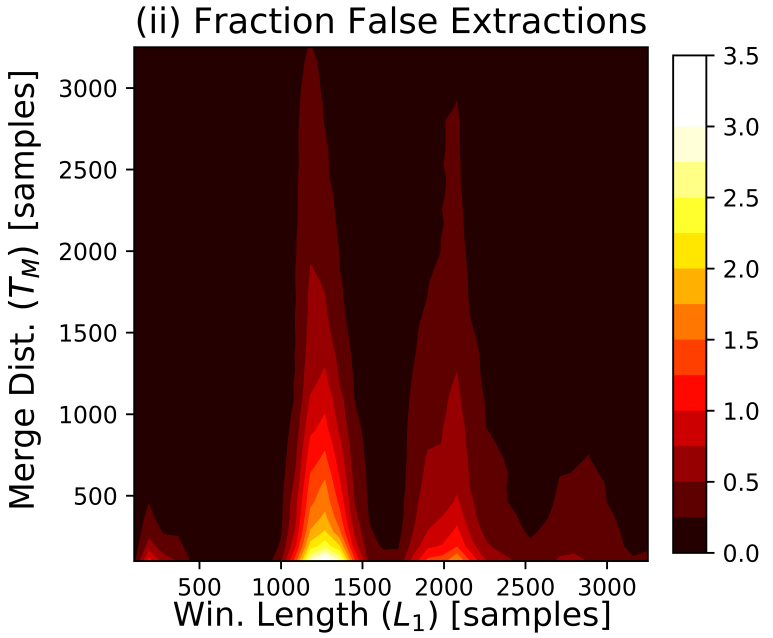

Figure 5.3: The number of erroneously extracted regions as a fraction of the number of ground truth transients for values of T_M and L_1.

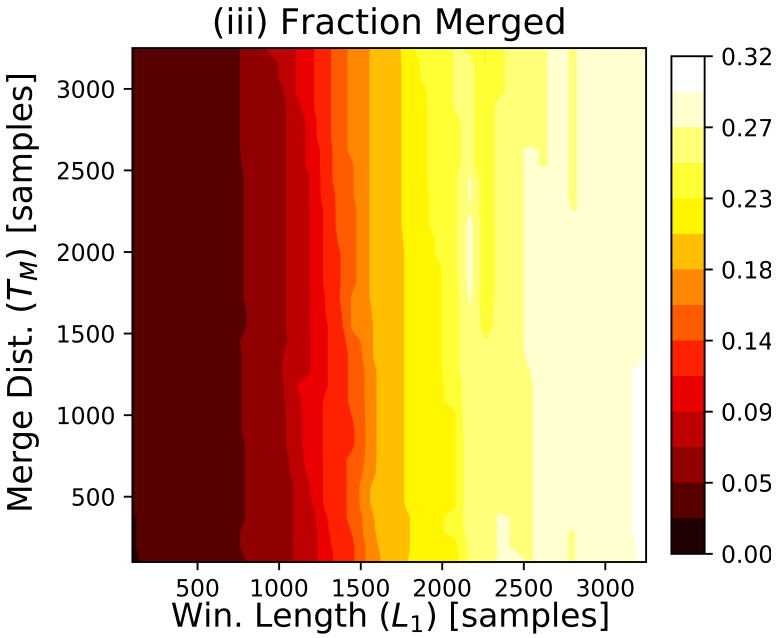

Figure 5.4: The fraction of ground truth transients that are erroneously merged for different values of T_M and L_1.

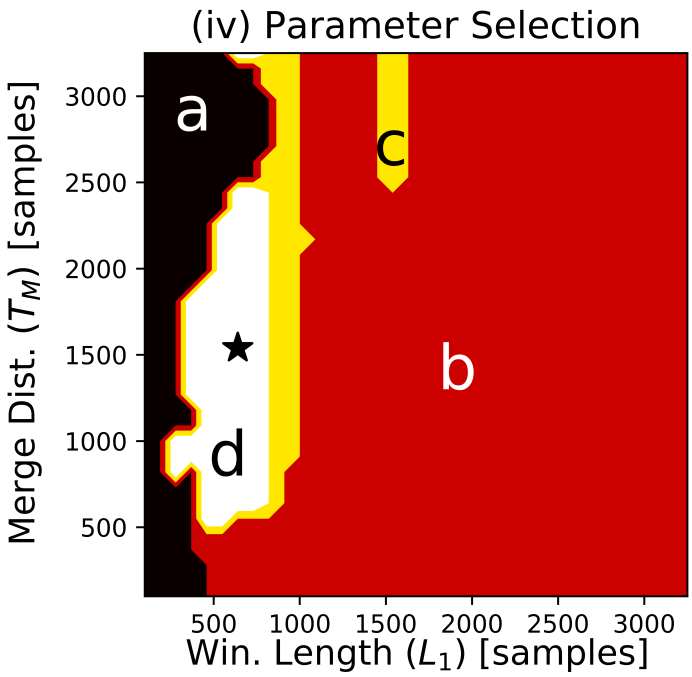

Figure 5.5: The selection of T_M and L_1, illustrated. Best results are obtained for $L_1 = 640$ and $T_M = 1540$. For these values, $93,4\%$ of the ground truth transients are correctly extracted, while 2.6% are incorrectly merged. The number of spurious extractions as a percentage of the number of ground truth transients is 2.2%.

The star indicates the chosen parameters: $L_1 = 640$ and $T_M = 1540$. At this location, the result of metric 1 (correctly extracted true transients) minus metric 2 (erroneously extracted regions) is highest (within region **d**). 93.4% of the ground truth transients are correctly extracted, while 2.6% are incorrectly merged together. The number of spurious extractions as a percentage of the number of ground truth transients is 2.2%.

5.2.2 Feature Selection

The result of the procedure in the preceding section is a set of transients and associated metadata: their sequence position, the number of their parent event from which they were extracted and the class to which they belong. The next step is to find discriminating features with which to label the transients. Given

the findings of Chapter 4, kernel PCA is used for this task. Since the transients are extracted from the full RFI events analysed in Chapter 4, it is likely that the same findings also hold true in this case. The procedure described here is applied only to training data. New, unseen transients are projected into the principal components space already computed from the training dataset.

Before the application of kernel PCA, several additional preprocessing steps are taken. Firstly, the transients are aligned by their maximum positive peaks and padded with zeros where necessary. Transients longer than 9000 time-steps from either start to peak or peak to end are excluded and assigned a special label for classification (Section 5.3.2). This step is taken to reduce computational cost without much loss (since very few transients need discarding).

Kernel PCA is applied as in Chapter 4, using a radial basis function (see Table 4.1). The parameter γ is set as $\frac{1}{no.\ features}$. Another way to select γ would be to consider it a parameter to be tuned during cross validation; however this would carry a large computational cost. The top components that explain 65% of the variance are retained, with the exception of the first three. Typically, components that together explain 80% of the variance would be kept (according to the rule of thumb); however using a lower percentage improves results here. Choosing 65% over 80% could be considered a rudimentary form of feature selection (applied to the choice of which components to retain).

Before continuing with the labelling step, a further scaling is applied to the components. It has been suggested previously that the higher-order components are the result of noise and do not represent discriminating features. In work by Schlkopf et al. [101], denoising steps entail discarding such higher-order components. Therefore, the scaled components $\boldsymbol{X'}$ are calculated as follows:

$$\boldsymbol{X'} = \boldsymbol{X} \times \frac{\boldsymbol{\lambda} - \mathtt{min}(\boldsymbol{\lambda})}{\mathtt{max}(\boldsymbol{\lambda}) - \mathtt{min}(\boldsymbol{\lambda})} \tag{5.2}$$

Here, $\boldsymbol{X}$ is the matrix of components produced by kernel PCA. The components in $\boldsymbol{X}$ are sorted by their corresponding eigenvalues in descending order. This set of eigenvalues is given by $\boldsymbol{\lambda}$.

5.2.3 Transient Labelling

Following the application of kernel PCA, the transients are labelled to form a dictionary from which full RFI signals can be represented. The procedure outlined here applies only to training data; new, unseen transients are labelled differently (see Section 5.3.1).

Unsupervised clustering algorithms are used to label the transients as projected in the principal components space. The k-means algorithm was attempted first, but its shortcomings quickly became apparent. In the principal components domain, visibly separable clusters were often incorrectly merged or assigned to multiple classes. These incorrect assignments were due to the elongated shapes of some clusters or their presence within sparse distributions of points. Class assignment also varied depending on the initial conditions of the k-means algorithm.

As a result, it was decided to use a density-based clustering algorithm, such as DBSCAN [102]. DBSCAN forms clusters from dense collections of points and labels distant or sparsely distributed points between such dense regions as noise. It does not require the number of clusters to be specified (as is the case with k-means clustering). Furthermore, irregularly shaped clusters do not pose a problem for DBSCAN. A full discussion is available in the original work by Ester et al. [102].

One disadvantage of DBSCAN, however, is that its performance suffers when the densities of clusters differ significantly. A number of variations and improvements on the original algorithm have been proposed to address this limitation [103, 104, 105]. However, instead of applying one of these variations, a simpler approach is chosen. To identify clusters in the data, DBSCAN is applied twice, once with parameters that favour denser clusters, and again with parameters which favour sparser clusters. The points labelled as noise by the first application of DBSCAN are used as the input for the second application. Parameter selection for each implementation of DBSCAN is dealt with in Section 5.3.3.

The results of this approach are illustrated in Fig. 5.6, in which three of 92 possible principal components are visualised. Examples of the original transient signals from three different clusters are also provided.

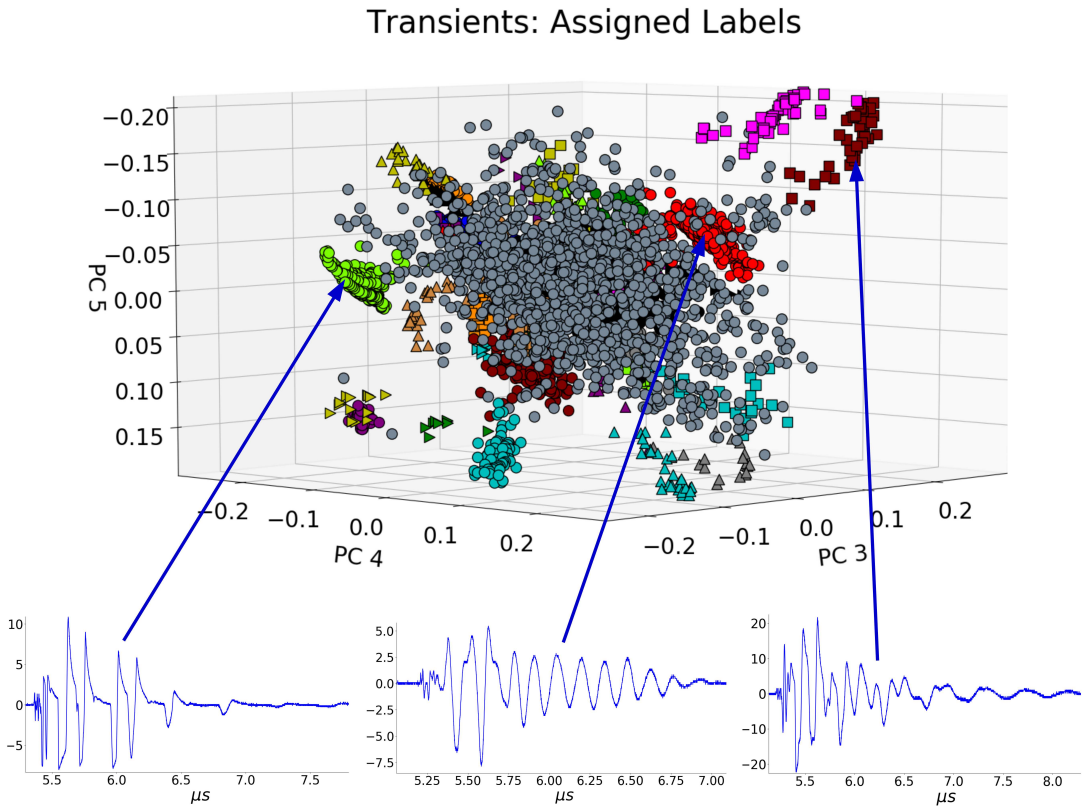

Figure 5.6: An example of the unsupervised clustering algorithm's output. Each marker represents a particular transient. The different colours indicate the labels assigned by the clustering algorithm. For legibility, transients that were labelled as noise are not included in this simplified plot. The clustering itself was carried out using 92 components, while only three of them are visualised here. Some clusters that appear to overlap are in fact well separated in higher dimensions. The original 3rd, 4th and 5th components are plotted. A few examples of the labelled transients are shown on the right. The transients' amplitudes are in arbitrary units, since they are displayed after pre-processing.

As indicated in Table 5.1, the largest clusters contain transients extracted from full recordings belonging to several different classes. This supports the notion of a common dictionary, the contents of which can be used to represent any of the RFI recordings.

Table 5.1: The distribution of three of the most important labels (out of 53) among the 8 classes. The transients belonging to individual labelled clusters are well distributed across the 8 classes, supporting the notion of a common dictionary.

Source	**% Label 1**	**% Label 2**	**% Label 3**
power tool	27.83	33.36	4.65
transformer	31.26	2.68	7.72
cable	21.66	57.49	7.77
relay (load)	27.48	7.62	0.25
relay	27.69	25.82	0.0
AC motor	27.91	42.55	1.08
CFL	33.11	13.33	12.80
PSU	27.79	25.81	11.79

5.3 Source Identification

The procedure for identifying the sources of new, unseen recordings of RFI signals is described in this section. HMMs are used, along with the dictionary of RFI transients.

5.3.1 Sequence Reconstruction and New Events

To identify RFI signals as planned, they must first be converted into sequences of the cluster labels determined in Section 5.2.3. Each full RFI signal is represented as a sequence of numbers, with each number corresponding to a particular cluster

(as established in Section 5.2.3). Fig. 5.7 illustrates an RFI signal represented as a sequence of labels.

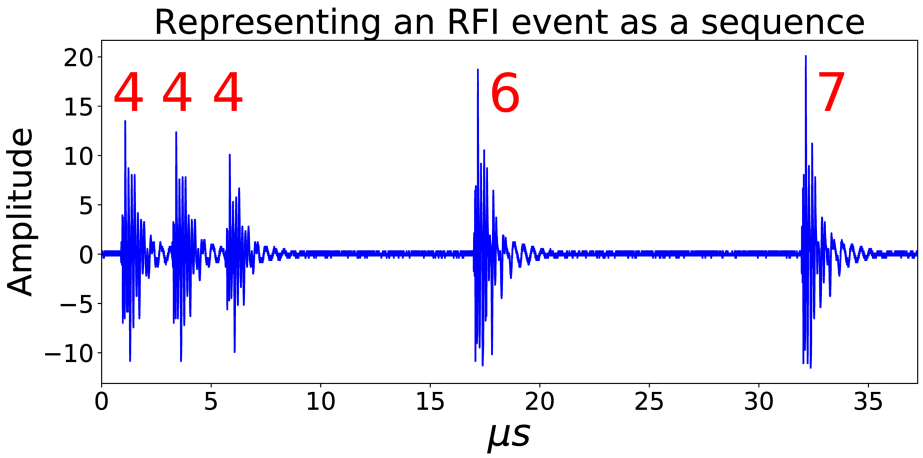

Figure 5.7: An example of an RFI event represented as a sequence of labels. The individual transients are first extracted as described in Section 5.2.1. They are then labelled in one of two ways (see Section 5.3.1) depending on whether or not the full RFI event is a training instance. This example is displayed after preprocessing, so units of amplitude are not included.

New, unseen RFI signals are dealt with separately. The individual transients of which the new signal consists are first extracted (using the algorithm in Section 5.2.1). Next, these transients are aligned and projected into the principal components space that was computed with the training data. The transients are then assigned the most common class of the k-nearest neighbouring training points. Finally, the new RFI signal is represented as a sequence of these assigned labels.

5.3.2 Source Identification Using Hidden Markov Models

Sources are identified using HMMs. As discussed in the introduction and Section 5.2, HMMs are well suited to classification tasks involving sequences of subunits. Furthermore, they are robust to variations in the outward expression of repeated examples of the same underlying sequence. Since many excellent references are freely available on the basic theory of HMMs, it is excluded from

this chapter. For a detailed tutorial on HMMs, please refer to the well-known publication by L Rabiner [71].

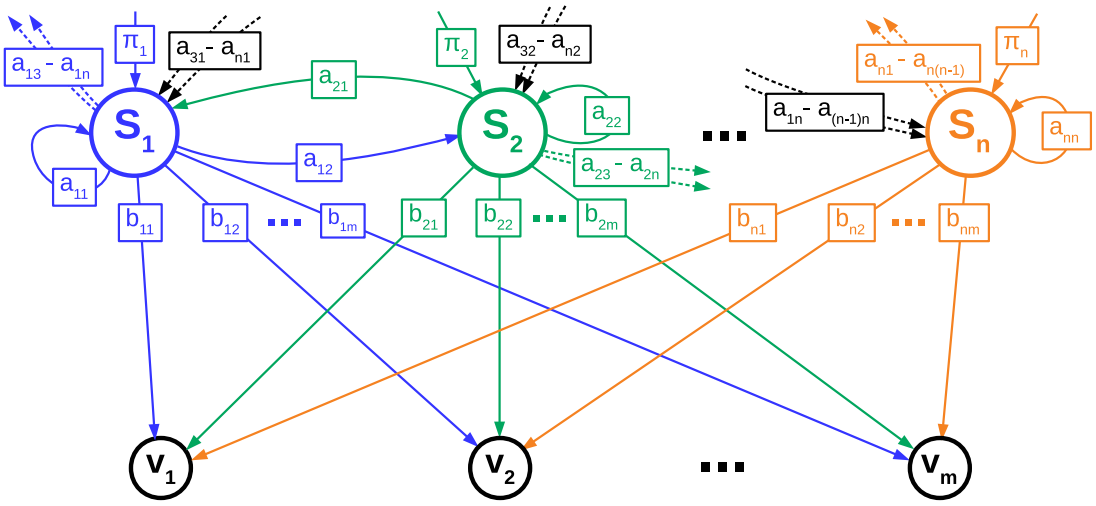

Figure 5.8: A diagram of one of the HMMs trained for each model. s_1 to s_n represent the n hidden states, while v_1 to v_m represent the m observable emissions. The elements of the transition matrix $\boldsymbol{A}$ are represented by each a_{ij}. $\boldsymbol{B}$, the emission probabilities, are given by each b_{ij}. The starting probabilities $\boldsymbol{\pi}$ are represented by $\boldsymbol{\pi}_1$ to $\boldsymbol{\pi}_n$.

The sequences of labels established from the RFI recordings in the training set are used to train an HMM for each class (see Fig. 5.8 for an example). The expectation-modification algorithm [71] is used for this purpose. For each new event, a figure of merit is calculated from the log probability obtained for each class model, using the Viterbi algorithm [71]. The new event is identified as belonging to the class for which the figure of merit is highest. To perform these computations, the Python package *hmmlearn* is employed [106]. Unrestricted HMMs are used, with Gaussian emission probabilities. When training the HMMs for each class, the number of hidden states must be specified beforehand. This value is treated as a hyperparameter, the selection procedure for which is described in the next section. Fig. 5.9 gives an example of a transition matrix ($\boldsymbol{A}$) computed for one of the classes. An example of an initial state distribution ($\boldsymbol{\pi}$) is illustrated in Fig. 5.10.

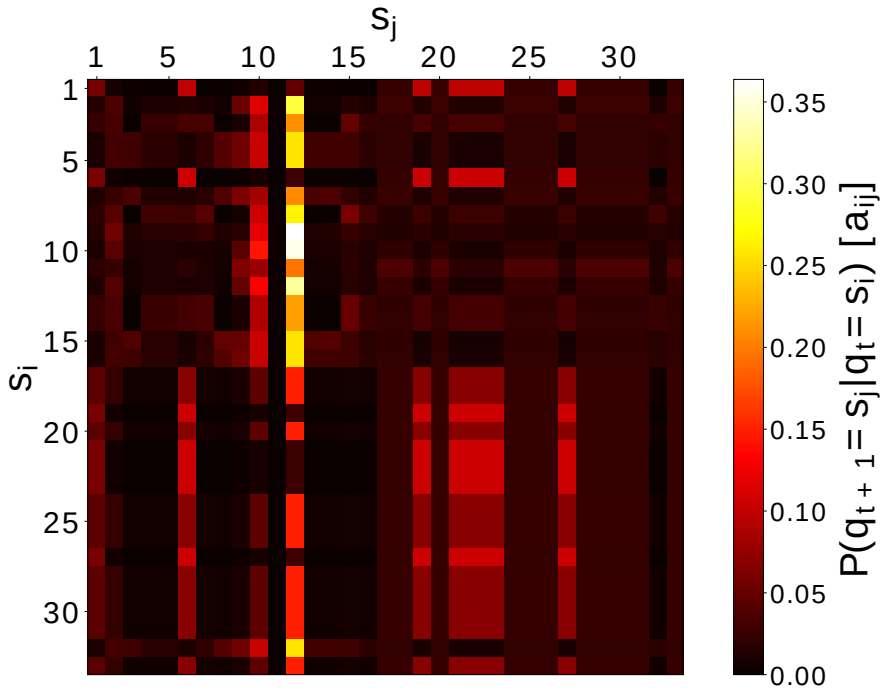

Figure 5.9: A visualisation of the transition matrix for the HMM trained for the CFL class. In an observed sequence of states $\boldsymbol{Q} = q_1, q_2, ..., a_{ij}$ is the probability that state q_{t+1} is s_j given that q_t is s_i. That is, $a_{ij} = P(q_{t+1} = s_j | q_t = s_i)$.

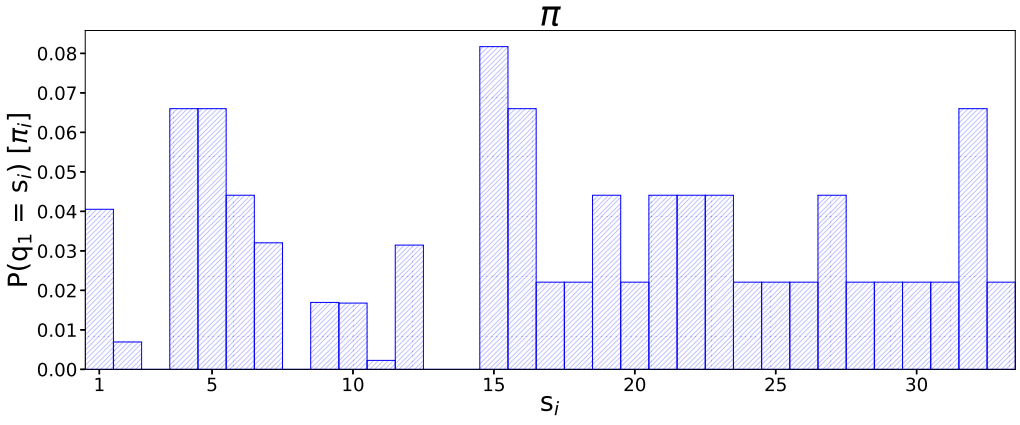

Figure 5.10: The computed probability distribution ($\boldsymbol{\pi}$) of the starting state for the HMM trained for the CFL class. $\boldsymbol{\pi} = P(q_1 = s_j)$.

5.3.3 Parameter Tuning and Classification Performance

The full dataset is divided into five randomised and stratified subsets, each containing a proportional number of instances from each class. Four of these subsets are used for parameter tuning via four-fold cross validation. Each training set consists of about 566 instances, and each validation set about 189 instances. The final fifth subset is kept as a separate testing set, unseen until final performance evaluation. These same subsets are also used in Section 4.5 in the preceding chapter, so those results are directly comparable with those in this section.

Six different parameters are tuned: The number of hidden states for the HMMs, the two sets of DBSCAN parameters (`eps` and `minpts`) and the number of nearest neighbours to consider when assigning new labels to unseen test signals. Tuning is accomplished via parametric searches, the results of which are provided in Table 5.2.

Table 5.2: The value of each parameter tuned via four-fold cross validation. The training accuracy is 65.54%, calculated as the mean of the four cross validation results. A test accuracy of 69.47% is obtained by training on the full cross validation dataset (80% of the data) and testing on the remaining untouched testing data.

Parameter Name	**Selected Value**
`minpts1`	40
`minpts2`	10
`eps1`	0.0745
`eps2`	0.0792
k	11
m	40

Classification accuracy on the testing set exceeds that of the validation set. A possible explanation is that significantly more training data are available for final evaluation. During four-fold cross validation for parameter tuning, 60% of the full original dataset are used for training. However, during final testing, all four of the subsets used in cross validation (comprising 80% of the data) can be used for training.

Table 5.3: A confusion matrix of the final results on the unseen testing data. Altogether, the testing data is classified correctly 69.47% of the time. If each class was correctly classified 100% of the time, then each element on the diagonal would be 100%.

	power tool	transformer	cable	relay (load)	relay	AC motor	CFL	PSU
power tool	**51.9**	11.1	7.4	0.0	22.2	3.7	0.0	3.7
transformer	0.0	**93.1**	0.0	0.0	3.4	0.0	3.4	0.0
cable	0.0	0.0	**80.0**	5.0	10.0	5.0	0.0	0.0
relay (load)	0.0	0.0	3.8	**69.2**	7.7	15.4	0.0	3.8
relay	0.0	3.6	25.0	0.0	**71.4**	0.0	0.0	0.0
AC motor	23.1	7.7	0.0	0.0	0.0	**38.5**	7.7	23.1
CFL	0.0	0.0	11.5	0.0	11.5	0.0	**73.1**	3.8
PSU	4.8	4.8	4.8	0.0	14.3	0.0	9.5	**61.9**

A confusion matrix is provided in Table 5.3, indicating how the instances from each class are classified and misclassified. The best classification result is obtained for the transformer, at 93.1%. Some possible reasons for this result might be that its RFI signals adhere to a more stable or unique sequence than other sources; or that the transients in its sequences are particularly distinct from others. Further investigation is needed to provide a definitive answer.

The worst-classified class is the AC motor, at only 38.5%. It is misclassified as the power tool 23.1% of the time, and as the switching power supply unit a further 23.1% of the time. Since the power tool also contains an AC motor of a similar type, some similarities in the associated RFI signals (and therefore some

misclassification) might be expected.

5.4 Conclusion

A new approach to identifying transient RFI that manifests as sequences of individual transients is proposed. Such individual transients are extracted from their parent signals by means of an automated algorithm, developed and tested with manually annotated signals. These transients are then labelled by applying kernel PCA and using unsupervised density-based clustering techniques such as DBSCAN. Full RFI signals are then represented as sequences of labels, and used to train HMMs, one for each class of RFI source. New, unseen RFI signals are split into their constituent transients, which are subsequently projected into the principal components space computed with the training signals and labelled accordingly. Each new RFI signal is then represented as a sequence, and scored for each trained HMM, thus predicting its class.

The results obtained using this method improve upon those attained via the more direct methods presented in Chapter 4. The dictionary-based approach achieved an overall accuracy of 69.47%, while the accuracies obtained in Chapter 4 ranged from 18.95% for the kNN classifier to 61.58% for the SVM classifier. These results are directly comparable, as the same training, validation and testing splits were used for training, parameter tuning and testing.

In recent years, recurrent neural networks have proven highly capable in fields such as automated speech recognition [107]. This is due in part to increases in available computing power and training data along with the development of new techniques such as Long Short-Term Memory (LSTM) units [108] and Gated Recurrent Units (GRUs) [109]. The next chapter examines the potential of such deep learning techniques for classifying transient RFI by source.

Chapter 6

Convolutional Neural Networks and Long Short-Term Memory for Transient RFI Classification[1]

6.1 Introduction

In recent years, deep learning techniques have risen to prominence across many domains, from automated speech recognition to image processing [12]. In many cases, for example in computer vision and ASR, deep neural networks including convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have supplanted or enhanced older techniques [110, 12, 111]. Given that some of these techniques have proven widely successful at classification tasks, they may be an appropriate approach to identifying transient RFI.

For a full discussion of prior literature please see Chapter 2. In summary, there have been few prior attempts (and none in a radio astronomy context) to classify transient RFI by source using deep learning techniques. Basic neural networks have been used to classify vehicles by make based on their RF emissions [67]. In other work, CNNs have been used to identify sources of interference in WiFi signals [112], although most of the sources considered were

[1]This chapter is based in part upon the following publication:
Czech, D, A Mishra, and M Inggs. "A CNN and LSTM-Based Approach to Classifying Transient Radio Frequency Interference." Astronomy and Computing 25 (2018): 52-57.

continuous transmitters. CNNs have also been used to flag RFI in astronomical time-frequency plots, but not determine its specific source [39].

In this chapter, CNNs and bidirectional LSTMs are used for the first time to classify individual time-domain RFI transients by source.

6.2 Data and Preprocessing

Individual transients are extracted from the full RFI events in Dataset 2 (see Chapter 3), which each consist of sequences of transients. The procedure described in Algorithm 1 (Chapter 5) is used to extract the individual transients from each event. For the analysis in this chapter, a minor modification is made to Algorithm 1: raw transient signals with peak magnitudes ≤ 5 least significant bits are discarded. As in Chapter 5, the two CFL classes are joined. The number of extracted transients per class is given in Table 6.1.

Table 6.1: RFI sources and the number of transients belonging to each.

Class	Description	No. Transients
1	Compact fluorescent lamp	662
2	Power tool	543
3	Step-down transformer	5 523
4	Cable	264
5	Mechanical relay (700W resistive load)	16 006
6	Mechanical relay (without load)	35 932
7	AC motor (approximately 1 kW)	3 675
8	Small switching power supply	525

6.2.1 Preprocessing

To avoid unnecessary computational overhead, transients are limited in length to 5000 raw time samples, since the majority of extracted transients are shorter. In addition, transients are aligned by their maximum positive peak and padded with zeros where necessary.

A potential classifier should not be influenced by the relative amplitudes of different transient signals. For example, identical sources at different distances from an RFI monitoring station (and thus producing signals differing in amplitude) should be classified identically. To ensure this, each transient is scaled so that its amplitude ranges from -1 to 1. A scaled transient $\boldsymbol{u'}$ is calculated from its unscaled counterpart $\boldsymbol{u}$ as follows:

$$\boldsymbol{u'} = 2\frac{\boldsymbol{u} - \min(\boldsymbol{u})}{\max(\boldsymbol{u}) - \min(\boldsymbol{u})} - 1 \tag{6.1}$$

Note that this scaling was not applied to individual transients in Chapter 5, since entire RFI events (consisting of sequences of transients) were used for classification. The relative amplitudes of individual transients in the same sequence are useful discriminating features when an entire RFI event is used for classification. In that case it is instead important to ensure equivalence in event amplitude across classes.

Feature scaling is also applied to ensure that all features have equivalent influence. Each feature vector $\boldsymbol{x}_{\boldsymbol{u}j}$ contains the individual values at the same time-step, j, of every transient signal $\boldsymbol{u'}$ in a training set. Each scaled feature vector $\boldsymbol{x}_{\boldsymbol{u}'j}$ is calculated as follows:

$$\boldsymbol{x}_{\boldsymbol{u}'j} = \frac{\boldsymbol{x}_{\boldsymbol{u}j} - \mu_j}{\sigma_j} \tag{6.2}$$

The mean of the feature vector $\boldsymbol{x}_{\boldsymbol{u}'j}$ is given by μ_j and σ_j represents its standard deviation. The standardisation parameters are determined from the training data alone. These predetermined parameters are used when applying feature standardisation to testing data.

6.2.2 Division of Data

The full dataset is divided into three subsets for training, hyperparameter tuning and final testing. The initial training set consists of 60% of the data, while the validation and testing sets account for 20% each. Hyperparameters are tuned by training on the initial training set and evaluating on the validation set. For final evaluation, the chosen model is trained on the validation and initial training sets combined, and tested on the unseen testing set.

6.3 Model Architecture

Two approaches are considered and compared. The first uses a 1-D CNN and a BLSTM layer, while the second replaces the BLSTM layer with a linear SVM classifier. The CNN layer reduces the length of the input sequence for each transient and helps to determine discriminating features. A BLSTM was chosen since BLSTMs have been shown to be superior to others in tasks such as ASR [107, 111]. For example, on the TIMIT corpus, BLSTMs have provided a slight improvement over their unidirectional counterparts [111]. In the same publication, a combination of frequency domain CNNs and deep LSTMs is proposed for future work. Architectures involving 1D-CNN and BLSTM layers have been applied to time domain data before in end-to-end speech emotion recognition [113], improving on prior approaches using designed features.

Both approaches mentioned above use a pre-trained CNN layer (see Fig. 6.1). An initial model is formed by replacing all subsequent layers with a temporary fully-connected output layer (delivering classification predictions in 1-hot form). This initial model is trained, following which the CNN's weights are frozen and the temporary output layer is removed. In the case of the BLSTM approach, it is replaced with a BLSTM layer, followed again by a fully-connected output layer. In the case of the SVM approach, it is replaced with a set of linear SVMs. Some examples of the filters learned by the CNN are given in Fig. 6.2.

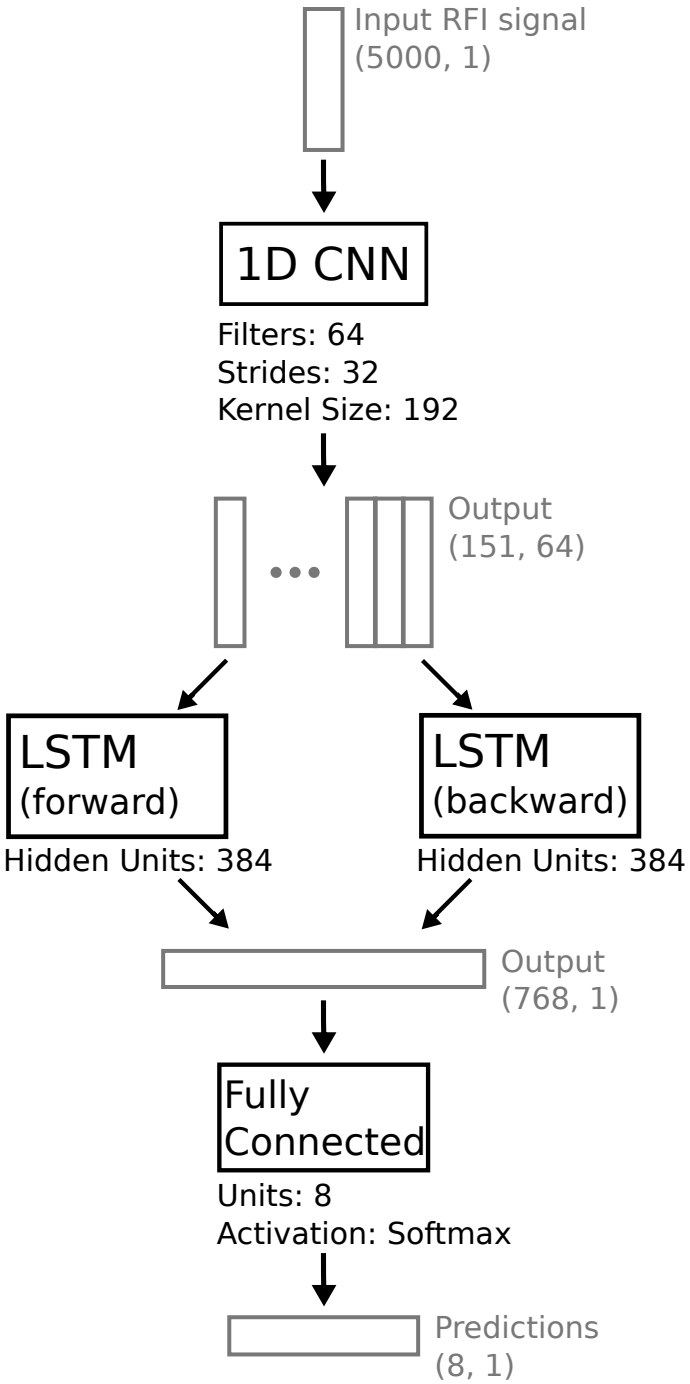

Figure 6.1: The structure of the selected model. The parameters shown are those that are used for the reduced dataset. For the full dataset (for which class imbalance is managed via class weighting), the kernel size is reduced to 160 time steps and the batch size increased from 128 to 256. The outputs of the forward and backward LSTMs are concatenated in the bidirectional LSTM layer.

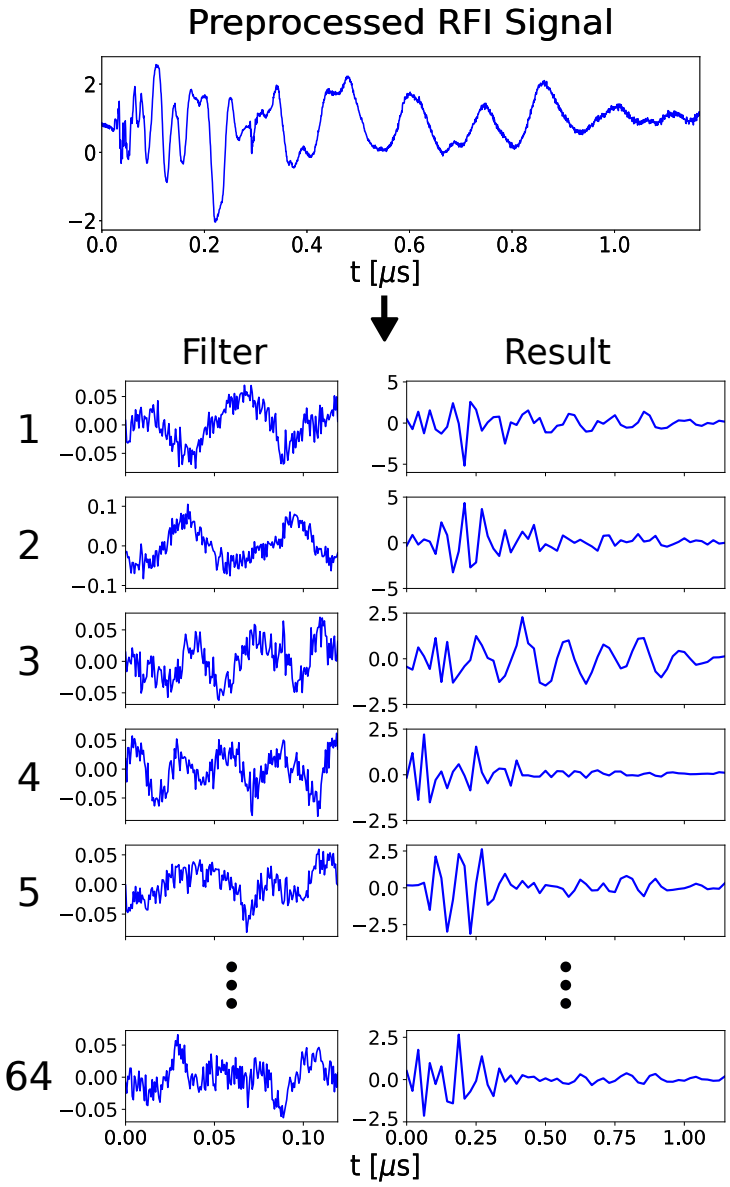

Figure 6.2: Some examples of the CNN's different filters are displayed here. The results of applying each of these filters to one of the preprocessed transient RFI signals are also illustrated.

6.3.1 Frameworks and Computation

Keras [13] (with Tensorflow [14]) is used build and train models. Hyperparameters are tuned on the training and validation datasets (the testing dataset is set aside for final evaluation). The Python library Hyperopt [15] is used to help automate this process. An Amazon p2.xlarge instance (2.7 GHz Broadwell CPU; 61 GiB RAM; 12 GiB NVIDIA Kepler K80 GPU; Ubuntu 16.04) is used to perform computations. SciKit-Learn's implementation of a linear SVM [114] is used in the second approach.

6.3.2 CNN-Bidirectional LSTM Approach

Fig. 6.1 illustrates the architecture of the CNN-BLSTM approach.

The RFI events from certain classes contain on average many more transients than others. Many more individual transients are extracted for some classes than others, as is evident in Table 6.1. To avoid favouring the more prevalent classes, this imbalance must be taken into account during training. When using backpropagation to train neural networks, one method of taking class imbalance into account is to weight instances by class in the loss function as follows:

If $\boldsymbol{G}$ is the vector containing the number of instances G_i in each class i then a vector of class weights $\boldsymbol{c}$ may be calculated as follows:

$$\boldsymbol{c} = \langle \frac{\max(\boldsymbol{G})}{G_1}, \frac{\max(\boldsymbol{G})}{G_2}, \dots \frac{\max(\boldsymbol{G})}{G_i} \rangle \tag{6.3}$$

These weights may be applied in the loss function $\mathcal{L}$ during training. Using the categorical cross-entropy loss function, with M classes and a batch-size of N:

$$\mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{N} \sum_{i=1}^{M} \sum_{j=1}^{N} y_{ij} \log(\hat{y}_{ij}) c_i \tag{6.4}$$

$\boldsymbol{y}$ is the 1-hot encoded ground truth. $\hat{\boldsymbol{y}}$ is the output of the final layer after the application of an activation function. The softmax activation function is used in the final fully-connected layer to obtain $\hat{\boldsymbol{y}}$. The activation output $\hat{y}_i$ at each of

the i final nodes (one per class) is:

$$\hat{y}_i = \frac{e^{o_i}}{\sum_{j=0}^{M} e^{o_j}} \tag{6.5}$$

where o is the weighted sum of activations in the hidden layers. Given the choice of activation function, the partial derivative of the loss function with respect to the output of the final layer can be derived as:

$$\frac{\partial \mathcal{L}}{\partial o_i} = c_i(\hat{y}_i - y_i) \tag{6.6}$$

where o_i is the activation output. The gradients of the rarer classes are steepened relative to those of the more common classes due to the class weighting. As a result, misclassifying an instance belonging to a rare class incurs a proportionately greater penalty than misclassifying an instance belonging to a common class.

6.3.3 CNN-SVM Approach

In the CNN-SVM approach, the BLSTM and final fully-connected layers are replaced with a set of linear SVMs in a one-vs-all configuration. Before being fed to the SVMs, the output of the CNN is flattened such that the outputs of the filters are concatenated together. The output of the CNN for a single instance is thus of the shape $(\texttt{timesteps} \times \texttt{num\_filters}, 1)$ rather than $(\texttt{timesteps}, \texttt{num\_filters})$.

To account for class imbalance, class weights are calculated as before in (6.3). Each class weight is used to scale the corresponding SVM's regularisation parameter, C_r. The penalty for misclassifying an instance from a rare class is thus proportionally greater than misclassifying an instance from a common class.

6.4 Results

Results are presented for both the CNN-BLSTM and CNN-SVM approaches. In addition, results are reported using an alternative strategy for handling class imbalance. In this strategy, the number of instances in each class is made equal

by discarding (at random) excess instances in the larger classes. The number of instances available for training and testing is reduced, and therefore classification performance suffers. This basic method of dealing with class imbalance may be compared with the other class-weighting methods employed in Sections 6.3.2 and 6.3.3.

Accuracy, precision, recall and F_1 score metrics are provided in Table 6.2 and a full confusion matrix in Table 6.3. Precision, recall and the F1 score are calculated individually for each class and the average reported. For M classes,

$$\text{precision} = \frac{1}{M} \sum_{i=1}^{M} \frac{\texttt{tp}_i}{\texttt{tp}_i + \texttt{fp}_i} \tag{6.7}$$

$$\text{recall} = \frac{1}{M} \sum_{i=1}^{M} \frac{\texttt{tp}_i}{\texttt{tp}_i + \texttt{fn}_i} \tag{6.8}$$

where $\texttt{tp}_i$ = true positives, $\texttt{fp}_i$ = false positives and $\texttt{fn}_i$ = false negatives for class i. The combined F_1 score is calculated as follows:

$$F_1 = 2 \, \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{6.9}$$

The CNN-BLSTM approach performs better than the CNN-SVM approach in every respect. This is likely due (in part) to the namesake ability of the BLSTMs to model sequential structure in the input data. There is an inherent sequential structure in the transients (clearly apparent in Fig. 6.2 for example). In contrast to BLSTMs, SVMs do not explicitly model sequential relationships.

Furthermore, the CNN-BLSTM approach handles class imbalance significantly better. For example, it correctly classifies the smallest class 96.15% of the time, while the CNN-SVM approach only manages correct classifications 37.14% of the time. The most poorly-classified class by both approaches is the switching power supply unit, with an accuracy of 77.14% for the CNN-BLSTM approach, while only 37.14% for the CNN-SVM approach.

Table 6.2: Evaluation of results.

Metric	Reduced Dataset		Full Dataset	
	CNN + SVM	CNN + BLSTM	CNN + SVM	CNN + BLSTM
Accuracy	0.7764	0.8197	0.9403	0.9636
Precision	0.7794	0.8239	0.7843	0.8467
Recall	0.7764	0.8197	0.7453	0.9138
F_1	0.7754	0.8195	0.7606	0.8764

Table 6.3: The confusion matrix for the approach using a CNN and bidirectional LSTM on the full (imbalanced) testing set.

	CFL	power tool	transformer	cable	relay (load)	relay	AC motor	PSU	% Correct
CFL	**120**	1	7	0	2	0	0	2	90.91
power tool	1	**88**	6	2	2	3	0	6	81.48
transformer	9	4	**1035**	0	1	19	5	31	93.75
cable	0	1	1	**50**	0	0	0	0	96.15
relay (load)	43	4	9	2	**3117**	13	13	0	97.38
relay	2	9	166	0	18	**6957**	22	12	96.81
AC motor	1	0	11	0	4	3	**716**	0	97.41
PSU	2	3	16	0	0	3	0	**81**	77.14

6.5 Conclusion

This chapter illustrates how deep learning techniques like CNNs and LSTMs can be used to identify the sources of transient RFI. The use of an initial CNN layer appears to be a powerful approach in this scenario. On the face of it, the

Table 6.4: The confusion matrix for the approach using a CNN and linear SVM classifier on the full (imbalanced) testing set.

	CFL	power tool	transformer	cable	relay (load)	relay	AC motor	PSU	% Correct
CFL	**81**	2	11	1	28	5	2	2	61.36
power tool	4	**59**	16	0	13	10	2	4	54.63
transformer	6	5	**931**	4	47	78	21	12	84.33
cable	0	0	3	**39**	3	4	2	1	75.00
relay (load)	16	6	38	5	**3068**	35	26	7	95.85
relay	4	5	80	5	67	**6986**	30	9	97.22
AC motor	1	1	24	2	23	15	**667**	2	90.75
PSU	0	1	46	2	7	9	1	**39**	37.14

classification results (Table 6.2) are better than those achieved via the dictionary approach (Table 5.2) or other initial approaches (Section 4.5). In part, the improved results might be attributable to the relative increase in training examples. Since the other approaches require full sequences of transients, 944 instances are available for training and testing. By comparison, in this approach 63 130 instances are available since each individual transient is extracted and considered an instance.

Chapter 7

Conclusion

As explained in prior chapters, transient RFI threatens the quality of astronomical observations made by radio telescopes. In some cases, transient RFI appears very similar to legitimate astronomical sources, potentially resulting in wasted resources or false findings. New approaches are needed for dealing with transient RFI. Research into such approaches is all the more pertinent as transient radio astronomy has grown significantly since the discovery of phenomena like Fast Radio Bursts.

One such approach is to identify the sources of human-generated transient RFI in recordings from monitoring stations independent of the main telescope array. Once identified, the sources of transient RFI can be removed or replaced where possible.

The goal of this thesis is to test the hypothesis that sources of human-generated transient RFI can be identified by their RF emissions alone. To accomplish this, several research questions are posed. In answering them, a number of novel contributions are realised. A chapter is dedicated to answering each of the research questions.

7.1 Key Contributions

In Chapter 3, human-generated transient RFI is characterised. This involved recording various labelled datasets of transient RFI and attempting to model

them statistically. The distribution across recordings from each source is shown to be in essence non-Gaussian. A variety of parametric models are applied to the data but all fail goodness of fit tests. As a result, kernel density estimation techniques are proposed instead.

Chapter 4 investigates whether or not components analysis techniques are suitable for feature reduction prior to classifying transient RFI. Standard linear PCA is shown to be less appropriate than nonlinear techniques such as kernel PCA, perhaps not surprising given the findings in Chapter 3. A naïve approach to classification is attempted, using kernel PCA along with SVM and kNN classifiers. In keeping with a phenomenological approach, the relationship between the phase of the supply voltage and cluster separation in the principal components domain is examined. For some RFI-emitting devices powered by an AC mains supply, cluster separation is improved when $|V_{supply}|$ is near V_{peak}.

A dictionary-based approach to identification is proposed in Chapter 5. A method for automatically extracting individual transients is proposed and evaluated, along with a process for building a canonical dictionary. It is described how full RFI events (which consist of sequences of transients) can be represented as sequences of labels drawn from such a dictionary. Hidden Markov models are then applied successfully to identify events. The entire process is evaluated using a rigorous approach involving cross validation and an unseen testing set.

Finally, in Chapter 6, the suitability of deep learning techniques for transient RFI classification is tested. An approach using CNNs and bidirectional LSTMs is developed and evaluated using Keras and Tensorflow. The model is trained on a dataset of individual transients extracted from full RFI events. Special steps are taken to mitigate significant class imbalance. Despite the use of individual transients only, good classification results are achieved.

Encouragingly, each approach to classifying transient RFI has improved on the last. The best results are achieved using deep learning techniques, which improve significantly on the dictionary approach. It would seem on the surface that deep learning approaches have trounced the other competing techniques, as they have in many other fields. However, deep learning approaches do in general suffer from certain disadvantages (for example computational expense and the need for large training datasets). It remains to be seen how robust the models

developed in Chapter 6 will be in the face of new, unknown sources. This is dependent on the stability of the structure of individual transients. For example, contacts may gradually wear away and environmental changes could affect the dielectric between them. It may yet turn out that sequences of transients are more stable than the individual transients themselves. Further investigation is needed to answer these questions.

7.2 Recommendations for Future Work

One important task for future work is to determine the extensibility of the proposed algorithms. While encouraging, the results quoted in this thesis are obtained for a few specific datasets. Testing conditions are kept constant within each separate labelled dataset - the same recording equipment was used and devices were switched using the same switching arrangement. It remains to be seen how much training data is needed before these algorithms can be properly applied in a practical setting.

A much larger dataset consisting of many different sources switched in a variety of different ways would help settle this question. In addition, this would reveal which of the approaches in this thesis is most robust to unseen sources. In the ideal case, all similar sources would produce similar transient signals. Thus, if there is only one example of a specific type of source in the training set, a new, unseen example of this source would likely be classified correctly. This has shown to be true in at least one particular case in this thesis (two different CFLs; see Chapter 4).

It could still be that there is overlap between different sources switched under different circumstances. If it turns out to be difficult to identify specific sources, then perhaps a more sensible approach would be to classify coarser classes of source. One way to achieve this could be to classify sources by their physical characteristics which give rise to the emissions they generate. If this is the case, it may be possible to detect that a specific device contains a large inductor, a non-ohmic resistive element or a rotor with brushed contacts, for example. With this knowledge, it might be possible to infer the identity of a new, unseen source not part of the training data.

One possible experiment would be to construct a number of archetypal devices and vary their characteristics while recording their RF emissions. For example, a relay with a variable coil size and contact area. It would be interesting to see if it is possible to link the chosen attributes to components in the principal components domain, similar to the way the influence of the supply voltage phase was investigated in Chapter 3. This would help to provide a human-comprehensible explanation for classification decisions.

Of the different classification approaches developed in this thesis, those employing deep learning techniques offer the best performance (see Chapter 6). This is the case when trained and tested using the full dataset, however. It may be that some of the other methods (such as the dictionary-based approach in Chapter 5) would fare better when reduced training data is available, for example. Therefore, another avenue for future analysis would be to compare the accuracy of each approach while varying the quantity of available training data.

Another investigation might be to evaluate the different classification approaches on degraded data. That is, a dataset of known transient RFI signals could be recorded in the presence of another source of RFI, such as an intentional CW transmitter of some kind. This would simulate a potentially rare scenario (at least for radio telescope sites) in which transient RFI is continuously contaminated by a relatively powerful CW source. It would be intriguing to observe how well the different classification approaches cope with transient RFI signals that have been contaminated by (comparatively) more narrowband CW signals. Transient RFI signals might also be made to overlap to varying degrees; however, overlapping transient RFI is much less likely to occur since transient RFI signals are so short and intermittent.

Interesting replacements could be found for the dictionary construction step in Chapter 5. Instead of kernel PCA and density-based clustering, perhaps wavelets could provide the archetypal transients with which to assign labels and generate sequences. Or, CNNs could conceivably be used to generate a set of filters that would stand in for the archetypal transients.

Another task would be to create a fully automated software package to run on the latest incarnation of the Real Time Analyser to be deployed at the MeerKAT site. This would entail adapting these different algorithms for use in real-time

(not such a difficult task). The principal difficulty would likely be the storage and retrieval of labelled training data as well as new recorded signals.

It may be fruitful to determine if transient RFI recorded by monitoring stations could be linked to transients in the power supply network. If a link could be found, monitoring the power supply network may aid in identifying sources of transient RFI. Conversely, such a link would also be useful for non-intrusive load monitoring. The ability to monitor load usage at a distance would be particularly powerful.

Another important area for future work would be to integrate the transient RFI identification algorithms with a transient RFI location system. The inclusion of location information may also enable more accurate source identification.

There are a number of modifications and enhancements to the dictionary approach in Chapter 5 that could be investigated. Different HMM architectures could be explored, for example.

It may be possible to adapt the approaches investigated in this thesis to the classification of astronomical and atmospheric radio transients. This would be an exciting avenue for future work, given the rise in prominence of transient radio astronomy and the discovery of mysterious phenomena such as FRBs. Such research may also be applicable to radio searches for extra-terrestrial intelligence.

Finally, some of these approaches may be adapted for RFI excision in data directly from radio telescopes. The ability to identify particular RFI sources may translate into more accurate RFI excision. Furthermore, these approaches may be better able to avoid excising genuine astronomical transients that nevertheless share some characteristics with RFI signals. Rudimentary excision techniques may be less capable of discriminating between such transients, thus discarding signals of interest. Given the vast data rates from modern radio telescopes, computation time would likely be more of an obstacle than is the case for remote monitoring station applications.

Bibliography

[1] M. Bailes, E. Barr, N. Bhat, J. Brink, S. Buchner, M. Burgay, F. Camilo, D. Champion, J. Hessels, G. Janssen, et al., MeerTime - the MeerKAT key science program on pulsar timing, arXiv preprint (2018) arXiv:1803.07424.

[2] L. R. Brederode, L. van den Heever, W. Esterhuyse, J. L. Jonas, MeerKAT: a project status report, in: Ground-based and Airborne Telescopes VI, Vol. 9906, International Society for Optics and Photonics, 2016, p. 990625.

[3] S. Burke-Spolaor, M. Bailes, R. Ekers, J.-P. Macquart, F. Crawford III, Radio bursts with extragalactic spectral characteristics show terrestrial origins, The Astrophysical Journal 727 (1) (2010) 18.

[4] E. Petroff, E. Keane, E. Barr, J. Reynolds, J. Sarkissian, P. Edwards, J. Stevens, C. Brem, A. Jameson, S. Burke-Spolaor, et al., Identifying the source of perytons at the Parkes radio telescope, Monthly Notices of the Royal Astronomical Society 451 (4) (2015) 3933–3940.

[5] J. Manley, F. Kapp, The MeerKAT array and its digital signal processor, in: 2012 International Conference on Electromagnetics in Advanced Applications (ICEAA), IEEE, 2012, pp. 462–465.

[6] Astronomy Geographic Advantage Act, No 21 of 2007, Government Gazette, Vol. 516, No. 31157, Cape Town, Republic of South Africa, 2008.

[7] D. R. DeBoer, A. R. Parsons, J. E. Aguirre, P. Alexander, Z. S. Ali, A. P. Beardsley, G. Bernardi, J. D. Bowman, R. F. Bradley, C. L. Carilli, et al., Hydrogen epoch of reionization array (HERA), Publications of the Astronomical Society of the Pacific 129 (974) (2017) 045001.

[8] R. F. Bradley, The precision array for probing the epoch of reionization (PAPER): A modern scientific adventure, URSI Radio Science Bulletin 2017 (362) (2017) 39–52.

[9] T. J. Pearson, et al., C-BASS: The C-band all sky survey, in: American Astronomical Society Meeting Abstracts, Vol. 228, 2016.

[10] L. Newburgh, K. Bandura, M. Bucher, T.-C. Chang, H. Chiang, J. Cliche, R. Davé, M. Dobbs, C. Clarkson, K. Ganga, et al., HIRAX: a probe of dark energy and radio transients, in: Ground-based and Airborne Telescopes VI, Vol. 9906, International Society for Optics and Photonics, 2016, p. 99065X.

[11] A. Foley, T. Alberts, R. Armstrong, A. Barta, E. Bauermeister, H. Bester, S. Blose, R. Booth, D. Botha, S. Buchner, et al., Engineering and science highlights of the KAT-7 radio telescope, Monthly Notices of the Royal Astronomical Society 460 (2) (2016) 1664–1679.

[12] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436.

[13] F. Chollet, et al., Keras, https://github.com/keras-team/keras (2015).

[14] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, et al., TensorFlow: Large-scale machine learning on heterogeneous systems, https://www.tensorflow.org/, software available from tensorflow.org (2015).

[15] J. Bergstra, D. Yamins, D. Cox, Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures, in: International Conference on Machine Learning, 2013, pp. 115–123.

[16] I. Dodin, N. Fisch, Are perytons signatures of ball lightning?, The Astrophysical Journal 794 (2) (2014) 98.

[17] Y. Cendes, P. Prasad, A. Rowlinson, R. Wijers, J. Swinbank, C. Law, A. van der Horst, D. Carbone, J. Broderick, T. Staley, et al., RFI flagging implications for short-duration transients, Astronomy and Computing 23 (2018) 103–114.

[18] M. A. McLaughlin, J. Cordes, Searches for giant pulses from extragalactic pulsars, The Astrophysical Journal 596 (2) (2003) 982.

[19] G. Doran, Characterizing interference in radio astronomy observations through active and unsupervised learning, Tech. rep., Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration. (2013).

[20] E. Anterrieu, On the detection and quantification of RFI in L1a signals provided by SMOS, IEEE Transactions on Geoscience and Remote Sensing 49 (10) (2011) 3986–3992.

[21] M. Andrews, H. Li, J. T. Johnson, A. Bringer, The ultra-wideband software defined microwave radiometer (UWBRAD) for ice sheet subsurface temperature sensing: RFI algorithms and performance, in: 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE, 2017, pp. 1263–1265.

[22] G. Tsihrintzis, C. L. Nikias, et al., Fast estimation of the parameters of alpha-stable impulsive interference, IEEE Transactions on Signal Processing 44 (6) (1996) 1492–1503.

[23] R. F. Dwyer, Detection of non-Gaussian signals by frequency domain kurtosis estimation, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'83., Vol. 8, IEEE, 1983, pp. 607–610.

[24] L. T. DeCarlo, On the meaning and use of kurtosis., Psychological methods 2 (3) (1997) 292.

[25] S. S. Sobjaerg, J. Svoboda, J. E. Balling, N. Skou, Detection of radio-frequency interference in microwave radiometers using spectral kurtosis, in: 2012 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE, 2012, pp. 7141–7144.

[26] S. Misra, P. N. Mohammed, B. Güner, C. S. Ruf, J. R. Piepmeier, J. T. Johnson, Microwave radiometer radio-frequency interference detection algorithms: A comparative study, IEEE Transactions on Geoscience and Remote Sensing 47 (11) (2009) 3742–3754.

[27] C. S. Ruf, S. M. Gross, S. Misra, RFI detection and mitigation for microwave radiometry with an agile digital detector, IEEE Transactions on Geoscience and Remote Sensing 44 (3) (2006) 694–706.

[28] A. Offringa, A. de Bruyn, M. Biehl, S. Zaroubi, G. Bernardi, V. Pandey, Post-correlation radio frequency interference classification methods, Monthly Notices of the Royal Astronomical Society 405 (1) (2010) 155–167.

[29] P. Fridman, A method of detecting radio transients, Monthly Notices of the Royal Astronomical Society 409 (2) (2010) 808–820.

[30] R. P. Eatough, E. F. Keane, A. G. Lyne, An interference removal technique for radio pulsar searches, Monthly Notices of the Royal Astronomical Society 395 (1) (2009) 410–415.

[31] R. P. Eatough, N. Molkenthin, M. Kramer, A. Noutsos, M. Keith, B. Stappers, A. Lyne, Selection of radio pulsar candidates using artificial neural networks, Monthly Notices of the Royal Astronomical Society 407 (4) (2010) 2443–2450.

[32] V. Morello, E. Barr, M. Bailes, C. Flynn, E. Keane, W. van Straten, SPINN: a straightforward machine learning solution to the pulsar candidate selection problem, Monthly Notices of the Royal Astronomical Society 443 (2) (2014) 1651–1662.

[33] T. R. Devine, K. Goseva-Popstojanova, M. McLaughlin, Detection of dispersed radio pulses: a machine learning approach to candidate identification and classification, Monthly Notices of the Royal Astronomical Society 459 (2) (2016) 1519–1532.

[34] K. L. Wagstaff, B. Tang, D. R. Thompson, S. Khudikyan, J. Wyngaard, A. T. Deller, D. Palaniswamy, S. J. Tingay, R. B. Wayth, A machine learning classifier for fast radio burst detection at the VLBA, Publications of the Astronomical Society of the Pacific 128 (966) (2016) 084503.

[35] L. Connor, J. van Leeuwen, Applying deep learning to fast radio burst classification, arXiv preprint (2018) arXiv:1803.03084.

[36] A. Offringa, R. Wayth, N. Hurley-Walker, D. Kaplan, N. Barry, A. Beardsley, M. Bell, G. Bernardi, J. Bowman, F. Briggs, et al., The low-frequency environment of the Murchison Widefield Array: radio-frequency interference analysis and mitigation, Publications of the Astronomical Society of Australia 32 (2015).

[37] A. Offringa, J. Van De Gronde, J. Roerdink, A morphological algorithm for improving radio-frequency interference detection, Astronomy & Astrophysics 539 (2012) A95.

[38] L. W. Peck, D. M. Fenech, SERPent: Automated reduction and RFI-mitigation software for e-MERLIN, Astronomy and Computing 2 (2013) 54–66.

[39] J. Akeret, C. Chang, A. Lucchi, A. Refregier, Radio frequency interference mitigation using deep convolutional neural networks, Astronomy and Computing 18 (2017) 35–39.

[40] B. T. Indermuehle, L. Harvey-Smith, RFI mitigation through prediction and avoidance, in: 32nd International Union of Radio Science General Assembly & Scientific Symposium, 2017.

[41] C. Schollar, RFI monitoring for the MeerKAT radio telescope, Master's thesis, University of Cape Town (2015).

[42] A. Botha, H. Reader, J. Manley, S. Malan, H. Kriel, P. van der Merwe, P. Meyer, P. van der Walt, W. Croukamp, R. Anderson, Dynamic RFI measurement systems on a ROACH-2 platform, in: 2013 International Conference on Electromagnetics in Advanced Applications (ICEAA), IEEE, 2013, pp. 502–505.

[43] A. Botha, Development of a real-time transient analyser for the SKA, Ph.D. thesis, Stellenbosch: Stellenbosch University (2014).

[44] R. P. Millenaar, A. J. Otto, Innovations in instrumentation for RFI monitoring, in: Radio Frequency Interference (RFI), IEEE, 2016, pp. 65–68.

[45] P. Bolli, F. Gaudiomonte, F. Messina, R. Ambrosini, C. Bortolotti, M. Roma, The RFI monitoring systems for the Medicina and the Sardinia radio telescopes, PoS RFI2010 (2010) 29.

[46] B. T. Indermuehle, L. Harvey-Smith, C. Wilson, K. Chow, The ASKAP RFI environment as seen through BETA, in: Radio Frequency Interference (RFI), IEEE, 2016, pp. 43–48.

[47] P. Wiid, The answer is in fact 41, or how to get 35:1 bandwidth from a cone antenna, in: 2015 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), IEEE, 2015, pp. 1060–1063.

[48] J. Andriambeloson, P. Wiid, Time and frequency domain characterisation of a 3D-printed bi-conical antenna dispersion, in: 2016 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), IEEE, 2016, pp. 221–224.

[49] M. De Beer, Wideband direction finding of RFI for MeerKAT, Ph.D. thesis, Stellenbosch: Stellenbosch University (2017).

[50] J. Andriambeloson, P. Wiid, Hyperband bi-conical antenna design using 3D printing technique, in: IOP Conference Series: Materials Science and Engineering, Vol. 120, IOP Publishing, 2016, p. 012010.

[51] J. Andriambeloson, P. Wiid, A 3D-printed PLA plastic conical antenna with conductive-paint coating for RFI measurements on MeerKAT site, in: 2015 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), IEEE, 2015, pp. 945–948.

[52] I. E. Portugues, P. J. Moore, I. A. Glover, R. J. Watson, A portable wideband impulsive noise location system, IEEE Transactions on Instrumentation and Measurement 57 (9) (2008) 2059–2066.

[53] M. Shao, C. L. Nikias, Signal processing with fractional lower order moments: stable processes and their applications, Proceedings of the IEEE 81 (7) (1993) 986–1010.

[54] D. Middleton, Statistical-physical models of electromagnetic interference, IEEE Transactions on Electromagnetic Compatibility EMC-19 (3) (1977) 106–127.

[55] B. L. Evans, K. Gulati, M. Nassar, N. Aghasadeghi, A. Sujeeth, In-platform radio frequency interference mitigation for wireless communications, Embedded Signal Processing Laboratory, The University of Texas at Austin (2007).

[56] G. Tzagkarakis, J. P. Nolan, P. Tsakalides, Compressive sensing using symmetric alpha-stable distributions for robust sparse signal reconstruction, IEEE Transactions on Signal Processing 67 (3) (2019) 808–820.

[57] M. G. Sánchez, L. De Haro, M. C. Ramón, A. Mansilla, C. M. Ortega, D. Oliver, Impulsive noise measurements and characterization in a UHF digital TV channel, IEEE Transactions on Electromagnetic Compatibility 41 (2) (1999) 124–136.

[58] H. Kanemoto, S. Miyamoto, N. Morinaga, Statistical model of microwave oven interference and optimum reception, in: 1998 IEEE International Conference on Communications, 1998. ICC 98, Vol. 3, IEEE, 1998, pp. 1660–1664.

[59] M. Nassar, X. E. Lin, B. L. Evans, Stochastic modeling of microwave oven interference in WLANs, in: 2011 IEEE International Conference on Communications (ICC), IEEE, 2011, pp. 1–6.

[60] T. M. Taher, M. J. Misurac, J. L. LoCicero, D. R. Ucci, Microwave oven signal modelling, in: 2008 Wireless Communications and Networking Conference (WCNC), IEEE, 2008, pp. 1235–1238.

[61] M. Zimmermann, K. Dostert, Analysis and modeling of impulsive noise in broad-band powerline communications, IEEE Transactions on Electromagnetic Compatibility 44 (1) (2002) 249–258.

[62] L. D. Bert, P. Caldera, D. Schwingshackl, A. M. Tonello, On noise modeling for power line communications, in: 2011 IEEE International Symposium

on Power Line Communications and Its Applications (ISPLC), IEEE, 2011, pp. 283–288.

[63] H. Meng, Y. L. Guan, S. Chen, Modeling and analysis of noise effects on broadband power-line communications, IEEE Transactions on Power Delivery 20 (2) (2005) 630–637.

[64] T. Esmailian, F. R. Kschischang, P. Glenn Gulak, In-building power lines as high-speed communication channels: channel characterization and a test channel ensemble, International Journal of Communication Systems 16 (5) (2003) 381–400.

[65] M. Tlich, H. Chaouche, A. Zeddam, F. Gauthier, Impulsive noise characterization at source, in: Wireless Days, 2008. WD'08. 1st IFIP, IEEE, 2008, pp. 1–6.

[66] N. Andreadou, F.-N. Pavlidou, PLC channel: impulsive noise modelling and its performance evaluation under different array coding schemes, IEEE Transactions on Power Delivery 24 (2) (2009) 585–595.

[67] X. Dong, H. Weng, D. G. Beetner, T. H. Hubing, D. C. Wunsch, M. Noll, H. Gksu, B. Moss, et al., Detection and identification of vehicles based on their unintended electromagnetic emissions, IEEE Transactions on Electromagnetic Compatibility 48 (4) (2006) 752–759.

[68] H. Weng, X. Dong, X. Hu, D. G. Beetner, T. Hubing, D. Wunsch, Neural network detection and identification of electronic devices based on their unintended emissions, in: 2005 International Symposium on Electromagnetic Compatibility, Vol. 1, IEEE, 2005, pp. 245–249.

[69] M. Gulati, V. K. Singh, S. K. Agarwal, V. A. Bohara, Appliance activity recognition using radio frequency interference emissions, IEEE Sensors Journal 16 (16) (2016) 6197–6204.

[70] C. J. Wolfaardt, D. Davidson, T. Niesler, Statistical classification of radio frequency interference (RFI) in a radio astronomy environment, in: Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), IEEE, 2016, pp. 1–5.

[71] L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE 77 (2) (1989) 257–286.

[72] C. Dietrich, G. Palm, K. Riede, F. Schwenker, Classification of bioacoustic time series based on the combination of global and local decisions, Pattern Recognition 37 (12) (2004) 2293–2305.

[73] B. Barshan, B. Ayrulu, Fuzzy clustering and enumeration of target type based on sonar returns, Pattern recognition 37 (2) (2004) 189–199.

[74] O. Ureten, N. Serinken, Wireless security through RF fingerprinting, Canadian Journal of Electrical and Computer Engineering 32 (1) (2007) 27–33.

[75] C. Bertoncini, K. Rudd, B. Nousain, M. Hinders, Wavelet fingerprinting of radio-frequency identification (RFID) tags, IEEE Transactions on Industrial Electronics 59 (12) (2012) 4843–4850.

[76] J. Hall, M. Barbeau, E. Kranakis, Detection of transient in radio frequency fingerprinting using signal phase, Wireless and Optical Communications (2003) 13–18.

[77] R. W. Klein, M. A. Temple, M. J. Mendenhall, D. R. Reising, Sensitivity analysis of burst detection and RF fingerprinting classification performance, in: IEEE International Conference on Communications, 2009. ICC'09., IEEE, 2009, pp. 1–5.

[78] ROACH, Collaboration for Astronomy Signal Processing and Electronics Research, [Online]. Available: https://casper.berkeley.edu/wiki/ROACH, [Accessed: 2016-09-27] (2013).

[79] M. Wagner, J. Manley, W. New, A. Siemion, Introduction to Simulink, Collaboration for Astronomy Signal Processing and Electronics Research (2013).

[80] ADC2x1000-8, Collaboration for Astronomy Signal Processing and Electronics Research, [Online]. Available: https://casper.berkeley.edu/wiki/ADC2x1000-8, [Accessed: 2016-01-06] (2012).

[81] V. Ravi, R. M. Shannon, M. Bailes, K. Bannister, S. Bhandari, N. D. R. Bhat, S. Burke-Spolaor, M. Caleb, C. Flynn, A. Jameson, et al., The magnetic field and turbulence of the cosmic web measured using a brilliant fast radio burst, Science 354 (6317) (2016) 1249–1252.

[82] J. W. Hessels, S. M. Ransom, I. H. Stairs, P. C. Freire, V. M. Kaspi, F. Camilo, A radio pulsar spinning at 716 Hz, Science 311 (5769) (2006) 1901–1904.

[83] M. Young, R. Manchester, S. Johnston, A radio pulsar with an 8.5-second period that challenges emission models, Nature 400 (6747) (1999) 848.

[84] S. Sallmen, D. Backer, T. Hankins, D. Moffett, S. Lundgren, Simultaneous dual-frequency observations of giant pulses from the Crab pulsar, The Astrophysical Journal 517 (1) (1999) 460.

[85] A. Leshem, A.-J. van der Veen, A.-J. Boonstra, Multichannel interference mitigation techniques in radio astronomy, The Astrophysical Journal Supplement Series 131 (1) (2000) 355.

[86] H. W. Lilliefors, On the Kolmogorov-Smirnov test for normality with mean and variance unknown, Journal of the American Statistical Association 62 (318) (1967) 399–402.

[87] S. S. Shapiro, M. B. Wilk, An analysis of variance test for normality (complete samples), Biometrika 52 (3/4) (1965) 591–611.

[88] V. A. Epanechnikov, Non-parametric estimation of a multivariate probability density, Theory of Probability & Its Applications 14 (1) (1969) 153–158.

[89] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of cognitive neuroscience 3 (1) (1991) 71–86.

[90] B. Schölkopf, A. Smola, K.-R. Müller, Kernel principal component analysis, in: Artificial Neural Networks–ICANN'97, Springer, 1997, pp. 583–588.

[91] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, NY, USA, 2006.

[92] C. Thornton, Separability is a learner's best friend, in: 4th Neural Computation and Psychology Workshop, Springer, 1998, pp. 40–46.

[93] J. Greene, Feature subset selection using Thornton's separability index and its applicability to a number of sparse proximity-based classifiers, in: Twelfth Annual Symposium of the South African Pattern Recognition Association, 2001.

[94] J. Cohen, A coefficient of agreement for nominal scales, Educational and psychological measurement 20 (1) (1960) 37–46.

[95] P. J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, Journal of computational and applied mathematics 20 (1987) 53–65.

[96] B. H. Juang, L. R. Rabiner, Hidden Markov models for speech recognition, Technometrics 33 (3) (1991) 251–272.

[97] M. Benzeghiba, R. De Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, et al., Automatic speech recognition and speech variability: A review, Speech Communication 49 (10-11) (2007) 763–786.

[98] J. Hu, S. G. Lim, M. K. Brown, Writer independent on-line handwriting recognition using an HMM approach, Pattern Recognition 33 (1) (2000) 133 – 147.

[99] H. Cooper, E.-J. Ong, N. Pugeault, R. Bowden, Sign language recognition using sub-units, Journal of Machine Learning Research 13 (Jul) (2012) 2205–2231.

[100] F. Pace, Automated classification of humpback whale (Megaptera novaeangliae) songs using hidden Markov models, Ph.D. thesis, University of Southampton (2013).

[101] B. Schölkopf, S. Mika, A. Smola, G. Rätsch, K.-R. Müller, Kernel PCA pattern reconstruction via approximate pre-images, in: ICANN 98, Springer, 1998, pp. 147–152.

[102] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al., A density-based algorithm for discovering clusters in large spatial databases with noise, in: Kdd, Vol. 96, 1996, pp. 226–231.

[103] Y. Zhu, K. M. Ting, M. J. Carman, Density-ratio based clustering for discovering clusters with varying densities, Pattern Recognition 60 (2016) 983–997.

[104] R. J. Campello, D. Moulavi, J. Sander, Density-based clustering based on hierarchical density estimates, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2013, pp. 160–172.

[105] M. Ankerst, M. M. Breunig, H.-P. Kriegel, J. Sander, OPTICS: ordering points to identify the clustering structure, in: ACM Sigmod record, Vol. 28, ACM, 1999, pp. 49–60.

[106] R. Weiss, S. Du, J. Grobler, S. Lebedev, G. Varoquaux, hmmlearn 0.2.1, [Online]. Available: https://github.com/hmmlearn/hmmlearn, [Accessed: 2017-11-08] (2017).

[107] A. Zeyer, P. Doetsch, P. Voigtlaender, R. Schlüter, H. Ney, A comprehensive study of deep bidirectional LSTM RNNs for acoustic modeling in speech recognition, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2017, pp. 2462–2466.

[108] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (8) (1997) 1735–1780.

[109] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint (2014) arXiv:1412.3555.

[110] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Processing Magazine 29 (6) (2012) 82–97.

[111] A. Graves, A.-r. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2013, pp. 6645–6649.

[112] K. Longi, T. Pulkkinen, A. Klami, Semi-supervised convolutional neural networks for identifying Wi-Fi interference sources, in: Asian Conference on Machine Learning, 2017, pp. 391–406.

[113] G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M. A. Nicolaou, B. Schuller, S. Zafeiriou, Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network, in: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2016, pp. 5200–5204.

[114] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, et al., Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.