“Classifying the Unknown” - Radio Frequency Signal Identification Utilising Neural Networks

Author: Jonathan Rawlinson - jr2014@ic.ac.uk
Supervisor: Esther Perea, Imperial College
September 30, 2018

Abstract

Future cognitive radio systems will be operating in very signal dense environments, with numerous generations of radio systems in operation utilising different protocols all in the same frequency bands. The aim of this project was to design and build a radio system that is capable of classifying different radio signals using convolutional neural networks and novel feature extraction methods. It was proposed that this technology can be used to build more reliable and more agile Dynamic Spectrum Access (DSA) systems. Overall, a robust classification system for signals in the shortwave radio bands has been designed and constructed using a HackRF Software Defined Radio (SDR) as the data source.

1 Introduction

The Radio Frequency (RF) spectrum is under high demand all over the world. With the advancement of Long Term Evolution (LTE, 4G cellular radio) and 5G on the way, the RF spectrum is going to be even harder to secure for new services due to the extra bandwidth that will be required. In addition to the bandwidth demand, new services also need to work around existing services such as Global System for Mobile Communications (GSM, 2G) or WiFi.

To enable future cognitive communications systems to work effectively and efficiently (i.e. avoiding transmitting over existing systems), they will need to know what type of systems are in operation around them in real time. There are a large number of modulation schemes and protocols that exist, all with different configurations and parameters (even different implementation of the same specification can have operational differences due to different manufacturers and tolerances). This presents a non-trivial challenge to the new communications system.

It was proposed that this signal classification problem can be solved utilising neural networks and the power of modern open-source software toolkits and frameworks.

2 Aims and Objectives

The aim of this project was to construct a system that is capable accurately identifying and classifying the signals within a HF radio band using novel techniques. The target band is the 40m amateur radio band and 41m Broadcast band. This translates to the frequency range 7.000 MHz - 7.450 MHz. The following signals will be “taught” to the classifier:

- AM (Amplitude Modulation) containing human speech or music
- SSB (Single sideband) containing human speech
- FSK (Frequency shift keying)
- CW (Morse code)
- Static carriers (carrier wave containing no data)

3 Literature Review

This section gives an overview of the current work in this research area. Two areas of the signal classification industry were analysed:

- The current academic progress
- The current state of the art commercial classification systems that are currently available

3.1 Current Academic Work

A large amount of work in signal classification and identification is currently taking place in this area by a number of industry sectors, the largest three being defence (for communications and surveillance operations), medicine (medical data analysis and prediction with a large amount of work going on in Electroencephalogram (EEG) analysis) and communications technology manufacturers (looking towards Dynamic Spectrum Access (DSA) and higher performance of existing specifications).

A system has been designed to use the time frequency properties of the input spectrum to classify different signal types [1] (in this case Bluetooth, 802.11b and 802.11g). This system extracts a number of spectral features (bandwidth, centre frequency and transmit time) and inputs these into a trained neural network.

Researchers from Ostwestfalen-Lippe University of Applied Sciences and Wroclaw University of Technology have developed a ”Neuro-Fuzzy Signal Classifier” that is able to classify signals to known communication standards. This system used the Power Spectral Density (PSD) as the identification parameter for signals [2]. PSD was used as the classification feature due to the higher computational efficiency of calculation compared to the Spectral Coherence Function (SCF).

A paper titled “Wireless Interference Identification with Convolutional Neural Networks” [3] from the Ostwestfalen-Lippe University of Applied Sciences describes a system designed to detect different signals in the 2.4 GHz band utilising a complex valued Fast Fourier Transform (FFT) as the classification input data. A convolutional neural network was trained to classify 15 different signal classes.

The Virginia Polytechnic Institute and State University published a paper on ”A New Approach to Signal Classification Using Spectral Correlation and Neural Networks”. This method uses the SCF as the main parameter for classification [4]. The SCF method of feature extraction has also been implemented as described in the papers: “Cyclostationary Approaches to Signal Detection and Classification in Cognitive Radio" [5] and "A New Approach to Improve Signal Classification in Low SNR Environment in Spectrum Sensing"[6]. The SCF method of feature extraction (otherwise known as cyclic spectral density) is popular in this field due to the fact that it provides a more information packed domain for signal analysis [7]. The major downside for this method is that computing the SCF is computationally expensive compared to other features (such as the PSD) [2]. A SCF method of classification using GNURadio is also described in the paper "Practical Signal Detection and Classification in GNU Radio" [8].

A paper titled "Convolutional Radio Modulation Recognition Networks" describes work that analyses the classification accuracy of training a convolutional neural network on raw time domain data and "expert features" which are composed of cyclic-moment based features [9]. It was found that using time domain data as the input to the CNN worked well compared to the "expert features" and is a high accuracy approach for signal classification. This work is also discussed and reinforced in a second paper [10]. Time domain data was also used as the classification input in the paper "Very Deep Convolutional Neural Networks for Raw Waveforms" where deep CNNs (up to 34 layers) were seen to outperform shallow CNNs [11].

Another method of signal classification is via the signal constellations as described in the thesis "Signal Detection and Digital Modulation Classification-Based Spectrum Sensing for Cognitive Radio" [12]. This project uses a multi-class SVM (Support Vector Machine) to classify the data.

It is noted that in the paper "Spectrum Monitoring for Radar Bands using Deep Convolutional Neural Networks" spectrograms and an amplitude-phase difference combination were used as classification parameters [13]. It was concluded that the amplitude-phase difference method was more robust to noise.

Feature extraction using wavelet transformations was implemented and tested to identify mosquito signatures from audio recordings. This method was found to be very effective and accurate, even marginally surpassing human experts [14].

In the paper "Fast and Unsupervised Classification of Radio Frequency Data Sets Utilizing Machine Learning Algorithms" 96 data sets were clustered and classified using time domain statistical features and spectrograms [15]. Overall it was found that using only three statistical features was required to produce satisfactory results.

Kickview Corporation have documented how to use CNNs with spectrogram data for accurate classification [16]. The spectrogram method has also been used for radar detection over tele-communication signals [17].

The work "Convolutional Neural Network for Classification of Solar Radio Spectrum" [18] uses a deep CNN to classify the spectrums of solar microwave bursts. A four layer CNN was found to produce classification results of between 83% and 89%. The topic and problem tackled in this paper directly relates to the subjects covered in this project.

In the paper "Deep Neural Network Architectures for Modulation Classification" [19] multiple CNN architectures were investigated with a basic 4 layer CNN providing a classification output of 83.8%. This basic CNN was then integrated with more advanced topologies such as LSTM (Long Short Term Memory) and CLDNN (Convolutional, Long Short-Term Memory) to increase the classification accuracy (CLDNN reached 88.5%).

A github project for classifying signals using deep CNNs and using a low cost SDR (the RTL-SDR) has been found to produce good results [20]. This technique initially used the raw IQ samples, restructured into a NxN array for classification. Then different pre-processing tactics were introduced including computing the FFT, demodulating the signal as if it was AM and demodulating the signal as if it was FM. Keras was used as the machine learning API.

The paper "Deep Architectures for Modulation Recognition" describes the analysis of a number of different deep classification networks for RF signal classification [21]. A number of standard modulations were investigated from the "RadioML2016.10a" data-set [22]. Overall it was found that the classification accuracy is not limited by the network architecture. To improve further classification accuracy novel transformation and feature generation techniques must be investigated furthur.

In the paper "Over the Air Deep Learning Based Radio Signal Classification" a moment based method of signal classification is investigated. A number of ML models were investigated including gradient boosted tree ensembles, CNNs and RNNs. RNNs were found to provide the optimum performance in this instance and found to achieve 95.6% test accuracy after training. This system was tested over the air using Ettus Research B210 SDRs (i.e. signal generation and reception) where the test accuracy dropped to 87%.

3.2 Existing Commercial Signal Classification Systems

- CRFS provide a "signal recognition" module for their analysis software "RFeye". This software uses machine learning techniques to classify the modulation schemes for RF signals that it has detected. Machine learning is used to improve classifier performance. [23, 24]
- The Rohde and Schwarz CA100 and CA120 is a PC based signal analysis package that takes input from a number of sources (including pre-recorded IQ data and Rohde and Schwarz SDRs). The system can automatically classify signals from a database provided by Rohde and Schwarz. The system is also able to demodulate the signals classified for further analysis. No in-depth information is provided to the public on how this classification unit operates. [25]
- Keysight Technologies provide software that is capable of detecting and classifying data from a number of sources. Their software is also able to locate the SoI using TDOA (Time Difference of Arrival) and RSS (Relative Signal Strength) techniques when multiple antennas are in use[26].

3.3 Discussion

In conclusion of the initial research, it has been shown that a large part of the signal classification system problem is selecting the classification features that will be analysed. This is a non-trivial problem that a large number of academics are working on.
The following classification parameters have been observed being used in the literature:

- Raw real/IQ data
- Complex and Magnitude Spectrogram
- PSD
- SCF
- CWT
- Bandwidth
- Transmit period

In addition to the above, the following machine learning tools have been observed in use for dealing with RF baseband data:

- Deep Neural Networks
- Deep Convolutional Neural Networks
- Multi-Class Support Vector Machines
- K-means clustering
- Support Vector Machines

It has was decided that the classifiers for this project will initially be formed of DCNNs. This has been chosen because a number of promising examples of signal classifiers using DCNNs have been shown in the published literature (4 layer convolutional layers).

4 Methodology

The project has been split into two major areas of work:

- Designing a software framework capable obtaining data from a Commercial Off The Shelf (COTS) SDR and classifying the detected signals.
- Investigating classification features and neural network architectures for suitability of classifying RF signals

5 System Design

The classification system is formed of a number of processing servers written in python that communicate over a network connection using ZMQ. The SoapySDR framework has been selected to interface with the COTS SDR (HackRF) as it has been found during testing to provide a more reliable API for this application than GNURadio (which was originally investigated). A 'spectrum processor" analyses buffers of data from the SDR and selects the SoIs that have been found using the detection algorithm. These signals are then extracted using frequency domain decimation. The Python packages SciPy and NumPy were used as the main tools to implement this. A "spectrum classifier" receives the individually selected signals, generates the classification features and produces the class results using an ensemble of pre-trained CNNs. The multiple class results are then converted into a single result using a majority voting system. The neural networks are designed and implemented in Keras with Tensorflow as the backend. The class results are then passed onto the webapp that provides a human usable interface to view the locations and classes of signals that have been detected.

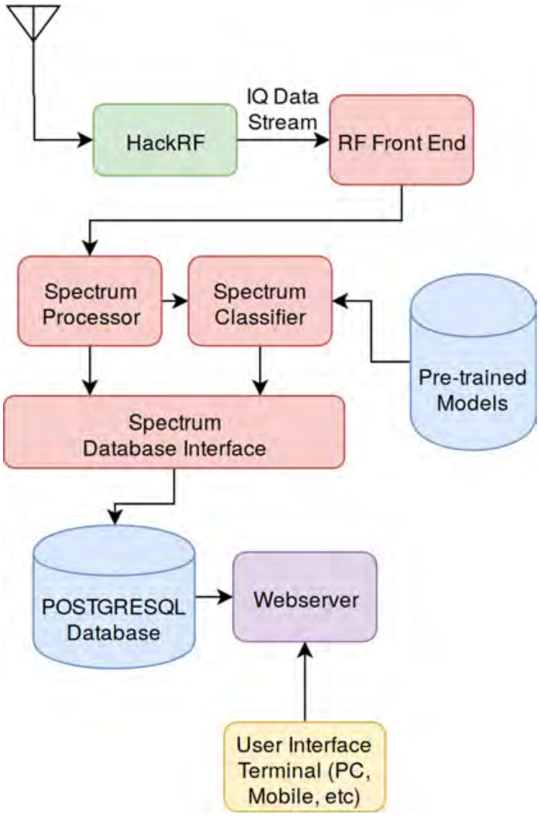

Figure 1: High Level System Diagram of the classification software "SignalDoctor"

6 Feature Generation

Feature calculation is a very important section of the classification process. Raw IQ data (without interpretation into another format, such as a waterfall plot) is not suitable for direct input into a neural network. The main reason for this is that the system must be able to take different signal bandwidths into account. Different bandwidth signals mean that the number of IQ samples can be different (over the same sampling time) which means that the neural network would have to have a variable input size which is outside the remit of this study. Feature generation allows different IQ sequences to be represented in a standard format and hence allows standard neural networking tools to be utilised.

6.1 Spectrogram Generation - Short Time Fourier Transform

The complex spectrogram is calculated using the `signal.stft()` function [27] provided by Scipy. This complex spectrogram is the main source of classification data for the classifiers. All classification data stems from this initial calculation.
Once the complex spectrogram has been calculated (this produces a complex 2D array), the array is shifted such that 0Hz is in the centre of the spectrum.

6.2 Magnitude Spectrogram

To compute the magnitude spectrum, the complex spectrum is squared and the absolute value is taken:

$$X_{mag} = |X_{cplx}|^2 \tag{1}$$

This produces a NxN array of real values that represents the magnitude time-frequency components of the SoI.

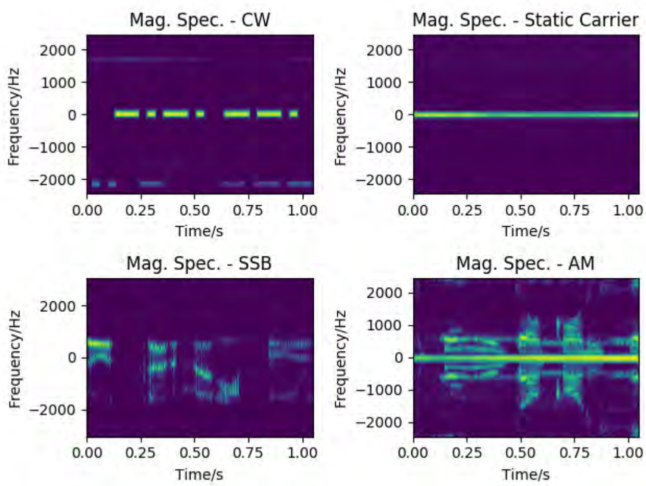

Figure 2: Magnitude Spectrogram Examples

Figure 2 shows four different magnitude spectrogram examples. The magnitude spectrogram is a very standard method of viewing RF signals due to the easy to recognise features produced.

6.3 Phase Spectrogram

The phase spectrogram was investigated in a similar way to the magnitude spectrogram. Unfortunately the phase spectrogram was found to not provide much defining information for classification purposes so was abandoned.

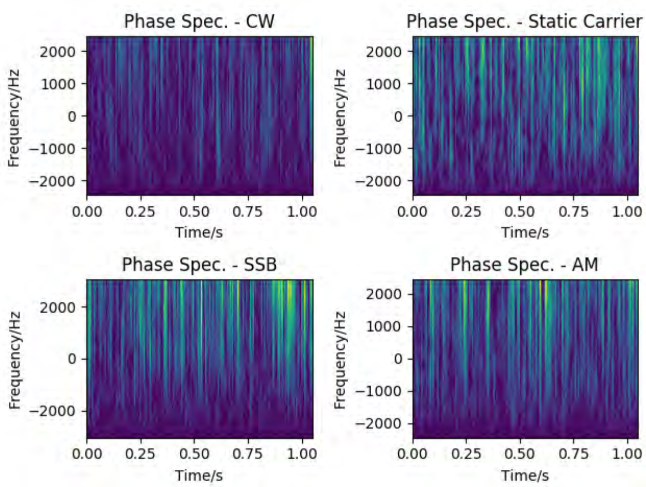

Figure 3: Phase Spectrogram Examples

Figure 3 shows examples of the phase spectrogram. This figure can be seen to not provide much information suitable for classification.

This feature is calculated by "unrolling" the calculated phase angle from the complex spectrogram along the frequency axis [28].

6.4 Mean PSD

The PSD is generated by calculating the mean of the magnitude spectrum along the time axis to obtain a time averaged PSD estimate. This method is detailed in figure 4.

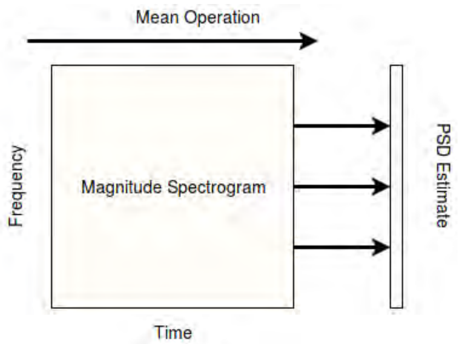

Figure 4: PSD Generation - The magnitude spectrogram has the mean calculated across the time axis to produce a 1xN output vector.

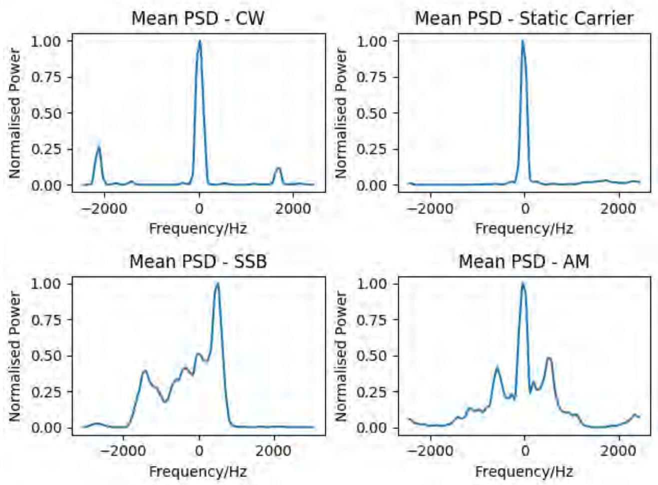

Figure 5: Mean PSD Examples

Example PSD plots are shown in figure 5. It is noted that the differences between AM and SSB are very apparent although it is not possible to differentiate the CW and static carrier plots as both appear as a single spike of power at a single frequency.

6.5 Variance PSD

The variance PSD was initially investigated due to the issue of detecting the difference in the PSD vector between a constant carrier signal and a modulating carrier (such as Morse code).
Variance is calculated as so[29]:

$$Var(x) = \frac{\sum_{i=0}^{n}(x_i - \bar{x})^2}{n-1} \tag{2}$$

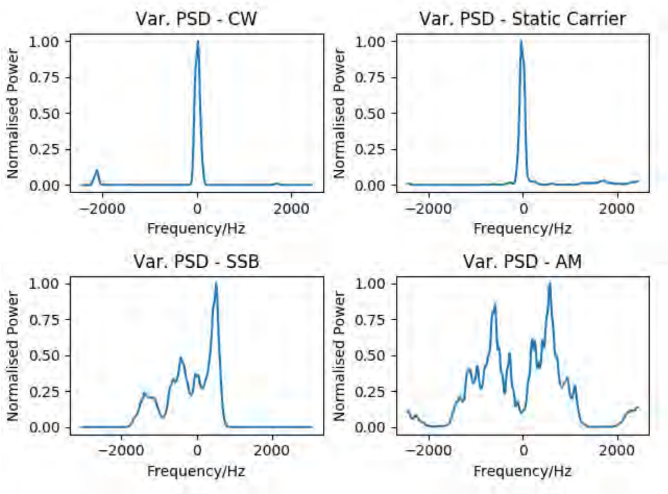

Figure 6: Variance PSD Examples

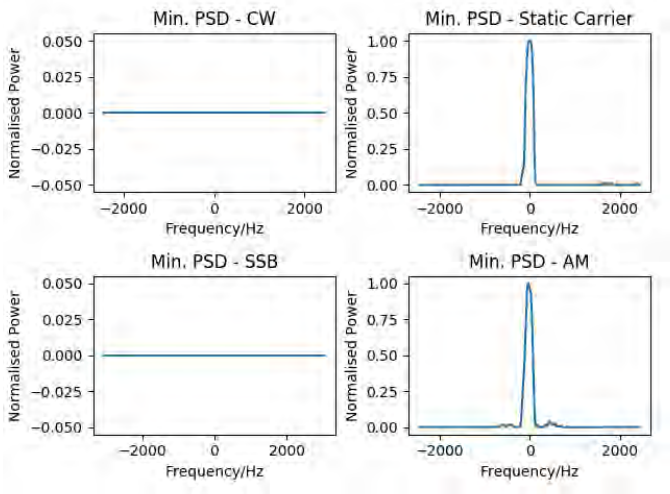

Figure 8: Min PSD Examples

Figure 6 shows a number of examples of the variance PSD. Unfortunately it is still not possible to differentiate between the CW transmission and the static carrier (due to the fact that the static carrier has a slight variance in time due to the radio channel conditions) although it is noted that the technique has had the desired effect on the AM signal where the centre carrier has disappeared compared to the mean PSD.

6.6 Min/Max PSD

The min/max PSDs were also investigated to solve the static carrier versus dynamic carrier classification problem. These PSD vectors are calculated in a similar way to the standard PSD although instead of calculating the mean of the magnitude spectrum, the minimum/maximum value is taken for each frequency bin.

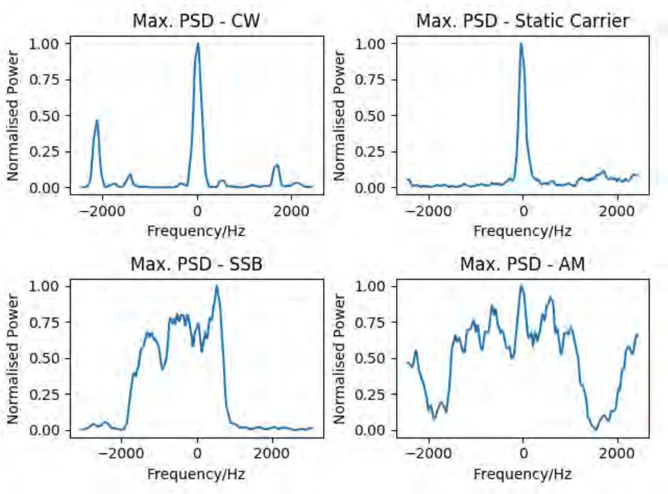

Figure 7: Max PSD Examples

Figure 7 shows examples of the max PSD. The response is very similar to the mean PSD and has the same classification issues associated with this.
Figure 8 shows examples of the min PSD. Here it can be seen that the CW and SSB signals produce a zero vector. For CW this is due to the "on/off" nature of the signal. For SSB this is because the human speech components vary a large amount in frequency and have no long term constant components. In the static carrier and AM signals the centre carrier is visible (as this never varies below a certain point).

6.7 Magnitude Variance and Auto-Correlation Coefficient Matrices

A number of NxN features generated from the magnitude spectrogram have been explored.
One defining feature of a number of signals is the symmetry in time and frequency (e.g. AM). The auto-covarience matrix was explored as a classification feature to represent the symmetrical features of signals. The covarience matrix is calculated as follows[30]:

$$C_{XX} = E\left[(X - \mu_X)(X - \mu_X)^T\right] \tag{3}$$

This feature was also further investigated by calculating the Pearson product-moment correlation coefficients (aka correlation coefficients). These are calculated using the previously calculated variance matrices as follows[31]:

$$R_{ij} = \frac{C_{ij}}{\sqrt{C_{ii} * C_{jj}}} \tag{4}$$

The two feature generation methods produce matrices that attempt to capture the correlation features in the magnitude spectrograms.
Figure 9 shows examples of the variance matrix. It can be seen that the SSB and AM matrices differ greatly. Unfortunately, once again the CW and static carrier signals are hard to differentiate due to the similarity of the signals.
Figure 10 shows examples of the correlation coefficient matrix. This figure shows that there are many different differing features produced by each signal. The CW and static carrier signals are still rather similar but it can be seen that the centre spike produced differs.

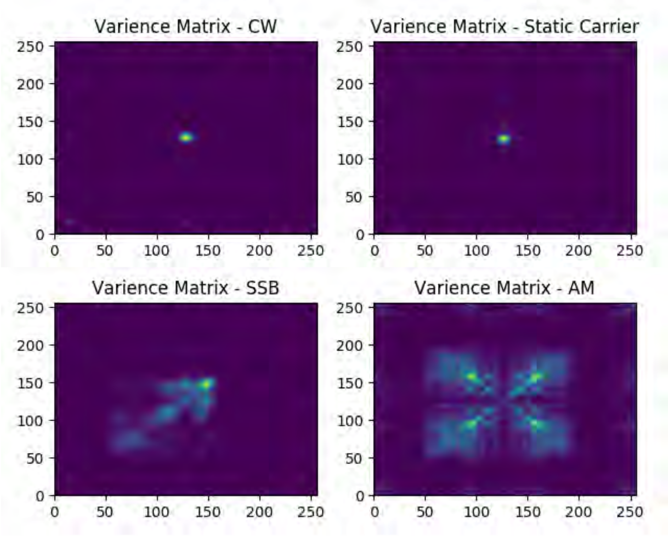

Figure 9: Variance Matrix Examples

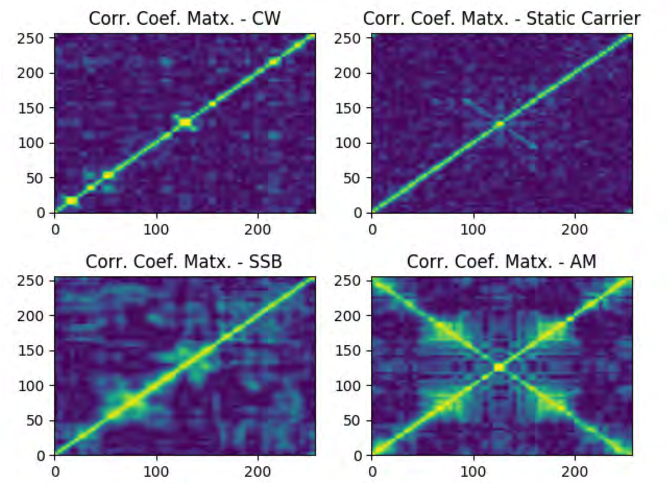

Figure 10: Correlation Coefficient Matrix Examples

6.8 Differential Spectra and Spectrograms

Another method investigated to determine relative changes in the processed signals was calculating the differentials of the magnitude spectrum in both the time and frequency directions.

Taking the differential in the time direction allows constant signals (such as static carriers) to be filtered out, only leaving the changing parts of the signal.

Taking the differential in the frequency direction attempts to filter out broadband changes in amplitude. This can help to remove broadband interference and leave only the SoI for further processing.

To improve upon the variance spectrum discusses earlier, the mean of the time differential matrix allows a more representative concept of change to be presented. This is calculated as shown below. The derivative is taken in the time domain and then summed across the time domain.

$$Z_{array} = \sum \left| \frac{d}{dx} S(x) \right| \tag{5}$$

Figure 11 shows examples of the differentiated magnitude spectrum in the frequency domain. It can be seen that using this method produces two magnitude spikes on frequency features of the original signal.

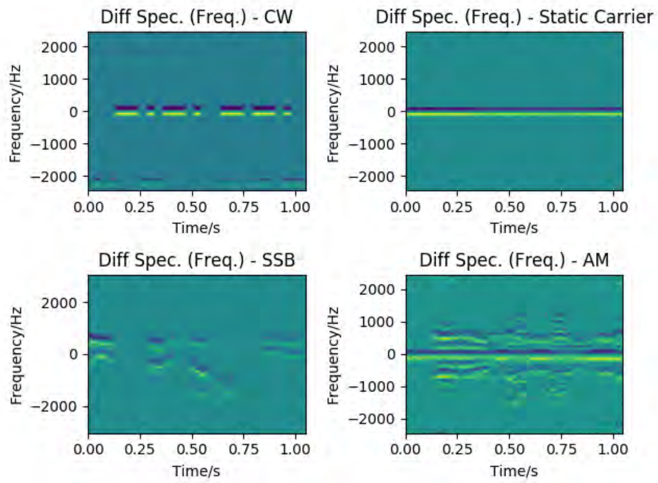

Figure 11: Differential Spectrogram (Frequency Axis) Examples

Figure 12 shows examples of the differentiated magnitude spectrum in the time domain. It can be clearly seen that any sharp changes in magnitude over time produces a spike in the differential output. Unfortunately this process is very sensitive to amplitude changes as can be seen by the static carrier plot.

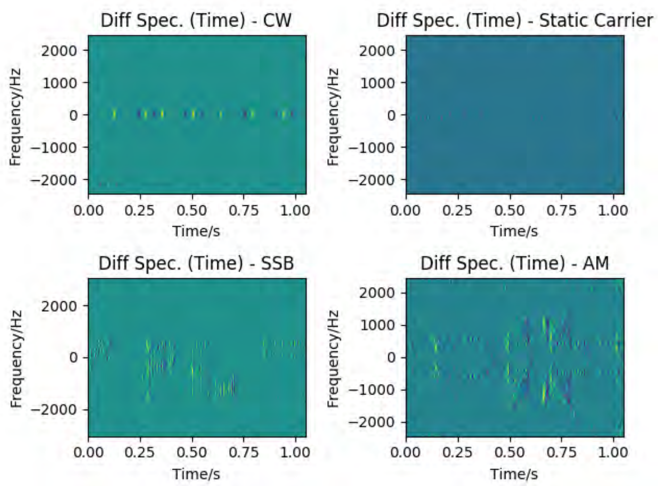

Figure 12: Differential Spectrogram (Time Axis) Examples

Figure 13 shows the differential PSD. Unfortunately due to the normalisation function it it very hard to differentiate the CW and static carrier. It is also noted that the output vector is very noisy.

6.9 Fourier Transform of Spectrograms

The absolute 2D Fourier transform of the magnitude spectrogram was investigated as a classification feature. As expected this compressed the majority of the data into the low frequency parts of the transform.

The feature is computed as below:

$$X_{fftabs} = |fft_{2D}(X_{mag})| \tag{6}$$

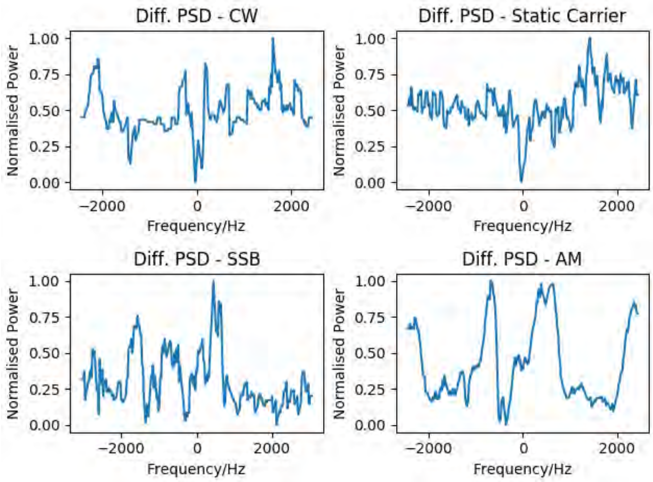

Figure 13: Differential Spectrum (Mean across time axis) Examples

The output real value matrix is then shifted such that the DC frequency bin is in the centre of the matrix for easier analysis.

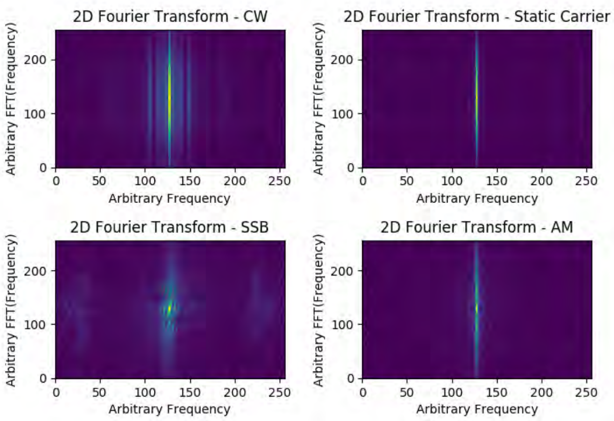

Figure 14: 2D Fourier transform of the magnitude spectrogram.

Examples of the 2D Fourier transform are shown in figure 14. It can be seen that each signal produces different features in the Fourier transform.

- The CW signal produces a lot of high frequency harmonics (depicted as vertical lines going out left and right from the centre spike) due to the sharp transitions in amplitude (as an instantaneous transmission produces infinite harmonics).
- As the static carrier by definition carries little data, therefore only produced a centre spike at DC.
- The SSB example has the interesting effect of producing a herring-bone style signal and the feature produced is not symmetrical.
- The AM signal shows a similar herring bone structure (albeit less pronounced) and produces a symmetrical transform.

Overall a large number of feature generation techniques have been shown. The features chosen for the classification system are listed in section 8.1.

7 Neural Network Optimisation

A number of classification features from the detected signals have been investigated including: standard PSD estimate, maximum PSD, minimum PSD, variance PSD, magnitude spectrogram, correlation coefficient matrix and quefrency-frequency matrix.

A number of different neural network configurations have been investigated around the following basic structure:

- Input convolutional layer
- Convolutional Layer - Pooling layer - Dropout layer (Unit 1)
- (Repeat (Unit 1) a number of times) and flatten output
- Dense layer - Dropout layer (Unit 2)
- (repeat (Unit 2) a number of times)
- Output layer

8 Results

8.1 Feature Results

The following 1D features have been chosen to be used for the classification system:

- Power Spectral Density - mean
- Power Spectral Density - max
- Power Spectral Density - min
- Power Spectral Density - variance

The following 2D features have been chosen:

- Magnitude Spectrum
- Autocorrelation Coefficient Matrix
- Fourier Spectrogram

These features have been chosen over the others as they provide a number of clearly visible differences between the signal classes being investigated which can be easily observed.

8.1.1 Power Spectral Density - Mean

The mean PSD feature was found to have a testing accuracy of 94.2259%. This PSD type provides a high degree of accuracy (greater than 89%) for all signals apart from CW. It is noted that AM and CW get confused by a large margin, presumably due to the similarity of the signals (the PSD of a AM signal with low speech components and a CW transmission are very similar.

8.1.2 Power Spectral Density - Max

The max PSD feature was found to have a testing accuracy of 88.1980%. Once again, CW and AM are regularly confused. It is also noted that SSB has a relatively high confusion rate. Also, the accuracies for FSK and SSB are lower than 85%

8.1.3 Power Spectral Density - Min

The min PSD feature was found to have a testing accuracy of 80.9010%. The overall classification rate for the Min PSD is very low with SSB preforming the worst with 45% accuracy.

8.1.4 Power Spectral Density - Variance

The variance PSD feature was found to have a testing accuracy of 91.8147%. Although this is not as high as the mean PSD value, it is the 1D feature set to have the highest minimum classification accuracy. The lowest class accuracy is 85%.

8.1.5 Magnitude Spectrum

The magnitude spectrogram achieved a testing accuracy of 95.1142%. This feature was the highest performing feature set. Interestingly, there was high confusion between CW and SSB. This was not expected and will be further investigated at a later date.

8.1.6 Autocorrelation Coefficient Matrix

The correlation coefficient matrix achieved a testing accuracy of 69.6701%. This was the worst performing feature set out of the selection. There is a high amount of confusion between a number of classes, with CW only having 11% classification accuracy.

8.1.7 Fourier Spectrogram

The Fourier transform of the magnitude spectrogram achieved a testing accuracy of 92.5127%. This feature set has the highest minimum value out of all the classifiers of 86%. This indicates that it is a very reliable classifier (albeit not the “best” performing classifier).

8.2 Final Ensemble Results

A CNN with 4 convolutional layers and two dense layers at the output was found to work optimally for the 2D features the dataset. This architecture family was also found to work well by other academics [18, 19]. The full network ensemble achieved 98.6% testing accuracy on the five classes although it is noted that this is a very simple scenario and should be tested with more classes.

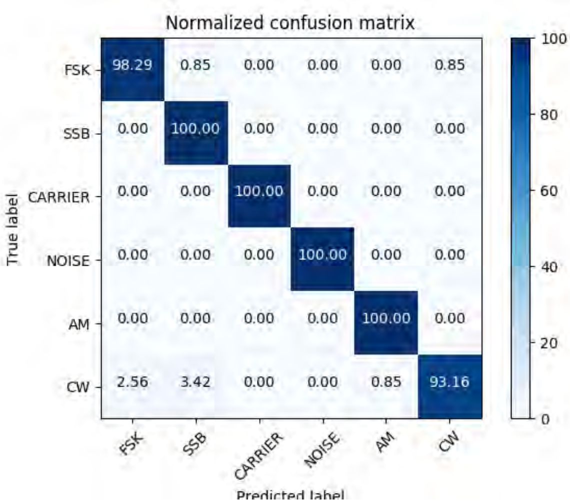

Figure 15: Confusion matrix for the classifier ensemble

Overall, the magnitude spectrogram was found to be the highest performing feature set and the Fourier spectro-

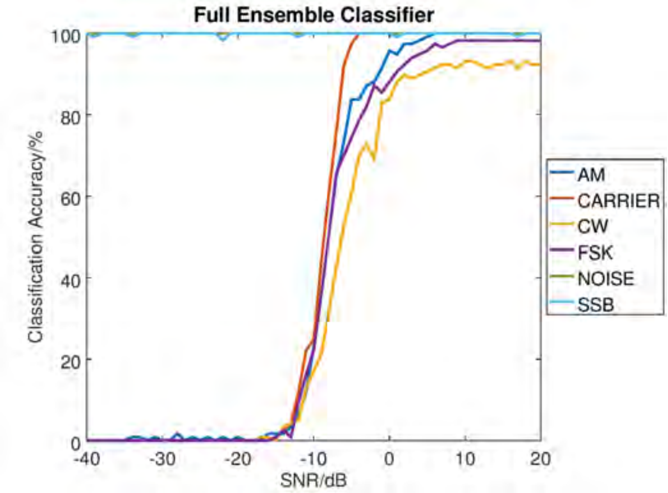

Figure 16: SNR results for the classifier ensemble

gram matrix was found to provide the most reliable classification (highest minimum classification value indicating that it is the least bias against a certain class).
It is also noted that the author investigated the use of regional CNNs to solve the signal location and classification challenge. This showed promising results and was able to segment and classify SSB signals from a magnitude spectrogram.

9 Concept Demonstrator

This section gives an overview of the concept demonstrator produced to showcase the work from this project. The system consists of a physically separated RF front end and back-end processing server. All of the base-band processing is carried out on a single laptop for the purposes of this demo.

9.1 RF Front End

The RF front end provides the system with raw baseband data. This data is streamed across an Ethernet link to the back-end server. A raspberry pi is used to host the SoapySDRServer utility provided by the developers of SoapySDR. A photo of the front end is shown in figure 17.

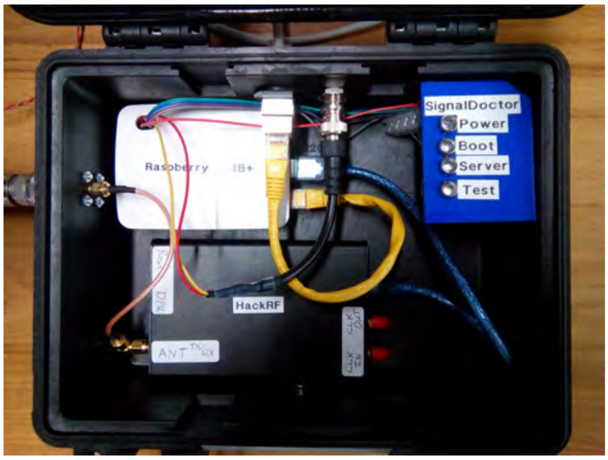

Figure 17: Photo of the RF Front End

A HackRF is used as the radio for this project, connected to the Raspberri Pi via a USB cable. The HackRF is then connected to a external antenna.

The two amateur radio HF antennas used during development of the project are as below:

- Super Antenna MP1 - Large tripod mounted antenna - Figure 18a
- Wonder Wand Widebander - Smaller case mounted antenna - Figure 18b

(a) “MP1 Super Antenna”

(b) “Wonder Wand Widebander”

Figure 18: HF Antennas

The whole RF front end system is housed in a plastic flight case to both protect the (expensive!) internal components and allow the unit to be easily transported.

A indicator panel has been included to assist the user in setting up the system. This panel contains four LEDs:

- Power - Indicates whether the 5.1V supply is active
- Boot - Indicates whether the Raspberry Pi has booted correctly
- Server - Indicates whether the radio server is active and accepting connections from the back end
- Test - When the system starts, a LED test is carried out. This LED indices that a test is in progress and the status LEDs should be ignored by the user until the test is complete

These LEDs have been selected to be be different colours to allow for easy identification by the user. It is also noted that the LEDs have been physically separated and clearly labelled such that a user with colour-blindness and/or poor eyesight is able to setup the system without assistance. A photo of the status panel is shown in figure 19.

Figure 19: Photo of the RF Front End Status Panel

The connection to the antenna is via a RF N connector (figure 20aon the outside for ruggedness and a SMA connector is used on the inside of the unit for easy integration. Power is supplied over a BNC connector (centre tip positive) and Ethernet is provided over a pass-through connector (figure 20b).

(a) N Antenna Connector

(b) Power and Ethernet Connectors

Figure 20: Demonstrator Electrical Connections

9.2 Network Interconnection

The network interconnection was synthesised using a simple Ethernet switch. In the real system, this would be replaced with a connection device such as 4G modem, satellite link or cable link. For the purposes of this demo it was decided to provide a “perfect link” (i.e. no other users, low latency, etc).

10 Impact and Outcome

When integrated into a next generation cognitive radio system, interference to existing systems will be reduced. Overall next generation cognitive radio systems will be more compatible due to all systems knowing the location and class of existing signals when using this technology which will in turn create a safer and connected world.

11 Future Work

The next steps for this work will be to expand the number of signals recognised by the classification networks to assess the scalability of the architecture. Gathering data and labelling signals requires a large amount of time due to the number of training examples required. In addition to this, increasing the throughput of the system by the use of a compiled language (such as C++) and the integration of FPGAs will be investigated.

12 Conclusion

Overall it has been confirmed that it is possible to classify RF signals utilising deep neural networks and deep convolutional neural networks. In addition it has been shown that a number of different feature generation techniques are capable of producing data suitable for RF signal classification. Also, a physical concept demonstrator has been produced that is capable of classifying RF signals in real time.

References

[1] A. N. Mody et al. “Recent advances in cognitive communications”. In: *IEEE Communications Magazine* 45.10 (Oct.

Figure 21: Image of the (happy!) author and the RF front end unit. Picture courtesy of Katrina Vu.

2007), pp. 54–61. ISSN: 0163-6804. DOI: 10.1109/MCOM.2007.4342823.

[2] K. Ahmad et al. "Neuro-fuzzy signal classifier (NFSC) for standard wireless technologies". In: *2010 7th International Symposium on Wireless Communication Systems*. Sept. 2010, pp. 616–620. DOI: 10.1109/ISWCS.2010.5624411.

[3] Uwe Meier Malte Schmidt Dimitri Block. *Wireless Interference Identification with Convolutional Neural Networks*. https://arxiv.org/pdf/1703.00737.pdf. Mar. 2017.

[4] A. Fehske, J. Gaeddert, and J. H. Reed. "A new approach to signal classification using spectral correlation and neural networks". In: *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005*. Nov. 2005, pp. 144–150. DOI: 10.1109/DYSPAN.2005.1542629.

[5] Kyouwoong Kim Ihsan A. Akbar Kyung K. Bae Jung-sun Um Chad M. Spooner and Jeffrey H. Reed. *Cyclostationary Approaches to Signal Detection and Classification in Cognitive Radio*. http://ieeexplore.ieee.org/document/1542629/. 2005.

[6] Dan Yu Hang Liu and Xiangwei Kong. *A New Approach to Improve Signal Classification in Low SNR Environment in Spectrum Sensing*. http://ieeexplore.ieee.org/document/4562526/. May 2008.

[7] Eric April. *The Advantage of Cyclic Spectral Analysis*. http://www.dtic.mil/get-tr-doc/pdf?AD=ADA258928. Dec. 1991.

[8] T O'Shea T Clancy H J. Ebeid. *PRACTICAL SIGNAL DETECTION AND CLASSIFICATION IN GNU RADIO*. http://www.wirelessinnovation.org/assets/Proceedings/2007/2007-sdr07-3.4-3-oshea.pdf. 2007.

[9] T J O'Shea J Corgan T C Clancy. *Convolutional Radio Modulation Recognition Networks*. https://arxiv.org/pdf/1602.04105.pdf. June 2016.

[10] T O'Shea and J Hoydis. *An Introduction to Deep Learning for the Physical Layer*. https://arxiv.org/pdf/1702.00832.pdf. July 2017.

[11] Wei Dai et al. *VERY DEEP CONVOLUTIONAL NEURAL NETWORKS FOR RAW WAVEFORMS*. https://arxiv.org/pdf/1610.00087.pdf. Oct. 2016.

[12] Curtis M. Watson. "Signal Detection and Digital Modulation Classification-based Spectrum Sensing for Cognitive Radio". AAI3604639. PhD thesis. Boston, MA, USA, 2013. ISBN: 978-1-303-60412-6.

[13] A Selim F Paisana J A Arokkiam Y Zhang L Doyle L A DaSilva. *Spectrum Monitoring for Radar Bands using Deep Convolutional Neural Networks*. https://arxiv.org/pdf/1705.00462.pdf. May 2017.

[14] Ivan Kiskina et al. *Mosquito Detection with Neural Networks: The Buzz of Deep Learning*. https://arxiv.org/pdf/1705.00462.pdf. May 2017.

[15] Phil Romero and Kalpak Dighe. *Fast and Unsupervised Classification of Radio Frequency Data Sets Utilizing Machine Learning Algorithms*. https://www.thinkmind.org/download.php?articleid=data_analytics_2015_8_30_60114. 2015.

[16] *Kickview Description*. https://blog.kickview.com. Accessed: 2018-01-24.

[17] Ahmed Selim et al. "Spectrum Monitoring for Radar Bands using Deep Convolutional Neural Networks". In: *CoRR* abs/1705.00462 (2017). arXiv: 1705.00462. URL: http://arxiv.org/abs/1705.00462.

[18] Sisi Chen et al. "Convolutional neural network for classification of solar radio spectrum". In: *2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*. July 2017, pp. 198–201. DOI: 10.1109/ICMEW.2017.8026227.

[19] Xiaoyu Liu, Diyu Yang, and Aly El Gamal. "Deep Neural Network Architectures for Modulation Classification". In: *CoRR* abs/1712.00443 (2017). arXiv: 1712.00443. URL: http://arxiv.org/abs/1712.00443.

[20] GitHub user randaller. *Deep learning signal classification using rtl-sdr dongle*. https://github.com/randaller/cnn-rtlsdr. Accessed: 2018-05-27. Feb. 2018.

[21] Nathan E. West and Timothy J. O'Shea. "Deep Architectures for Modulation Recognition". In: *CoRR* abs/1703.09197 (2017). arXiv: 1703.09197. URL: http://arxiv.org/abs/1703.09197.

[22] Timothy O'Shea and Nathan West. "Radio Machine Learning Dataset Generation with GNU Radio". In: *Proceedings of the GNU Radio Conference* 1.1 (2016). URL: https://pubs.gnuradio.org/index.php/grcon/article/view/11.

[23] *RFEye - Signal Recognition Module*. https://us.crfs.com/es/software/site/signal-recognition/. Accessed: 2018-01-24.

[24] *Machine Learning and RF Spectrum Intelligence Gathering*. www-media.crfs.com/media/file/GAyqhUXsSuqzD6fK1BELUQ/Machine_Learning_and_RF_Spectrum_Intelligence_Gathering.pdf. Accessed: 2018-01-28.

[25] *CA100 - PC-Based Signal Analysis and Signal Processing Software*. https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_brochures_and_datasheets/pdf_1/CA100_bro_en_3606-9340-12_v0401.pdf. Accessed: 2018-01-24.

[26] KeySight Technologies. *Spectrum Monitoring Software - A comprehensive set of tools for spectrum monitoring, interference analysis, signal identification and geolocation*. https://www.keysight.com/gb/en/software/application-sw/spectrum-monitoring-software.html. Accessed: 2018-05-30. 2018.

[27] The Scipy community. *scipy.signal.stft*. https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.stft.html. Accessed: 2018-06-11. May 2018.

[28] The SciPy community. *numpy.unwrap Documentation*. https://docs.scipy.org/doc/numpy/reference/generated/numpy.unwrap.html. Accessed: 2018-05-30. Apr. 2018.

[29] Steven W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing*. San Diego, CA, USA: California Technical Publishing, 1997, p. 14. ISBN: 0-9660176-3-3.

[30] National Institute of Standards and Technology (NIST) - Information Technology Laboratory. *Mean Vector and Covariance Matrix*. https://www.itl.nist.gov/div898/handbook/pmc/section5/pmc541.htm. Accessed: 2018-05-30. Oct. 2013.

[31] Paul Gingrich - UNIVERSITY OF REGINA. *Association Between Variables*. http://uregina.ca/~gingrich/corr.pdf. Accessed: 2018-05-30. Mar. 2018.