# Hierarchical Modulation Classification using Deep Learning

Garrett Vanhoy, Noah Thurston, Andrew Burger, Jacob Breckenridge, and Tamal Bose
Dept. of Electrical and Computer Engineering
The University of Arizona
Tucson, AZ 85721-0104
{gvanhoy, noahthurston, aburger, jbrecken, tbose}@email.arizona.edu

***Abstract*—Recent work in modulation classification using deep learning has produced promising results in classifying a diverse set of signals with only 128 IQ samples having undergone common radio impairments and frequency selective fading. Using deep learning as an approach differs significantly from established methods in classification where a set of expert features are derived and justified for specific channel conditions and computational resources. Instead, deep learning provides a way to efficiently identify features that are best suited for any set modulations, channel conditions, or available resources that may be of interest. However, deep learning approaches have only recently been used in this area and many questions about its limitations still exist. In this work, an expanded set of signals that include radar signals, multi-carrier signals, and higher order modulations are used with architectures presented in previous work to gain further insight into the flexibility of this approach. With a total of 29 signals to classify, it's shown how previous approaches can be augmented by training the network to classify signals on several hierarchical levels simultaneously with improved classification accuracy and a more flexible network architecture. Branch convolutional neural networks (B-CNN), which have been used to identify the subject of a body of text within a large hierarchy of subjects, are adapted to the problem of modulation classification to improve classification accuracy and facilitate the development of networks that can classify an even more diverse set of signals.**

***Index Terms*—Hierarchical Modulation Classification, Convolutional Neural Networks, Deep Learning**

## I. Introduction

Deep learning has been used to make great improvements in the state of the art in technical domains like computer vision, speech processing, and text classification. Recent applications of deep learning to problems in wireless communications show substantial promise for similar progress in the state of the art for many problems that are relevant to spectrum sharing including modulation classification [1], [2], interference identification [3], and traffic classification [4]. In the case of modulation classification, it has been shown that even relatively simple forms of deep learning can outperform the state of the art in terms of the number of samples needed to classify, the types of preprocessing necessary, the types of channels supported, the number and diversity of signals that can be classified, and overall performance.

Building upon this foundation, this work will demonstrate the performance of existing methods against an even larger and more diverse set of signals than previously tested and ultimately provide insight as to how a hierarchical approach to modulation classification may be a method to handle this. At the time of this writing, much of the existing work in this area uses a dataset, or some superset thereof [5]. The existence of this set is an excellent start and follow up works on the subject suggest possible extensions. However, it is evident that a broader conversation about what kinds of signals should be included in standard signal classification datasets should take place. It is likely that many standard datasets would help to represent the different applications for modulation classification because even some common waveforms, such as multi-carrier waveforms found in IEEE 802.11 and LTE, are not present in the set of classifiable signals. The dataset used in this work includes all of the signals used in [5], with the same type of channel impairments, but with the addition of several types of OFDM signals, radar signals, and higher order modulations.

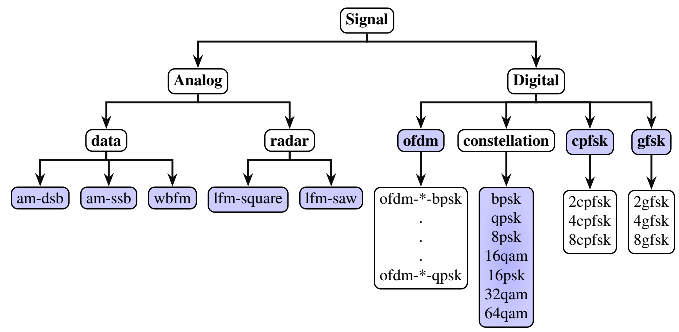


Fig. 1. Hierarchy for Classifying 29 Modulations

The inclusion of these new modulation orders and families, as it will become evident, sometimes requires more than just using a more complicated network architecture, but a different approach to network training as well. We propose the usage of a hierarchical structure to assist a neural network in first learning features that differentiate broader classes of signals, such as analog and digital, and then learning features to differentiate gradually finer-grained classes. The first such attempt at this using deep learning approaches is in [6], and we further expand on the concept by using branch convolutional neural networks [7], which allows for training classification at several hierarchical levels simultaneously.

In Section II, the background for existing deep learning based modulation classification is reviewed. In Section III, the models and assumptions are summarized. In Section IV, we provide the details for the hierarchical approach to modulation classification using convolutional neural networks. In Section V, we show and discuss the results of our classification method on more than 30 signals in harsh channel conditions and finally conclude in Section VI.

## II. Background

### A. Foundations of Deep Learning

One concept at the core of deep learning is that a complicated function or system can be approximated by a large complex network of very simple functions and systems. In deep neural networks, the simple functions are represented by “neurons” that take in a weighted sum of several inputs and are then fed into an activation function.

*1) Classic Neural Network:* Classic artificial neural networks are based on the model of actual neurons in the brain. All of the inputs $i$ into each node (neuron) are weighted by $w_i$, summed, and then ultimately passed through an activation function $\varphi(n)$ to get output $y_i$. These activation functions are typically sigmoidal in shape, and their outputs take on values between 0 and 1. An activation function commonly used in all networks presented in this paper is a rectified linear unit or ReLU. ReLU’s differ from other activation functions in that they do not take on values between 0 and 1. This helps mitigate the effect of vanishing gradients observed in the training of deep networks.

To train a network to produce a desired set of outputs for a set of inputs, an error $e_j$ for each neuron $j$ can be calculated as the difference between the desired and actual output summed across an entire layer to produce $E = \sum_j e_j^2$. The $i^{th}$ input from the previous layer to each neuron produces what is called an induced local field from a weighted sum of the inputs $\sum_j w_{ji} y_i$. To minimize the error $e_j$ at each node, it is necessary to effectively update the weights iteratively in the direction of $\frac{\partial E}{\partial w_{ji}}$. This is ultimately calculated as:

$$\frac{\partial E}{\partial w_{ji}(n)} = -e_j(n)\varphi_j'(V_j(n)y_i(n)) \tag{1}$$

This optimization process is commonly known as back-propagation because it requires propagating errors found at the network output back through the network to each layer.

*2) Convolutional Neural Networks:* Convolutional neural networks (CNNs) use filters, which is a common construct found in communication systems, to learn features which can distinguish between classes. Each CNN can have several filters of fixed length which are convolved with the input signal and fed into an activation to produce an output for the next later. As in seen in Figure 2, a convolutional layer uses a filter that has a length 2 and width 2 to transform an input vector that has two channels into an output vector that has a single channel. The output of one convolutional layer is a new set of feature maps and by cascading these layers together, the network acts as a filter. The mathematical output maps for each convolutional layer is shown in equation 2:

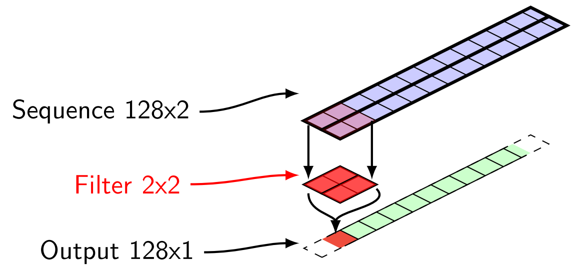


Fig. 2. Example Convolution in a Convolutional Neural Network

$$X_n = f\left(\sum_m \mathbf{W}_n^m * X^m + b_n\right), \tag{2}$$

where $X_n$ is the feature map output, $\mathbf{W}_n^m$ being the total number of filters being convolved with feature map $X^m$, and finally the bias term $b_n$. These convolutional layers are commonly used in image classification because they allow a network to more readily learn features that come from patterns observed by looking at input values that are close to each other and are likely to also work well with wireless signals for the same reason they work well with images.

### B. Prior Work

*1) Feature and Likelihood-based Approaches:* Existing methods in modulation classification can be largely separated into two types: likelihood-based and feature-based. Likelihood-based methods can be loosely described as feature extraction methods where the feature being extracted is a likelihood. These can be considered optimal in the sense that they achieve the least probability of misclassification. However, this comes at the cost of a computational complexity that eludes real-time implementation in many cases.

Among these approaches, the averaged likelihood ratio test (ALRT) performs the best. The ALRT is named as such because it attempts to look at the averaged distribution of the unknown parameters $\boldsymbol{v}_i$ with known probability density functions (PDF) conditioned on the $i^{th}$ modulation $H_i$. The likelihood function $\Lambda$ of the noisy received signal $r(t)$ under hypothesis $H_i$ is

$$\Lambda_A^{(i)}[r(t)] = \int \Lambda[r(t)|\boldsymbol{v}_i, H_i] p(\boldsymbol{v}_i|H_i) d\boldsymbol{v}_i. \tag{3}$$

The known PDFs are represented by $p(\boldsymbol{v}_i|H_i)$ and are effectively averaged through integration. This is the most accurate of the likelihood-based approaches and is often used as a theoretical upper bound for the probability of correct classification for other methods, but it is also the most computationally complex. However, as every unknown parameter about the channel model and every potential modulation that the ALRT attempts to estimate is taken into consideration,

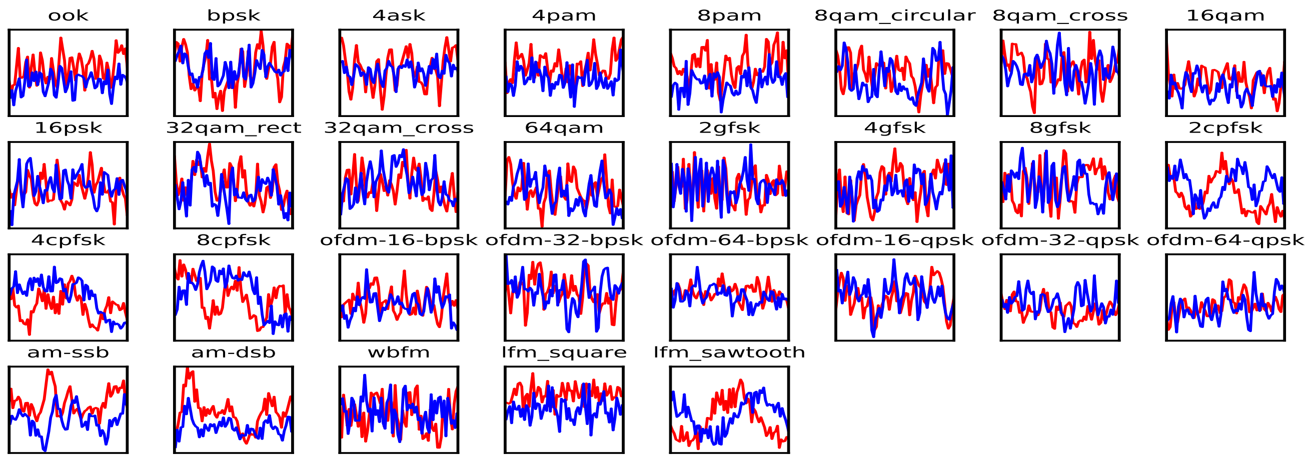


Fig. 3. 128 Raw IQ Samples of Included Modulations at 2 dB SNR in Rayleigh Frequency Selective Fading with Radio Impairments

the dimensionality of the computation increases, causing an exponential increase in computation.

On the other hand, a feature-based method extracts a set of descriptive values from the signal that differentiates each signal from each other. This approach is sub-optimal in terms of probability of correct classification, but substantially reduces the computational complexity for real-time applications. These features can include cumulants, time and frequency domain statistics, Wavelet Transform coefficients, or a combination of them. Finding the best set of features to accurately identify modulation schemes from each other has been the subject of the bulk of research in the area of modulation classification to date.

*2) Deep Learning Approaches:* Much of the innovation in deep learning has been in using novel architectures and network training methods that allow for using networks with even tens or hundreds of layers. Convolutional networks, recurrent networks, and multi-layer perceptrons are all commonly seen in the fields of voice recognition, image classification, and text processing. Although each of these types of networks can be trained to operate as complicated function approximators, their performance varies depending on the application and different architectures adopted for a specific application. For the classification of wireless signals, the most effective approaches use convolutional neural networks [3], [6], [8], recurrent neural networks [9] or a combination of them [2].

### C. Hierarchical Classification using B-CNN

Classifying modulations by grouping them into a hierarchical structure is not new [10] and is also a burgeoning concept used in the broader machine learning community [7] to do text classification. In this work, we focus on adapting the branch convolutional neural network (B-CNN) to the problem at hand. The B-CNN adds additional output layers on an existing network that are used to classify a different granularity of classes for each output layer. The extra output layers can be placed anywhere in an existing network but have generally been added at successively deeper layers. This results in earlier layers learning features to classify courser grained classes and later layers learning features to classify the most fine grained classes. The focus of our results are in demonstrating that using this branch-like structure in tandem with existing architectures, such as those mentioned in Section II-B2, can be used as a training method to stabilize gradients and ultimately improve accuracy. However, there are many questions about a hierarchical classification structure that can be expanded upon in further work regarding the method of constructing a good hierarchical structure, the placement of various branches, and the combination of different types of networks.

## III. Problem Formulation

### A. The Expanded Dataset

*1) Included Signals:* All of the included waveforms for classification can be seen in Figure 3. Orthogonal frequency division multiplexing (OFDM) signals, or more generally multi-carrier signals, are a common waveform used in protocols such as IEEE 802.11n and LTE. These waveforms transmit data by splitting large bandwidth channels into several more narrow channels in the frequency domain. This splitting is facilitated by using an FFT of a size at least as big as the number of channels that a stream can be divided, including a cyclic prefix. The dataset includes six OFDM signals, each with different numbers of sub-channels and sub-carrier modulations to emulate waveforms such as LTE and Wi-Fi.

Additionally, common radar signals are included. Although they seldom coexist with communication signals in the same frequency band, there are several applications where this stands to change. There are many different types of radar waveforms, some of which are indistinguishable from their communication waveform counterparts after normalizing signal power and bandwidth. It is because of this that we include two commonly linear frequency modulated signals.

Lastly, higher orders of M-ary CPFSK and GMSK, as well as higher order pulse amplitude modulations (PAM) and

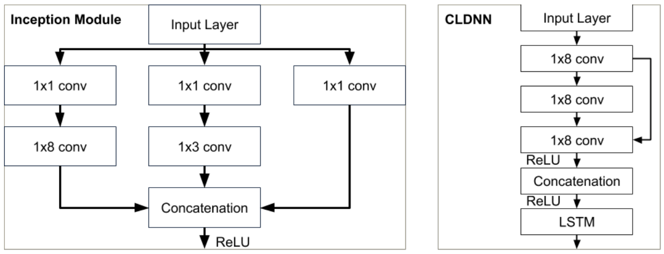


Fig. 4. Inception Module and CLDNN

quadratude amplitude modulations (QAM), are added. Both 4-ary and 8-ary versions of these signals are included in the dataset to explore whether or not deep learning can be used to classify these types of signals alongside others.

Channel impairments used in this dataset are the same as those in [5].

## IV. Technical Approach

### A. Baseline Approach

For the purposes of comparison, a relatively simple deep convolutional neural network architecture is employed. This network has two convolutional layers each with 20 filters of size 8. The output of the second convolutional layer is fed into a dense (fully connected) layer of 32 neurons with a ReLU activation function. The subsequent output is connected to a dense layer of 20 neurons with a softmax activation function, which essentially chooses the highest value coming from the weights sum of inputs. The hyper-parameters of this network were chosen according to what several recent works corroborate as adequate to get the close to the best performance out of this architecture.

### B. Existing Network Architectures

To leverage the existing work in the broader machine learning community, an inception module is also included in the types of architectures that are tested. An inception module is a type of layer based on concatenated convolutional layers that have different filter lengths. This layer structure allows for a network to learn features at different temporal scales simultaneously. The winner of one of the famous image classification tasks used this inception module in a deep network in 2015. In this case, three inception modules are cascaded each with an architecture shown in Figure 4.

Also, several architectures have been proposed for the modulation classification task, but only a few are relevant to the formulation that we employ. The convolutional long short-term deep neural network (CLDNN) architecture has been shown to consistently outperform other architectures in modulation classification [1]. This architecture, shown in Figure 4, also employs convolutional layers, a skip connection, and a type of recurrent layer called a long short-term memory (LSTM) module. Google has employed this architecture with its audio processing features because the LSTM has been shown to work well with time-series applications.

### C. Hierarchical Classification using B-CNN

B-CNN's were developed for text-classification but can be adapted for the problem at hand. but it is ambiguous as to where the branches can be attached. Since one purpose of making branches in the network is to force earlier layers to learn features to classify courser grained classes, the branches are made after each layer in the proposed networks. For example, the baseline approach will use three convolutional layers followed by two fully connected layers. When applying a hierarchical approach, a branch is made after each layer for successively finer grained classes. This pattern is repeated for the inception network but is different for the CLDNN, where instead all three branches are placed at the end of the network.

To effectively train a B-CNN, the loss function must be adapted after every epoch of training so that successively finer grained classes contribute more to the overall loss. This forces a network training to initially focus on classifying the most course grained classes first because a mis-classification at this level would necessarily result in worse classifications later.

*1) Choosing the Hierarchy:* The hierarchy chosen for our approach can be seen in Figure 1. This hierarchy was chosen according to what made the most sense, but results suggest that there can be improvements by choosing this properly. It is important to note here that all OFDM signals, CPFSK signals, and GMSK signals are grouped together because it was shown that these were not classifiable with any of the mentioned networks. This is further detailed in Section VI-A.

## V. Classification and Results

### A. Hierarchical Performance

Each of the previously mentioned network architectures in IV were implemented using the Keras framework for implementing deep learning on a P2 instance on Amazon AWS. 1,000 exemplars were generated for every SNR between -20 dB and 18 dB for a total of 20,000 signals exemplars. One third of the exemplars were used to train the network while the other two thirds were used to validate the network performance after every training epoch. Each network was trained using both a hierarchical approach based on B-CNN and a non-hierarchical approach for comparison and the overall classification accuracy for all 20 classes is presented in Figure 5.

The results are consistent with those found in [1] in that the CLDNN out performs both the baseline architecture and other architectures. The hierarchical approach appears to not have a significant affect for the performance of the baseline approach or the inception network, but has a marked increase on the performance of the CLDNN.

### B. Modulation Performance

Using the CLDNN architecture with hierarchical structure, the classification performance over SNR is plotted in Figure 6. It is apparent from these results that many of these modulations approach perfect classification as the SNR increases, but many modulations do not and hence the network architecture could not learn features that separate these classes even in high SNR.

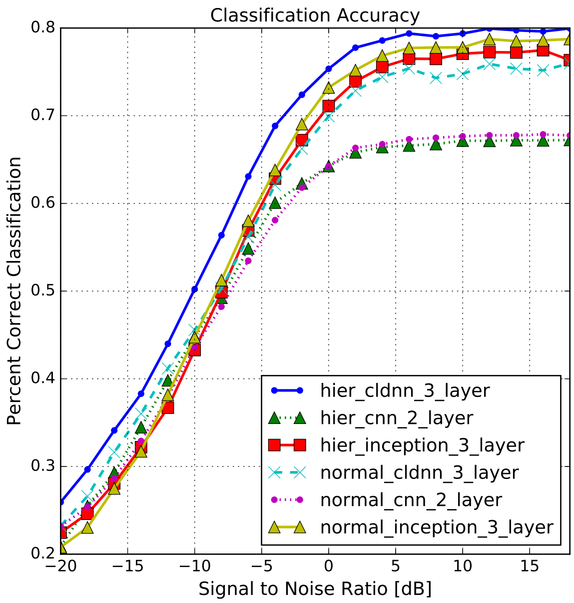


Fig. 5. PCC vs SNR Across All Trained Models

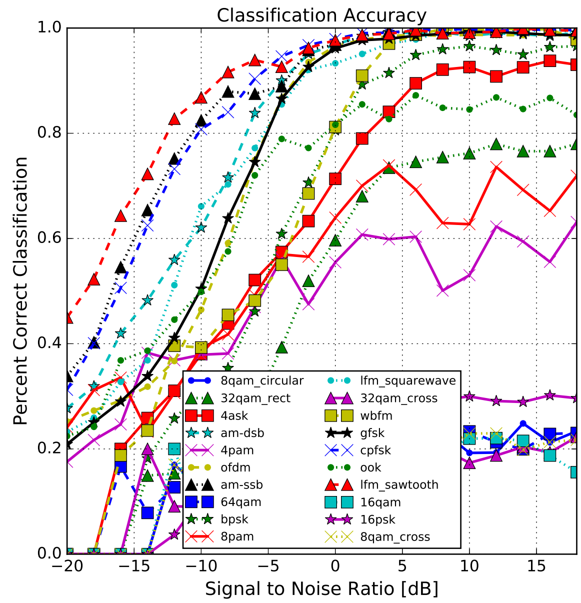


Fig. 6. PCC vs SNR Across All Modulation Groups

For example, signals that have M-ary values higher than 8 never exceed classification accuracies of 30%. Looking at the confusion matrix in Figure 7 which represents the confusion at 0 dB SNR, it is clear that the higher order modulations are often confused for each other. Intuitively it makes sense for these higher order modulations to often be confused with each other and difficult to classify for a sample size of 128 because all of the possible transitions between symbols may not be observed. It is expected that the classification accuracy of all of these modulations would significantly improve with the increase of the sample size as observed in [2].

350M

## VI. Discussion and Further Work

### A. *Classification of M-ary Signals*

One advantage of using a hierarchical structure for classification is the ability to use different network structures for different families of modulations. When attempting to classify the different types of OFDM, CPFSK, and GMSK signals, none of the existing architectures could do any better than guessing. For OFDM, each of the OFDM signals have a distinct feature that can be used to at least identify the number of sub-carriers based on how the signal repeats in time and with proper processing, the sub-carriers can be separated even in frequency-selective fading. Further work that includes subtypes of the OFDM signals will likely need more complex network architectures that can detect signal periodicities that are typically used to synchronize these signals in existing systems.

For CPFSK and GFSK-type signals, it is less clear as to why it is impossible to classify between these signals given the current network architectures. Even when the sample size is expanded to 1024 where more transitions between symbols can be observed, classification does not improve. These results suggest that future work towards expanding the set of possible classifiable modulations do not necessarily need to focus on being able to classify all of the presented modulations simultaneously, but rather the distinction between modulations within the same family.

### B. *Other Hierarchical Structures*

There might be many ways to choose a hierarchical structure and results may be better when chosen using more sophisticated methods. This work demonstrates how a hierarchy can be used to improve a convolutional classifier, however thus far the hierarchy has been determined solely using expert understanding of modulations. This assumes that the hierarchy identified by an expert coincides with the optimal hierarchy for classification by a neural network, which may not be true. It is possible to avoid this assumption and create a procedure for determining a hierarchy that is optimal for the classifier. One possibility to accomplish this is to use a convolutional autoencoder with a similar architecture to the classifier to extract features from the modulations and encode them into a lower dimensional vector space. Then, a clustering algorithm such as k-means can be used to cluster similar vectors into groups. These groups can then be clustered further, and a hierarchy can be constructed. If the convolutional autoencoder has a similar structure to that of the classifier and uses similar filters, then the vector embeddings it produces should indicate which modulations it can easily distinguish between and which modulations it cannot. and therefore produce an

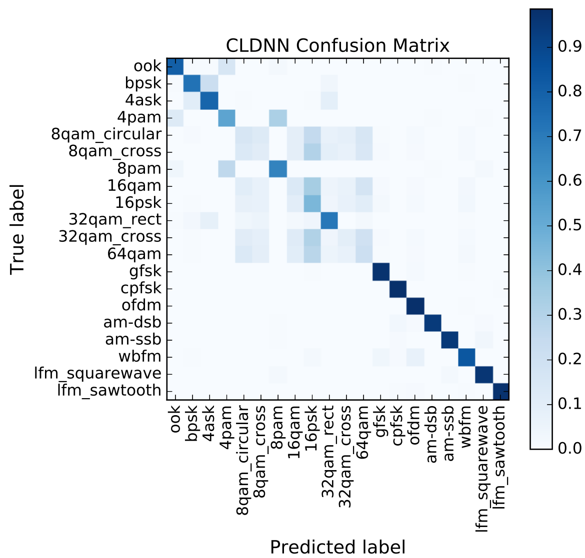


Fig. 7. Confusion Matrix at 0 SNR dB

optimal hierarchy for a given network architecture. However, the viability of this procedural approach for determining an optimal hierarchy is left for further work.

### *C. Dataset and Generation Code*

The dataset and code used to generate it will be posted.


## Acknowledgements

This project was partially supported by the Broadband Wireless Access and Applications Center (BWAC); NSF Award No. 1265960.


## References


[1] N. E. West and T. J. O'Shea, "Deep Architectures for Modulation Recognition," 2017. [Online]. Available: http://arxiv.org/abs/1703.09197

[2] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-Air Deep Learning Based Radio Signal Classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018. [Online]. Available: http://ieeexplore.ieee.org/document/8267032/

[3] S. Grunau, D. Block, and U. Meier, "Multi-Label Wireless Interference Identification with Convolutional Neural Networks," 2018. [Online]. Available: http://arxiv.org/abs/1804.04395

[4] T. J. O'Shea, S. Hitefield, and J. Corgan, "End-to-End Radio Traffic Sequence Recognition with Deep Recurrent Neural Networks," 2016. [Online]. Available: http://arxiv.org/abs/1610.00564

[5] T. J. O'Shea and N. West, "Radio machine learning dataset generation with gnu radio," *Proceedings of the 6th GNU Radio Conference*, 2016.

[6] K. Karra, S. Kuzdeba, and J. Petersen, "Modulation recognition using hierarchical deep neural networks," *2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pp. 1–3, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7920746/

[7] X. Zhu and T. Fujii, "Modulation classification for cognitive radios using stacked denoising autoencoders," *International Journal of Satellite Communications and Networking*, vol. 35, no. 5, pp. 517–531, 2017.

[8] S. Peng, H. Jiang, H. Wang, H. Alwageed, and Y.-D. Yao, "Modulation classification using convolutional Neural Network based deep learning model," *2017 26th Wireless and Optical Communication Conference (WOCC)*, no. 1, pp. 1–5, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7929000/

[9] D. Hong, Z. Zhang, and X. Xu, "Automatic Modulation Classification using Recurrent Neural Networks," no. 1, pp. 695–700, 2017.

[10] A. Swami and B. Sadler, "Hierarchical digital modulation classification using cumulants," *Communications, IEEE Transactions on*, vol. 48, no. 3, pp. 416–429, 2000. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=837045