AN ABSTRACT OF THE REPORT OF

Seifeddine Mejri for the degree of Master of Science in Electrical and Computer Engineering (ECE) presented on JUNE 16, 2020.

Title: Deep Learning Based Automatic Modulation Classification for Wideband Access Using Cyclostationarity Analysis

Abstract approved: ______________________________

Professor Bechir Hamdaoui

RF-based signal identification and classification has received a growing attention during the recent years due to its potential use in many application domains. Of particular interest is Automatic Modulation Classification (AMC), which has been useful to addressing various spectrum related challenges such as signal jamming and policy enforcement in spectrum sharing. Adopting AMC to wideband spectrum access presents, however, several practical challenges, mainly due to the high sampling rate, computational cost, and memory requirements, which make it unsuitable for real-time scenarios. To address these challenges, we combine the merits of cyclostationarity features and Convolutional Neural Networks (CNN) to propose an efficient wideband AMC technique. Our technique leverages Spectral Correlation Function (SCF) analysis to robustly identify modulation type of the occupied signal in each of the channels of the wideband spectrum. Our technique

does not require prior knowledge of signal parameters such as carrier frequency, symbol rate and phase offset. We show that out technique outperforms IQ based classifier in terms of accuracy especially under severe fading environment and requires less training time to converge. Finally, to reduce the cost furthermore, we establish a compressed learning scheme using a few measurements obtained by linear projections of the input features.

Deep Learning Based Automatic Modulation Classification for Wideband Access Using Cyclostationarity Analysis

by

Seifeddine Mejri

A REPORT

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented June 2020
Commencement June 2020

Master of Science report of Seifeddine Mejri presented on JUNE 16, 2020.

APPROVED:

Major Professor, representing Electrical and Computer Engineering (ECE)

of the School of Electrical Engineering and Computer Science (EECS) of Oregon State University

Dean of the Graduate School

I understand that my report will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my report to any reader upon request.

Seifeddine Mejri, Author

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special gratitude I give to my academic advisor, Professor Bechir Hamdaoui, whose contribution in stimulating suggestions and encouragements, helped me to coordinate my project, especially in writing this report. His expertise, suggestions, and constant support, added considerably to my graduate experience and my ability to carry out research. I want to express my sincere gratitude also to my committee members Professor Jinsub Kim and Professor Bella Bose, for accepting to serve as members of the committee of my defense. Finally, I want to thank my father and my mother who have been always there for me with their support, love, and encouragement.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

Chapter 1: Introduction

1.1 Objective

Radio frequency (RF) spectrum analysis is critical for spectrum awareness, policy enforcement, security, and interference mitigation. Automatic modulation classification (AMC) helps to automate the process of RF spectrum analysis by using state of the art decision theory and statistical pattern recognition. Recently, AMC has been an important topic in wireless communication due to the prospective use of convolutional neural networks and deep neural networks to apply classification instead of manual feature extraction combined with machine learning (ML). The advantage of deep learning (DL) over feature-based methods is flexibility and non-dependence on expert knowledge and more complex powerful models. However, DL simultaneously brings a lot of computation cost, especially when used with a large number of samples, making AMC computationally more expensive. With the growing demand towards application such AMC, computer vision, sensors based applications that require a lot of data to be processed, and with the increasing utilization of deep complex architecture such as VGG nets and GoogleNet, there is a need to look at this problem to reduce both the latency, throughput, and memory storage.

1.2 Limitations of Existing AMC Techniques

In wideband (WB) spectrum, there are three modulation classification techniques. Rule-based classification, classic machine learning, and representation learning. Rule-based systems are hand-designed if-then-else statements that embody the domain knowledge of human experts specialized in the field. A rule-based classifier is said to be limited in its ability to simulate intelligence. It is always limited by the size of its underlying rule base (also called knowledge base). On the other hand, in classical machine learning, all the features are engineered, and the system learns the features to the outputs. For both rule-based systems and classical machine learning, it is difficult to extract useful features, especially for very complex signals such as OFDM DSSS and CDMA. The recent advances in this field have used feature engineering to design high order features (moments, cumulants, and second order cyclostationarity of modulated signal) for classification. For instance, Aslam et al. [1] utilize the generic programming and K-nearest neighbors (GP-KNN) algorithms, which adopt the higher order cumulants(HOC) as features. Han et al. [2] propose a set of classifiers by using maximum-likelihood based (MLB) and feature-based (FB) classification. An approximate maximum likelihood (AML) classifier and a linear support vector machine (LSVM) are proposed with lower computation complexity than the conventional ML, but only for a known channel, which is not always the case. Furthermore, E. Like et al. [3] use spectral coherence function(SOF) to design a highly reliable classifier for multipath fading channel. El Khamy et al [4] consider the classification of multi-user chirp modulation signals using high order cumulant features, and they did the test with four classifiers. These classical machine learning approaches, used in this case, are vulnerable to the frequency and phase offset. Thus, their performance can be dramatically degraded by imperfect synchronization. Additionally, we cannot automate the process of modulation recognition. For instance, in rule-based classifiers, it is tough (to nearly impossible) to add rules to an already large knowledge base without introducing contradicting rules. This can be, in our case, when we want to add some new modulation to our dictionary. Indeed, we need to reconfigure the dictionary with the addition of the new modulation. This can become time-consuming and expensive, especially in large complex domains, and that's typically the showstopper for rule-based systems.

On the other side, representation learning [5] [6], however, goes one step further and eliminates the need to hand-design features. The important features are automatically extracted from within data using a representation of the input at the hidden layer(s), which is subsequently used for classification or regression at the output layer. Hence, in representation learning, the system automatically discovers the representations needed for feature detection or classification from raw data. We can think of feed-forward networks trained by using supervised learning as an example performing a kind of representation learning. In a like manner, multi-layer perceptron or convolution neural networks are used to learn more complex features in various levels or layers. It is shown that by using DL for AMC, promising results are found by T.O'shea [7], which conduct I/Q-based classifiers. The I/Q based classifier uses In-phase quadrature-phase components as input to the classification model to detect modulation.

This approach used in [8], is one of the first to conduct DL based multi-class classification. The signals are generated with different modulation types, and deep convolutional neural networks are directly applied to conduct the classification. For the most part, the complexity of the IQ in WB, makes it harder to perform classification. In particular, when we have a large number of samples, this becomes cumbersome and introduces high complexity which affects the training time and the inference time.

1.3 Our contribution

Since DL has a lot of computation costs and due to the high sampling rate, we propose channelization to be used at the receiver front end, which reduces the number of samples and allows parallel processing for features estimation. In addition, we try to solve AMC by using cyclostationary features instead of IQ as input features. Most of the work done does not consider the effect of channel impairments(including constructive and destructive interference, frequency and phase shifting), so a comprehensive study to compare both spectral and temporal IQ for classification under the effect of a variable path is conducted. Finally, we introduce an enhancement to the hardware implementation by using a compressed learning approach where features are learned based on a small number of measurements. This reduces the dimensionality of the data domain, while not losing much in terms of the classification accuracy. To the best of our knowledge, we don't know any work, paper, or journal which targets AMC similar to the way we do in this paper.

Chapter 2: Signal characterization

The system can be used in any band regardless of the frequency it can be in medium $300 - 3,000$ kHz, high $3 - 30$ MHz, very high $30 - 300$ MHz or ultra-high frequency $300 - 3,000$ MHz. Consider a WB spectrum of interest and multiple signals coming from multiple transmitters over the band, as explained in Figure 2.1. We have no information regarding the number of signals actively present(unknown) in the WB nor the waveform nor the carrier frequency of each signal. One user is occupying one channel, and we suppose that user signals are non-overlapping in frequency, since they could represent emitters from different services(without interference). A medium of transmission can be considered in this case to model any impairment on the composite signal. Additive white gaussian noise (AWGN) channel is often used as a model in which the only impairment to communication is a linear addition of WB noise with a constant spectral density. Another case could be using Rayleigh fading channel, which models multipath transmission in heavily built-up urban environments where we have scattering and no dominant line of sight(LOS). It produces a simple and tractable mathematical model, which is useful for gaining insight into the underlying behavior of the system with fading, frequency selectivity, interference.

The baseband signal received after the channel impairments, is the corrupt

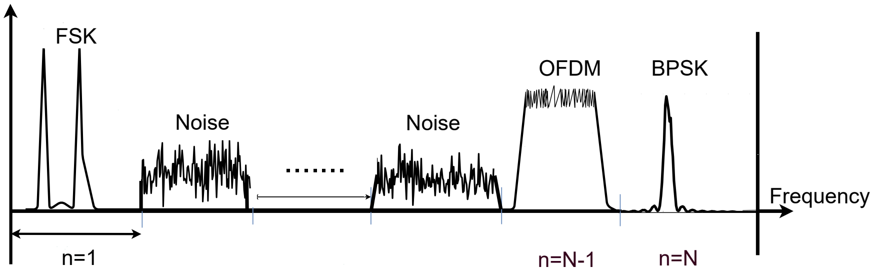


Figure 2.1: Signal model

version of the originally transmitted individual user signal, which is given by:

$$s(t) = x(t) + w(t) \tag{2.1}$$

Where $w(t)$ is the stationary Gaussian white noise and $x(t)$ is an unknown modulation type to be determined. $x(t)$ depends on the modulation type and is given by:

$$x(t) = ae^{j(2\pi\,\Delta f_c\,t+\theta)} \sum_k s_k p(t - kT - t_0) \tag{2.2}$$

Where a is the amplitude factor, θ is the phase shift, Δf_c is the carrier frequency offset, $s_k = s_{kl} + js_{kQ}$ represents the symbol transmitted within the k^{th} symbol period, T is the symbol period, t_0 is the propagation delay and $p(t)$ is the overall signaling pulse.

The receiver monitors the activity of the spectrum spanning all the frequency range from f_{min} until f_{max}. The receiver terminal is sweeping over multiple equal bandwidth sub-bands (SBs), and each of these SBs may be occupied by narrowband signal or not. Our goals are to estimate the spectrum occupancy and to reveal the modulations of individual signal components over the entire WB $[f_{min}, f_{max}]$.

Chapter 3: SCF Based Classification: A Convolutional Neural Network Approach

3.1 Proposed technqiue

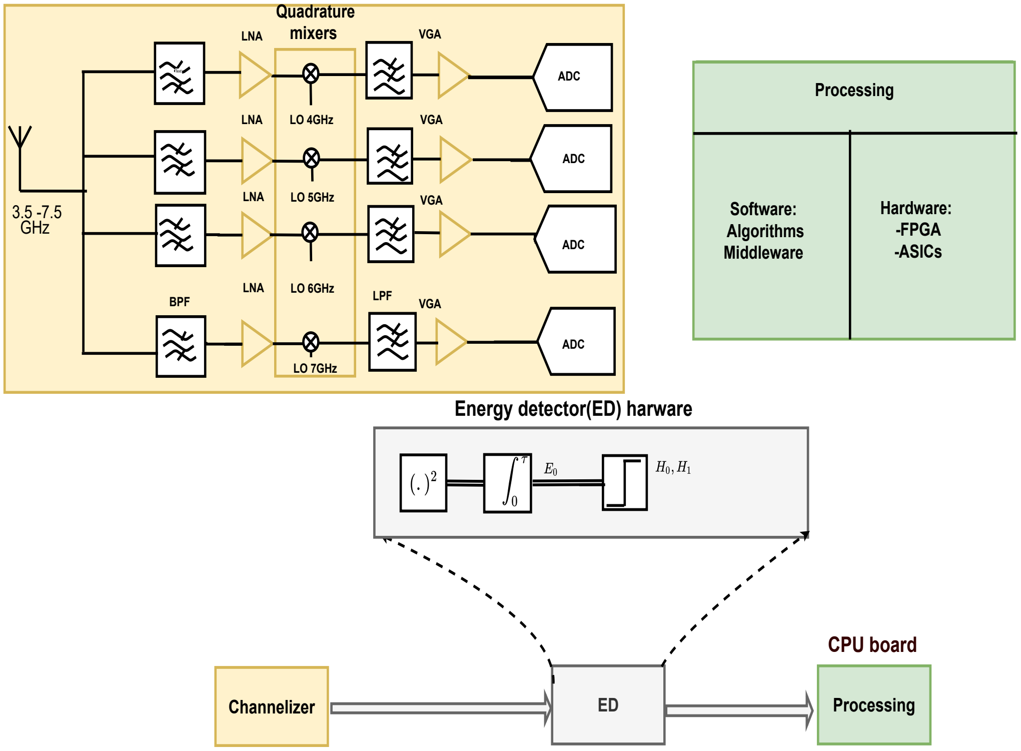


Figure 3.1: A channelized Receiver Architecture for WB Spectrum Modulation Classification

The proposed technique consists of 3 stages. First, a receiver front end, which uses an analog domain channelizer. The reason we suggest using channelization is mainly to overcome the ADC speed limits or circuit limitations. Also, to reduce the processing time in the digital domain. Let's take, for example, a receiver operating in the $[f_{min}, f_{max}]$ frequency band. In this case, after channelization, the band is split into several sub-channels, and we process each sub-channel independently. For instance, as highlighted in Figure 3.1, each sub-channel has its own mixers, band-pass filters, and ADCs. After down-conversion to baseband, we can sample each sub-channel at 1 GHz. We explain the channelization stage in more detail in the next Subsection. The second stage consists of an energy detector (ED) using energy to detect the presence of any signal based on a simple noise floor estimation for each sub-channel where the energy is compared to a threshold, as highlighted in Figure 3.1. Therefore, each sub-channel where a signal is actively present is subsequently processed separately in the digital domain.

Finally, a CPU board that uses the samples from the first and second stages to obtain the features as described in Figure 3.1. Furthermore, we use statistical cyclostationarity features due to their resilience to noise and co-channel interference comparable to the in-phase and quadrature-phase components. We choose one of the most widely used features in the cyclostationary signal processing, the spectral correlation function (SCF), for its efficiency in representing the cyclostationarity of a given signal. We consider supervised learning as a classifier and a convolutional neural network (CNN), in particular as one of the most popular and powerful deep learning algorithms specically designed to work with 2D structure inputs. In this section, the first and second stages are discussed in Subsection 3.2, the theory behind the SCF is explained in Subsection 3.3, and best-supervised learning technique is highlighted in Subsection 3.4. We also evaluate the performance under different propagation distortions that may affect the classification performance and end up finding the best features to use in classification based on the analysis of performance in terms of complexity and accuracy in Subsection 4.

3.2 Channelization and Energy Based Detection

One of the challenges of working with the WB RF spectrum, for instance, $3.5 - 7.5$ GHz range, is the need for high sampling required by ADC's to successfully recover the high rate WB signal. Also, in many cases, signals do not fully cover the whole spectrum; therefore, applying the classifier on the whole spectrum would result in an unnecessary increase in computational power. To overcome the aforementioned challenges, we use a channelization stage to split the WB input into non-overlapping narrowband sub-channels and only choose the sub-channels that contain a certain energy level. The goal is to reduce computation costs by estimating the SCF for only a few sets of the sub-channels. Let the input signal denoted by $x(t)$ which has bandwidth $B = |f_{max} - f_{min}|$ and is sampled at f_s Hz. The number of sub-channels in general could be more than four and is denoted in this case by N_t where each sub-channel has bandwidth $B_t = \frac{B}{N_t}$. In other terms, the band is downsampled by N_t Hz.

After the channelizer produces a set of parallel narrowband sub-channels, an energy detection stage as highlighted in Figure 3.1 is applied to each sub-channel. The square of the samples are summed. After that the energy for each sub-channel

E_0 is compared to certain threshold E_T. If the energy is more than the threshold E_T, then it is assumed that the signal is present. If the energy is below E_T it is assumed that signal is absent. Which makes the decision for the energy detection according to two hypothesis null (H_0) and alternative (H_1) expressed as following:

$$\begin{aligned} &(H_0) : s(t) = 0 && \text{if} \quad E_0 < E_T \\ &(H_1) : s(t) = h.x(t) && \text{if } E_0 \geq E_T \end{aligned}$$

where h is the channel amplitude gain.

In our framework, a frequency channelized receiver structure is chosen to implement the channelization stage where the channels are selected at the analog stage. We use a 4 sub-channels example here to make the visualization in Figure 3.1 clear to the reader. But a more interesting channelization problem is to handle bigger sets of channels. A digital domain channel selection can be implemented using a controller that tunes the analog front end to the desired operating frequency. After channelization, the spectral correlation function is estimated in each sub-channel separately, which we explain in the following Subsection.

3.3 Spectral Correlation Function (SCF)

The SCF can be seen as the Fourier transform of the cyclic autocorrelation function, which is the Fourier series of the autocorrelation function. The former is defined by the following:

$$R_x^\alpha(\tau) = \sum_{n=-\infty}^{\infty} x[n + \frac{\tau}{2}]x^*[n - \frac{\tau}{2}]e^{-i2\pi\alpha n} \tag{3.1}$$

Therefore, the SCF can be defined as:

$$S_x^\alpha(f) = \sum_{\tau=-\infty}^{\infty} R_x^\alpha(\tau)e^{-i2\pi f\tau}$$

where $x[n + \frac{\tau}{2}]$ and $x[n - \frac{\tau}{2}]$ are two random variables corresponding to two time instants of the random signal separated by the lagging variable τ while α represents the cycle frequencies (CF) of the signal.

We seek to leverage the second order cyclostationary SCF as our input features. The SCF has a better representation of the input data. In fact, the SCF can be harnessed to characterize the cyclostationarity of the dataset at a reasonable cost. The SCF function of a complex-valued stationary process $x(t)$ depends on estimation method: Time Smoothing Method (TSM), Frequency Smoothing Method, and the Strip Spectral Correlation Analyzer(SSCA). In theory [9], both the CAF and SCF require an infinite time/frequency averaging of the cyclic periodogram. However, for any of the estimation methods, they produce an estimate of the SCF over N samples. The TSM produces a power spectrum estimate by averaging M cyclic periodograms corresponding to the different blocks of a time series $X(t)$ over the time domain in order to reduce the variance of the cyclic periodogram. The TSM is very similar to Bartlett's estimation method for stationary signals except that this method averages the cyclic periodograms instead of the standard periodograms [10].

For further optimization and complexity reduction, the SCF estimate of the signal can be calculated using less number of samples by the applying channelization on each channel. Channelization (tunneling) can reduce the size of the input and compute an estimate SCF by exploiting only the tunnels that contain the highest energy of the signal and ignoring tunnels with lower energy below a threshold. This technique is a part of a tunneling framework described in [11]. The SCF function $S_x^\alpha(f)$ is typically sparse and has non-zero values only at the Cycle Frequencies (CFs), which reduces the computation time of the SCF calculation. The choice of SCF as our input feature for the classifier is driven by first, the periodicity of the correlation between the spectral components of $x(t)$ for different modulations such as BPSK, FSK, QPSK, ASK, and AM that exhibit different CF's and that can be used to discriminate between different modulations. Second, it is also noticeable that the spectral correlation of noise is zero since white noise is stationary and thus exhibits no spectral correlation. Third, we can use the SCF function with all α as our features to classify different modulation classes. However, this will result in a huge amount of computation due to the size of the matrix and how it varies with the number of cycle frequencies. Instead, fixing the number of cycle frequencies α can reduce further the input by considering a handful of cycle frequencies.

3.3.1 Time Smoothing Method

Several estimators can be used to estimate the SCF of a given signal: Time Smoothing method (TSM), Frequency smoothing method (FSM), and Strip Spectral Correlation Analyzer (SSCA). The TSM is used instead of both FSM, SSCA because FSM requires a Fourier Transform with a window equal to the size of one block of data. Obviously, the TSM segments the provided data block into M contiguous blocks of N samples each, computes the cyclic periodogram for each block, and averages the results. Consider the discrete Fourier transform of some data x shifted by some amount of time u.

$$X(u, f) = \sum_{t=0}^{N-1} x(t+u)e^{-i2\pi ft} \tag{3.2}$$

The non-conjugate cyclic periodogram is a function of time offset u as well

$$I^{\alpha}(u, f) = \frac{1}{N}X(u, f+\frac{\alpha}{2})X^{*}(u, f-\frac{\alpha}{2}) \tag{3.3}$$

The TSM estimate of the SCF simply averages M non-conjugate cyclic periodograms over time. The FFT is used to compute each non-conjugate cyclic periodogram in practice and because the FFTs are averaged over time, their relative phases are not taken into account. Thus, the TSM requires a phase compensation factor which is easily accounted for by multiplying each cyclic periodagram by $e^{-i2\pi\alpha D}$ where D represents the starting point or left edge of the sub-block [10].

The final TSM expression is

$$\hat{S_x^\alpha}(f) = \frac{1}{M}\sum_{j=0}^{M-1} \tilde{I^\alpha}(jN, f)e^{-i2\pi\alpha jN} \tag{3.4}$$

where $\tilde{I^\alpha}(jN, f)$ is the non-conjugate cyclic periodogram of the j^{th} block of N samples.

One of the main advantages of the TSM that makes it very appealing for hardware implementations for all non-blind automatic modulation classification is that the FFT length needed for the operation is independent of the data length. Unlike the Strip Spectral Correlation Analyzer method, for non-blind operations, the TSM has a low computation cost which is expressed in the formula [11]:

$$C_{TSM} = N\log_2(M) + 2N \tag{3.5}$$

where M is the number of blocks, and N is the number of samples in each block. Hence, TSM is the most suitable SCF estimation method for hardware implementation.

3.3.2 Implementing The TSM

From the implementation's point of view, to find the estimate of SCF, the input signal $X[n]$ is divided into M evenly split blocks with N samples each. Then, the cyclic periodogram is calculated for each block by applying the conjugate multiplication of the Fourier transform output with its conjugate shifted by the cycle frequency α. This can be illustrated by the following equation:

$$I^{\alpha}(jN, f) = \frac{1}{N} X(jN, f + \frac{\alpha}{2}) X^{*}(jN, f - \frac{\alpha}{2}) \tag{3.6}$$

where $I^{\alpha}(jN, f)$ is the cyclic periodogram of that block, and jN indicates the amount of time shift for a block j. In order to take into consideration the temporal start of each block, each block is shifted by a phase compensation factor, equals to $j\alpha N$. Finally, a smoothing kernel is applying that and averages the cyclic periodogram of all the M blocks over time, as shown in the following equation:

$$S_{x}^{\alpha}(f) = \frac{1}{M} \sum_{j=0}^{M-1} \widetilde{I}^{\alpha}(jN, f) e^{-i2\pi\alpha jN} \tag{3.7}$$

Then, these operations are repeated for each cycle frequency. This algorithm outputs a KxF matrix, where K is the number of cycle frequencies, and F is the FFT resolution. Each row of this matrix represents the spectral correlation of the cyclostationary signal at a cycle frequency of α.

3.4 Supervised Learning Model For Classification

3.4.1 Classifier Architecture

We investigate the effectiveness of the supervised algorithms for automatic modulation recognitionrecent work considered using a convolutional neural network. Convolutional neural networks are based on convolution operation with a given

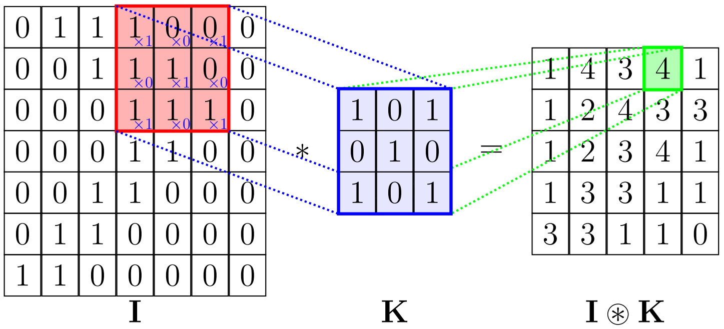


Figure 3.2: Convolution operation between input I and a (3x3) filter K

filter.

This is useful for applications such as image processing and computer vision.

Mainly there are two advantages of using convolutional networks over fully connected layers, first parameter sharing, and second sparsity of connections.

- **Parameter sharing**: A feature detector that's useful in one part of the data is probably useful in another part of the data. For example, if you have a $3x3$ filter and a feature detector such as vertical edge detector, that is useful in one part of the image is probably useful in another part of the image. Let's say you figure out a $3x3$ filter, then, you can apply that filter multiple times on the input on different positions, which results in output $I * K$.

- **Sparsity of connections**: In each layer, each output value depends only on a small number of inputs. As an example in Figure3.2 the output$(1, 1)$ of the $I * K$ matrix depends only on the convolution of the filter K with the first $3x3$ input matrix. The other inputs do not affect output$(1, 1)$ at all.

We start with a convolutional neural network similar to the CNN2 network proposed in [7]. The architecture we use, consists of two 2D Convolutional layers (conv2D) with 8 and 16 features maps respectively and two Batch Normalization layers. Finally, we add one fully connected layer (dense layer) and the output of the last fully-connected layer is fed to an 11-way softmax which produces a distribution over the 11 class labels. Further, the model used is inspired from models in [7] and [12], with a slight difference. ReLU activation function is used and adjustment has been made to reduce overfitting. The full architecture is drawn in Table 3.1.

1	Input Layer	6x64x1
2	Convolutional	8 1x8 convolutions
3	Batch normalization	Batch normalization
4	ReLU	ReLU
5	Convolutional	16 1x8 convolutions
6	Batch normalization	Batch normalization
7	ReLU	ReLU
8	Dense Layer	11 x 1
9	Softmax	Softmax
10	Classification Output	11

Table 3.1: Architecture of the CNN model

3.4.2 Classifier Training and Parameters

We also use Stochastic gradient for optimization and categorical cross entropy [13] to compute the loss [14]. We also build our model as defined, and we run the model after using MATLAB© and deep learning toolbox seamlessly on GPU

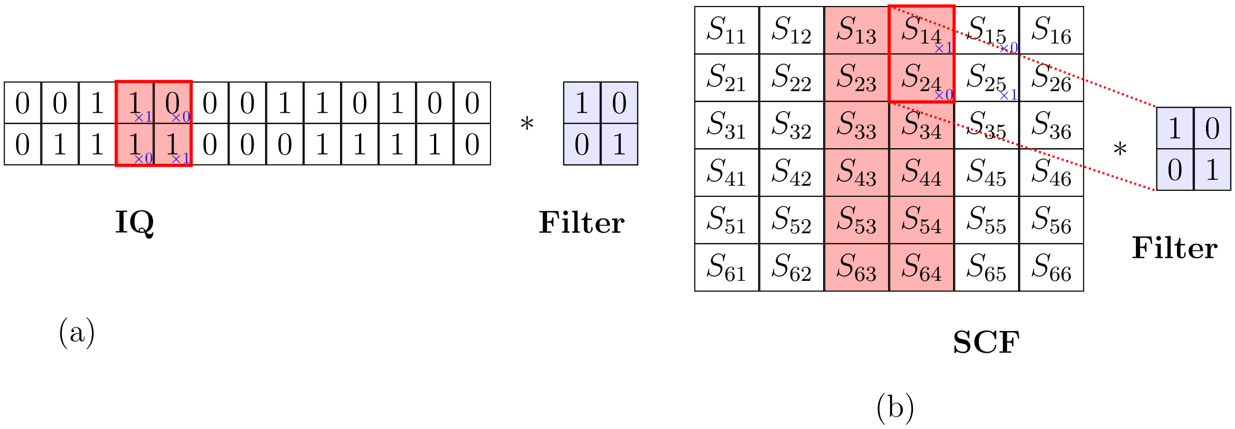

0 0 1 1 0 0 0 1 1 0 1 0 0
0 1 1 1 1 0 0 0 1 1 1 1 0
×1 ×0 ×0 ×1
*
1 0
0 1
IQ
Filter
(a)
S_{11} S_{12} S_{13} S_{14} S_{15} S_{16}
S_{21} S_{22} S_{23} S_{24} S_{25} S_{26}
S_{31} S_{32} S_{33} S_{34} S_{35} S_{36}
S_{41} S_{42} S_{43} S_{44} S_{45} S_{46}
S_{51} S_{52} S_{53} S_{54} S_{55} S_{56}
S_{61} S_{62} S_{63} S_{64} S_{65} S_{66}
×1 ×0 ×0 ×1
*
1 0
0 1
SCF
Filter
(b)

Figure 3.3: (a)Convolution between IQ and (2x2) kernel(filter) (b)Convolution between SCF and (2x2) kernel(filter)

server with NVIDIA GTX 1080 GPU and 16GB RAM. This paper considers two representations of the input signal. The first representation using In-Phase and Quadrature components (I/Q), while the second using SCF. If the IQ's are fed to the model as in Figure 3.3,(a) the input layer consists of a 2xN real data vector corresponding to the I/Q values. The first row corresponds to the in-phase component, and the second row holds the quadrature component. Specific to our problem, the input vector 2xN real data vector is fed to the model in frames of size 400 samples. If the SCF is fed to the model as in Figure 3.3,(b) the input layer consists of KxM matrix where K is the number of cycle frequencies of the spectral correlation function, and M is the size/number of samples in one block.

Chapter 4: Performance evaluation and analysis

Our evaluation aims to evaluate the performance of supervised learning using convolutional neural networks with respect to additive white noise, multipath channel. Both these channels can be used to model the transmission of signals over a more complex environment. AWGN channel is often used as a model in which the only impairment to communication is a linear addition of WB noise with a constant spectral density. Rayleigh fading channel, however, is a model used to describe multipath transmission in heavily built-up urban environments where we have scattering and no dominant line of sight (LOS). Hence, we want to see and test the performance of the CNN model with both IQ and SCF as input features under these different environments to validate the results. We train and test our model with both IQ and SCF as input, to see which one has better performance. We start in the next Subsubsection by presenting the dataset generation and simulation setup.

4.1 Dataset and Simulation

We use the dataset presented in [8] by T. O'Shea. The dataset contains a training set and test set, and we have 220000 examples. Since the number of samples in the aforementioned dataset, which is 128, is not enough to use for our scheme, a new dataset is generated by building upon the functions in MATLAB© communication toolbox and deep learning toolbox. Eleven modulations are generated, and we consider mainly two variety of wireless channels. Firstly, additive white gaussian noise channel and secondly Rayleigh fading channel with 4 paths.

We consider multiple short time frames (observations windows) with a size equal to 400 samples, and each frame is passed through a raised cosine FIR pulse shaping filter with a roll-off factor of $alpha = 0.25$.

We create multiple datasets to test the performance:

- $D0$, without impairments. Initially, the generated dataset $D0$ consists of 11000 frames without any channel impairment added.

- $D1$, with AWGN noise. We use AWGN noise with SNR values (in dB): $-20, -15, -10, -5, 0, 5, 10, 15, 20, 25$, to create 11000 frames.

- $D2$, with variable multipath delay impairment. We use a Rayleigh multipath fading channel with the sum of the 4 path delays values picked according to the first row of Table 4.1 and the sum of path gains fixed to 0 dB. The second dataset generated contains 11000 frames.

- $D3$, with variable multipath gain impairment. Additionally, the same Rayleigh multipath channel is used as in $D2$, where at this time Rayleigh multipath fading is added with fixed sum delays equal to $75e^{-2}\mu s$ and the sum of 4 path gains picked according to the second row of Table 4.1 to create the third dataset $D3$. The third dataset generated contains 11000 frames. We use 4 paths Rayleigh model, and the variation in path delays and path gains of all four paths is illustrated in the description Table 4.1.

sum of path delay (μs)	75e2	75e1	75e0	75e-1	75e-2
sum of path gains (dB)	0	-1.5	-3	-10	-25

Table 4.1: Variation in sum path delays and sum path gain

We use all these datasets as a benchmark for comparison between the proposed SCF and the IQ based classification. The specific modulations considered within each of these four dataset types are as follows:

- BPSK, QPSK, 8-PSK, 16-QAM, 64-QAM, PAM4, GFSK, CPFSK, B-FM, DSB-AM, SSB-AM

Overall, we have 11x1000x2x400 matrix with 11x1000 frame or observation windows total, with 1000 frames for each modulation. Each frame in one of the three datasets has 400 **IQ samples** complex values. To reiterate the findings from a large amount of data in hand, **10%** of the data is reserved for validation, **10%** is reserved for test and the remaining **80%** is reserved for training.

We use accuracy to quantify the performance of the AMC model proposed. We plot the accuracy curve and the confusion matrices for each modulation. Accuracy is one metric for evaluating classification models. Formally, accuracy is defined as the ratio of the number of correct predictions by the number of predictions:

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Number of predictions}}$$

It is actually inversely related to the loss, as:

$$Accuracy = 1 - 1/m \sum_{i=1}^{m} l(\hat{y}_i, y_i)$$

where l is the loss for a single observation or modulation among the 11 classes of modulations (BPSK, QPSK, 8-PSK and others). To train our model and to find all the optimal weights values, we use a cross entropy loss function appropriate for multi-class classification defined as

$$l(\hat{y}_i, y_i) = -y_{\Theta,i} log(p_{\Theta,i})$$

where:

- y_i: binary indicator (0 or 1) if class label i is the correct classification for observation Θ.
- $p_{\Theta,i}$: estimate or probability that the observation Θ is of class i.
- Θ: the actual observation

4.2 Evaluation under Multi-Path Channel With Variable Path Delays Profile

We performed classification experiments on dataset $D2$ where a number of different path delays are added to the 11 modulated signals. Values of the sum path delays (in μs) are described in Table 4.1. Using CNN only, with number of layers following

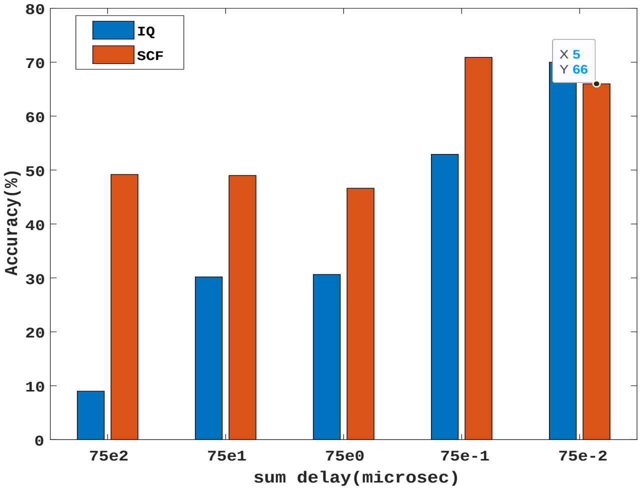


Figure 4.1: Performance in terms of test accuracy under variable path sum delays vector(fixed sum gain=0 dB)

the architecture described in Subsection 3.4 and summarized in Table 3.1, we show in Figure 4.1, that the classification performance of both SCF and IQ are impacted by the scattering delay dues to multipath propagation in the environment that it is likely to happen in urban cities.

The addition of the multipath fading does not impact the SCF+CNN model as much as it impacts the IQ+CNN model, where a much higher effect is seen. In fact, we can see from Figure 4.1, that for path sum delays equal or higher than 7.5 μs, SCF+CNN achieves almost $18\%, 16\%, 20\%, 40\%$ higher accuracy on average compared to IQ+CNN. It is shown from the Figure that the SCF based classifier performs better under severe fading, especially for values of path sum delays (in μs) equal to $7.5, 75, 750, 7500$. However, for IQ based classifier, the classification performance gets worse as we go for a more extreme and higher value of path sum delays. For instance, at 7500 μs path sum delays, the accuracy of IQ based classifier reaches below 10%. Meanwhile, the SCF based classifier has almost 50% accuracy. This shows how SCF and IQ based models will most likely act in an environment where the receiver is under such impairments.

4.3 Evaluation Under Multi-Path Channel With Variable Path Gains Profile

We performed classification experiments on dataset $D3$, where this time, a set of different multipath sum gains are added to the 11 modulated signals. Values of set (in dB) are described in Table 4.1. Using CNN only, with number of layers following the architecture described in Subsection 3.4 and summarized in Table 3.1, we show in Figure 4.2, the classification performance of both SCF, and IQ under variable sum path gains (in dB). Both SCF and IQ are impacted by the addition of fading dues to multipath propagation and variable path sum gains. The addition of the multipath fading does not impact the SCF+CNN model as much as it impacts the IQ+CNN model, where much bigger effect is seen. In fact, we can see from Figure 4.2, that for sum gain (in dB) under -1.5 dB, and specifically equal to $-25, -10, -3, -1.5$dB where SCF+CNN achieves $41\%, 1\%, 12.5\%, 1\%$ higher accuracy on average compared to IQ+CNN.

It is shown from the Figure that the SCF based classifier performs better under severe fading, especially for values of path sum gains (in dB) equal to $-25, -10, -3, -1.5$ dB. However, for IQ based classifier, the classification per-

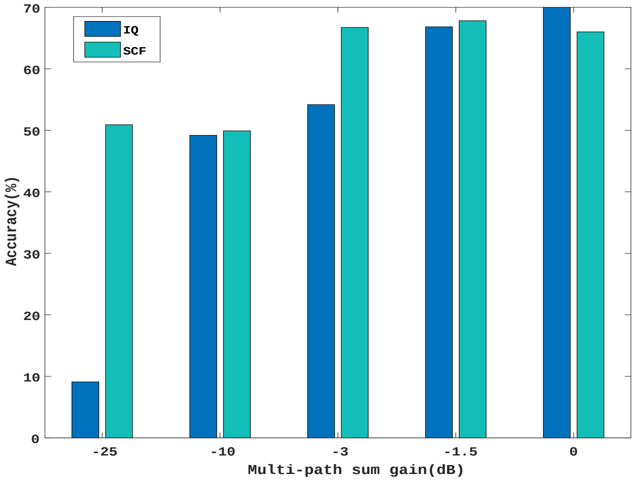
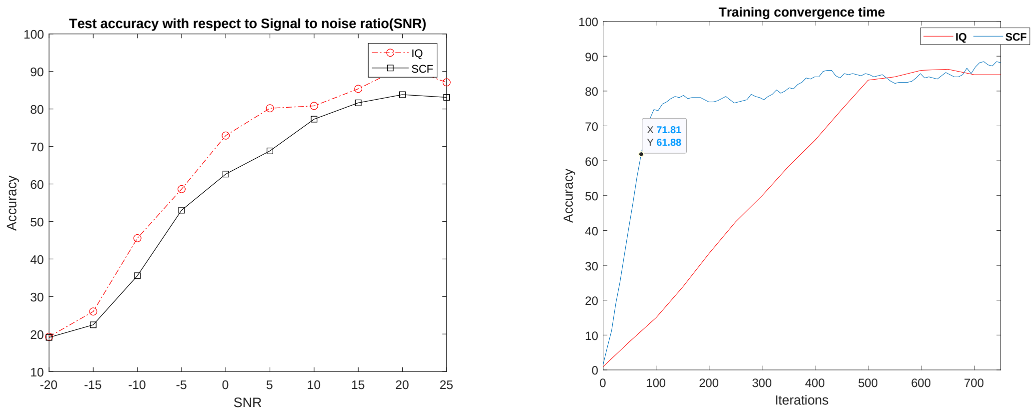


Figure 4.2: Performance in terms of test accuracy vs multi path sum gain vector

formance gets worse as we go for low values of the sum gain (in dB). For instance, at path sum delays equal to -25 dB, the accuracy of SCF based classifier is 5x more than the IQ based classifier, with accuracy more than 50%. This shows how SCF and IQ based models will most likely act in an environment where the receiver is under such impairments.

4.4 Evaluation Under Additive White Noise Channel Distortion

We perform classification on dataset $D1$, which is described previously in Subsection 4.1. Our goal is to see the robustness of both techniques under additive noise. Additive white noise is a linear addition of noise with a constant power spectral density. Typically, it is likely to observe such noise in WB spectrum, and we expect



(a)

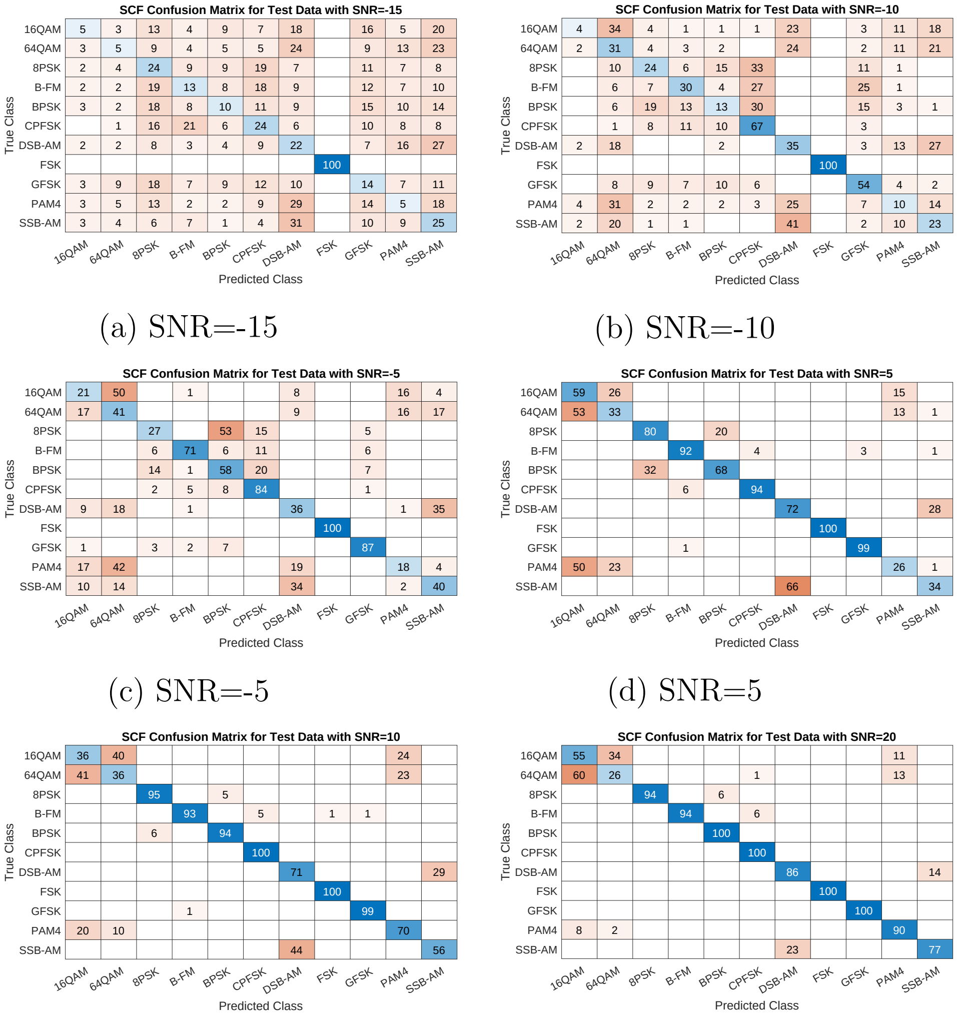
(b)

Figure 4.3: (a) Comparison between SCF+CNN and IQ+CNN under additive white noisy channel (b) Training convergence time for SCF+CNN as compared with IQ+CNN

that SCF would perform better than IQ since stationary noise exhibits no spectral correlation. Hence, it would not add any effect to the classification performance of the SCF function.

We perform classification on dataset $D1$ where we try different signal-to-noise ratio (SNR) equal to $-20, -15, -10, -5, 0, 5, 10, 15, 20, 25$ for different levels of additive noise. Using CNN only, we show in figure 4.3a, both results for CNN+IQ, CNN+SCF after training a convolutionnal neural networks following the architecture as in Subsection 3.4 and summarized in Table 3.1. The test accuracy for both model is shown, in Figure 4.3a.

It is shown from Figure 4.3a that with SCF+CNN, we get performance in terms of test accuracy quite close to the IQ+CNN. In fact, SCF+CNN achieves a test accuracy of 83.09% compared to 87.09% for IQ+CNN for SNR value equal to 25 dB. For values less than 5 dB until -20 dB, the gap between both models lessen, and at -20 dB both have the same accuracy. While IQ+CNN is better in terms of test accuracy for high SNR values, on the other hand, for lower SNR values, the SCF+CNN reaches almost the same performance in terms of test accuracy. Although the test accuracy is little less than IQ+CNN model, the convergence time is much better, as we can see from Figure 4.3b. In fact, we observe from Figure 4.3b, that CNN+SCF takes only 100 iterations to converge, although CNN+IQ takes around 500 with the same CNN architecture. Figure 4.4, highlights the individual performance with respect to each modulation under the additive white gaussian noisy channel for SCF+CNN model with different SNR values in $-15, -10, -5, 5, 10, 20$ dB. This Figure excludes IQ+CNN model. The model, in this case, does not perform well under additive white noise, especially for lower signal-to-noise ratio. Nevertheless, we expect the cause is due to the low resolution of the spectral correlation. This suggests that using higher resolution in both frequency and cyclic frequencies would definitely increase the test accuracy. However, this can increase the training and convergence time due to larger size. For the above reason, a good choice of resolution and size of the input is important so that we don't compromise much of the accuracy, training time. We can see from Figure 4.4, that as the signal-to-noise ratio rises, the classifier is able to classify better each modulation among the 11 modulations. We note that our goal is to justify the use of SCF with convolutional neural networks for improving the performance of the classification. Therefore, our focus is more on proving that SCF+CNN is more convenient than CNN+IQ with certain channel conditions,



(a) SNR=-15

(b) SNR=-10

(c) SNR=-5

(d) SNR=5

(e) SNR=10

(f) SNR=20

Figure 4.4: SCF confusion matrix under AWGN channel with different SNR (dB).

and therefore we did not try different other architectures such as AlexNet. Indeed, we think that trying different architectures and hyper-parameters can improve the performance further. This, however, requires a lot more training and computation time. Hence, we leave that for a future research direction, and for now, the next

Subsection, evaluation under compressed learning only focus more on improving state of the art proposed CNN by using compressed measurements and compressed learning.

Chapter 5: Enhanced SCF with Compressed learning

In previous Subsection, we provide results that SCF based classification outperforms IQ under severe fading conditions. In this Section, the proposed technique is enhanced for hardware implementation. As mentioned earlier in Subsection 4, we have to choose carefully the resolution of SCF. The reason is that both accuracy and training time depend on the SCF input size. In fact, the selection of high frequency resolution, and high cycle frequency resolution will increase the accuracy but at the expense of training time. On a similar manner, low frequency, and low cycle frequency resolution will reduce the accuracy. For this reason, our aim in this Subsection is to improve the classification/inference task by reducing the cost of learning in the data domain perceptibly while preserving almost the accuracy of classification. A general strategy for making this possible is called compressed learning. Compressed learning provides a projection from the data domain to a measurement domain that preserves the linear separability under certain conditions [15].

Classification in the measurement domain is possible after transforming the data to some appropriate compressed domain provided that the linear projection using a matrix ϕ preserves the structure of the instance space.

Being able to learn in the compressed domain is beneficial both in many applications. Examples include image recognition, natural language processing, sensor

Sensing Rate	Random Sensing + CNN [16]	Percent difference	Compressed Conv-Net	Percent difference
1	72.125	-	72.125	-
0.5	70.5	1.625	70.375	1.75
0.1	68.125	4	67.75	4
0.05	66.125	6	68.125	4
0.02	61.25	10.875	65.5	6.625
0.01	47	25.125	57.25	14.875

Table 5.1: classification accuracy(in %) for learning with compressed SCF vs sensing rates SR=M/N(averaged over 800 test SCF)

networks, wireless communication, and automatic modulation recognition. All these applications share the need to have an efficient data acquisition and need to compress the data in the data domain. For instance, in automatic modulation recognition, reconstruction of a signal(FM, GSM, Radar) in WB from the sensor data could be expensive and not necessary. Rather one wishes to identify a function of the signal that, for example, indicates whether the signal is consistent with a target signature. Compressed learning can be a solution to reduce the cost of high dimensionality especially in WB spectrum and reduce the computational complexity of the SCF during training time. For this reason, many have proposed compressed learning.

Calderbank et.al(2009) [15] is the first to show that learning and classification are possible in the compressed domain $y = \phi x$ and that the performance of SVM's classifier in the compressed domain is close enough to that of the best linear threshold classifier in the signal domain. Davenport et.al(2007) addresses the problem of compressed learning [17], using the idea of smashed filter, based on a lemma proposed by Johnson-Lindenstrauss [18] about dimension reduction of finite point sets in Hilbert space, show that for sufficiently large M, all pairwise distances between points on a manifold are well-preserved under projection to a random M-dimensional subspace. The paper propose a generalized maximum-likelihood classifier to be employed for target detection in the measurement(compressed) domain. A deep learning approach was introduced [19] [16] in which a convolutional neural networks that operates in the sensing domain used to optimize the sensing matrix. Others, have tried to proposed a theoretical guarantees for achievable accuracy in the compressed domain for both sparse and non-sparse domains [20] or solve the problem using classification on compressed data [21] [20] [16] [22] [23]. In the following, we leverage the same technique called compressive learning to perform classification directly on sensing domain for our SCF based automatic modulation classification instead of the signal domain. The aim of the proposed compressed Conv-Net is to improve the performance of the SCF based classification proposed in Subsection 4.

5.1 Proposed Compressed SCF: Compressed Conv-Net

First, we design an optimal sensing matrix to be used instead of random sensing. We propose a deep learning solution for CL, which jointly optimizes the sensing matrix and the inference operator. Our choice is motivated by the work of [19] which employs convolutional networks for the task of compressive classification on the MNIST dataset. Our approach, uses two dense layers to find the optimal sensing matrix ϕ, based on a sparse structure similar to the one highlighted in Figure 5.1. The network is composed of an input layer with N nodes, a compressed sensing fully-connected layer with $N * SR$ nodes, $SR << 1$ (its weights form the sensing matrix) (3) ReLU activation units (4) a fully-connected layer that expands the output of the sensing layer to the original image dimensions N (5) ReLU activation units. The input size is the same size as the SCF size, and the two fully connected layers are trained to optimize the sensing matrix ϕ, together with the CNN to find the inference operator. Again, we use the same CNN architecture presented in Subsection 3.3. The proposed Compressed Conv-Net framework is composed of : (1) Sensing stage, (2) Inference stage. The first stage (1), uses two subsequently fully connected layers to find the optimal sensing matrix ϕ. Meanwhile, the second stage (2), reshape the input size and uses two 2D convolutional layers(conv2D) with 8 and 16 features maps respectively, two Batch Normalization layers and one fully connected layer with Softmax activation at the output layer. This is the same architecture used in Subsection 3.4.1 to perform non-linear inference. The input features are the same SCF generated in Subsection 4, only in our Compressed Conv-Net framework we use compressed measurements, and the feature extracted from compressively sensed signals based on the sensing matrix ϕ.

For training, the two networks are jointly optimized to find the optimal sensing matrix ϕ and optimal weights. The sensing matrix ϕ and its transpose ϕ^T are jointly trained from the spectral correlation data. For inference, the input layer takes as input the original SCF and project it to a measurement domain $y = \phi x$. The second layer takes y and represents the back-projection of y to the original space to produce the same input SCF at the output. The architecture that is used is presented in Figure 5.1. The second stage, the inference stage, takes the output of the sensing stage as input to a convolutional neural network to realize some non linear inference task for automatic modulation classification.

In summary, we have two stages the first stage called sensing where we use the optimal sensing matrix ϕ to project the SCF into a measurement domain with low dimension. Once we have the output from the sensing stage, the inference stage kicks off, using the output of the first stage $y = \phi x$ for direct inference.

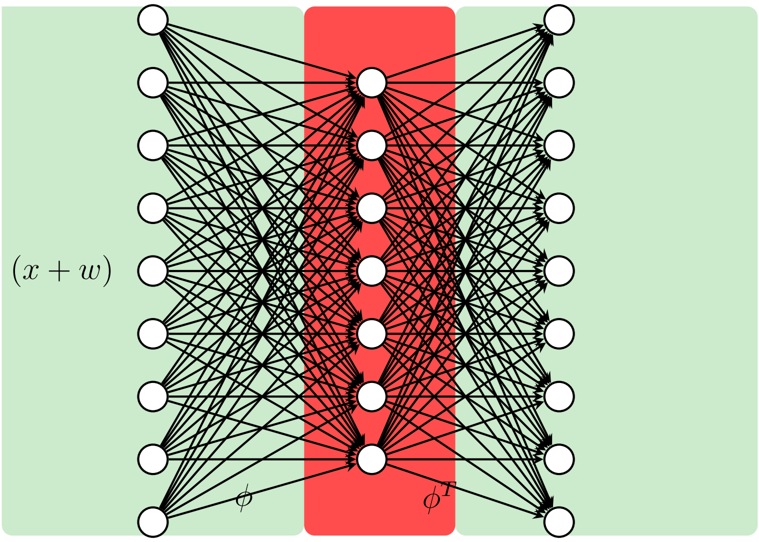


Figure 5.1: Sparse encoding and decoding

5.2 Performance Evaluation

We provide the performance evaluation of the proposed compressed Conv-Net compared to random sensing [16](random ortho-normal MxN matrix).

5.2.1 Simulation setup

We follow the same setup as in Subsection 4 except that we only use a noisy channel, with signal-to-noise ratio equal to 30dB to evaluate the performance of our proposed compressed Conv-Net scheme. A total of 11x1000 frames or observation windows total, with 1000 frames for each modulation. To reiterate the findings from the large amount of data in hand, **10%** of the data is reserved for validation, **10%** is reserved for test and the remaining **80%** is reserved for training.

We train the proposed network using the spectral correlation function and we try to fine tune all parameters by looking at best learning rate, weight decay and number of epochs. For best results we train our network using stochastic gradient descent with a variety of learning rates. For each training cycle, we fix the number of epochs as 17 based on the size of the dataset and the training takes around 3 minutes for one training cycle. Increasing the number of epochs to 500 makes the training 20-30minutes. The hardware we use for the training is GPU NVIDEA GeForce GTX 1050. The learning rate is 0.002 and we train the proposed networks and test on the same dataset with AWGN noise presented in Section 4.

5.2.2 Results

The performance evaluation of the proposed Compressed Conv-Net vs random sensing matrix followed by convolutional network is presented in Table 5.1. We define the Accuracy loss after compression as the difference in accuracy before compression and after compression. The performance results that are obtained with the aforementioned techniques in terms of accuracy for sensing rates SR in the range of $SR = 0.01$ to $SR = 0.5$ are presented in the table 5.1.

Classification accuracy results, reveal an advantage of the Compression using deep learning to find the optimal sensing matrix. In fact, the accuracy of Compressed Conv-Net is much better compared to random sensing for $SR < 0.1$. At lower sensing rates $SR < 0.1$, the accuracy of Compressed Conv-Net is not much degraded.

Chapter 6: Discussion and future extension

6.1 background on cumulants

The goal from this Section is to provide a brief study of higher order cyclostationarity, mainly high order features that can be used for statistical characterization of signals of interests and modulations. Even though, we can talk about moment and cumulants, mainly in this chapter we focus about the n^{th}order cumulants. Overall goal is to improve the automatic modulation classification further and make it more robust to interference and multipath. The most relevant work related to automatic modulation classification, using cumulants and cyclic cumulants is briefly reviewed in the sequel.

OA Dobre [24] employed fourth, sixth and eight orders cyclic cumulants(CC) respectively for classification between ASK, PSK and QAM modulations and analysis of cyclic cumulants of the basband signal at the receiver is performed and used as feature selection. A detailed function of the n^{th}order cyclic cumulants(CCs), cycle frequencies(CFs) and cycle spectrum of the baseband signal at the receiver are derived. In another work, OA Dobre [25] propose an algorithm based on higher-order cyclic cumulants for the automatic recognition of QAM signals. Swami and Sadler [26] proposed a hierarchical classifier, that shows to be effective particularity when used in decision tree, enabling separation into sub-classes at very low signal-to-noised ratio. Using fourth-order cumulants, higher order cumulants are shown to be robust when employed to discriminate ASK, PSK and QAM signals in the presence of carrier phase and frequency offsets.

Gardner and Spooner [27] [28] provided a cyclic cumulants foundation and introduced higher order moments and cumulants of narrowband spectral components of time-series. Furthermore, a comprehensive historical survey that traces the development of the ideas of temporal and spectral cumulants is introduced.

In this Section, we discuss the performance of cumulants based features to discriminate between 16-QAM 64-QAM QPSK family of modulation. The reason behind selecting this set of modulation is that SCF performs poorly with the later individual modulations. Theoretical results show that the spectral correlation function is the same for QPSK,QAM modulations for order two shown in Table 6.1. In fact, as it can be seen from Table 6.1, despite the three modulation scheme are different, they all have the same theoretical second order statistics. This shows that not all modulation schemes are distinguishable using the CAF, or the SCF. Thus, higher order statistics such as moments and cumulants are required to determine the type of modulation in use. In the following, we investigate the cyclic cumulants as part of an alternative for SCF. Given the signal, $x(t)$, the mathematical definition of the lag product is:

$$\prod_{j=1}^{n} x^{(*)}(t+\tau_j) \tag{6.1}$$

is simply the time delayed product of $x(t)$. Where notation (*) indicates the conjugation is optional. Different conjugation can produce different values, leading to multiple outcomes for the same order. An n^{th}-order statistic may have up to $\frac{n}{2}$ conjugations, and we can see some defined up to order 8 in Table 6.1 This way temporal moments are defined by taking the expected value of the lag product defined in Equation (6.1) which is expressed as:

$$R_x(t,\tau)_n = \mathbf{E}[L_x(t,\tau)_n]. \tag{6.2}$$

This equals to the n^{th}-order autocorrelation function, which when evaluated with n=2. Similarly, the Cyclic Temporal Moment Function generalizes the CAF. While moment can be used for classification, a much better alternative would be to use cumulants which are robust to noise especially when the radio scene has multiple signals.

The n^{th}order temporal cumulant can be expressed as a combination of products of lower-order temporal moment functions. The n^{th}order cyclic temporal cumulant function (cyclic cumulant) is just a Fourier coefficient of the n^{th}order temporal cumulant function.

The Temporal Cumulant Function can be expressed by a generalized Fourier series:

$$C_x(t,\tau;n,m) = \sum_{P_n}[C_x^{\beta}(\tau;n,m)e^{i2\pi\beta t}] \tag{6.3}$$

where $C_x^{\beta}(\tau;n,m)$ is the n^{th} cyclic temporal cumulant function, or just cyclic cumulant, n is the order, m is the number of conjugations, and β is the pure n^{th}-order cycle frequencies. The summation is performed over P_n, which is the set of all possible partitions of the set $1, 2, .., n$. The cyclic cumulants are pure sine-wave amplitudes, and they can be computed by combining cyclic-moment amplitudes which can be easily estimated from the signal data. Cumulants have a number of properties that make them attractive for constructing classification features. We show how cyclic cumulants can be used to discriminate between many of the modulations with same second order cyclostationarity features. This classes of modulation for example {16QAM, 64QAM, QPSK} have identical SOC. They are part of the digital QAM signals. The all have a similar spectral correlation function, so second order features would fails in this case to provide an information about the class of modulation of the received signal in case the signal is exposed to severe channel conditions. Cyclic cumulants have a number of properties that make them more suitable than moments for automatic modulation classification. Cyclic cumulants inherit all well-known properties of cumulants, such as tolerance to Gaussian contamination. If X is a set of jointly Gaussian variables, then the cumulant of X is zero for $n > 2$. Second most important cumulant property is the additive property, that is, the cumulant for the sum of N independent cyclostationary signals is the sum of the cumulants for the N individual signals. This property does not hold in general for the moments. Third property is selectivity, which implies that if two statistically independent time-series Y and Z defined as $X = \{X_j\}_{j=1}^{n} = \{x^{(*)j}(t + \tau_j)\}_{j=1}^{n}$ and $Y = \{Y_j\}_{j=1}^{n} = \{y^{(*)_j}(t + \tau_j)\}_{j=1}^{n}$, similarly for $Z = X + Y$. Then, if β_0 is nth-order cycle frequency for $y(t)$ but not for $x(t)$ Then $C_z^{\beta_0}(\tau; n, m) = C_y^{\beta_0}(\tau; n, m)$ Last but not least property is the delay property, if we have a delay signal of $x(t)$, $y(t)$ given by $y(t) = x(t - d)$. Then

$C_y^{\beta}(\tau; n, m) = C_x^{\beta}(\tau; n, m)e^{-i2\pi\beta d}$

It is shown previously [28] [29] that cumulants perform really well under non-ideal scenario(under interference and congested environment). In the following, we study their performance with respect to three modulation $\{$16QAM, 64QAM, QPSK$\}$. The scope of the next Subsection is to discuss the estimation of cyclic cumulants briefly and motivate the use of cyclic cumulantsx to discriminate between modulation of the same family.

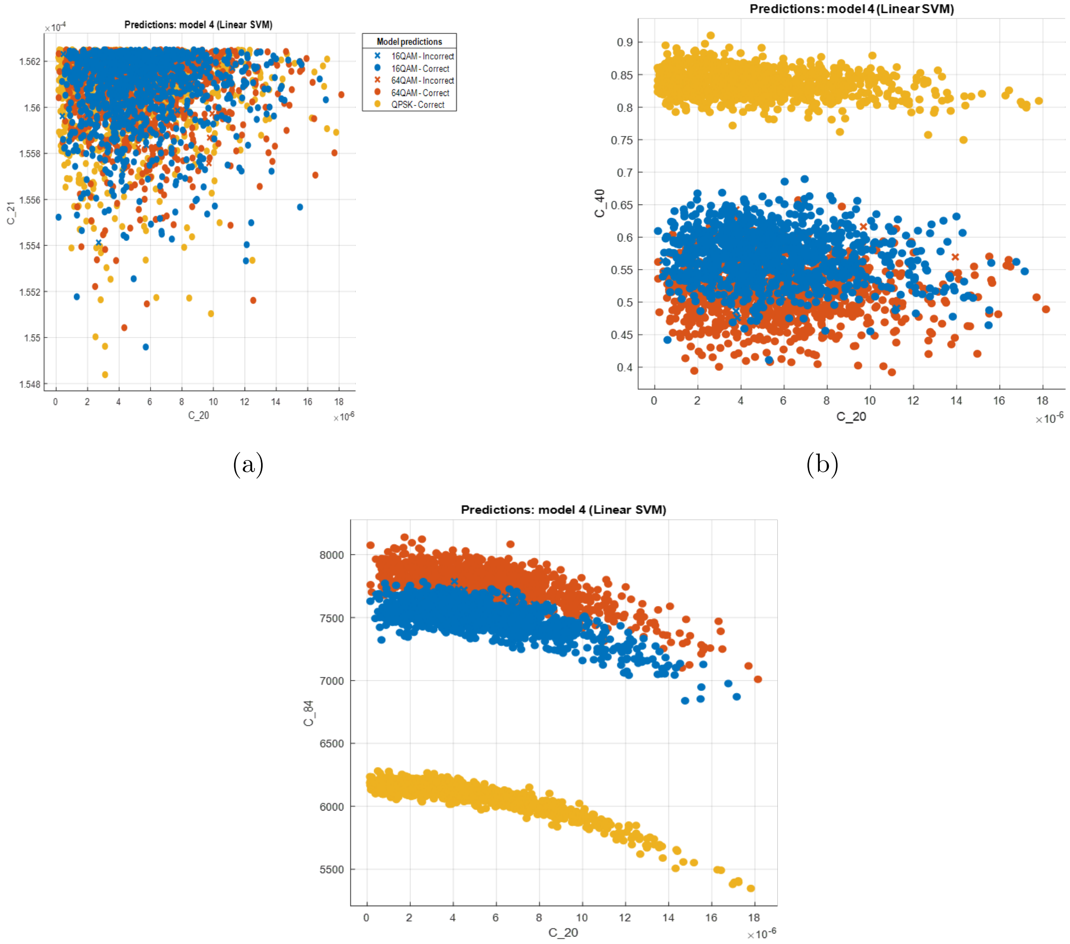
6.2 Proposed features

The problem of AMC is studied in this Section, with respect to three classes of modulation 16QAM,64QAM,QPSK as explained in previous Subsection, and consider the input data

$$x(t) = a(t)e^{i2\pi f_c t + i\phi_0}$$

where the complex-envelope signal a(t) is defined by defined by a complex valued PAM signal:

$$a(t) = \sum_{k=-\infty}^{+\infty} a_k p(t + kT_0 + t_0)$$



(a)

(b)

(c)

Figure 6.1: Cumulants prediction model: Linear SVM ((a) $C_{2,1}$ vs $C_{2,0}$; (b) $C_{4,0}$ vs $C_{2,0}$; (c) $C_{8,4}$ vs $C_{2,0}$)

The cyclic cumulant is given as a function of the n^{th}order cumulant according to the following:

$$C_x^{\gamma}(\tau; n, m) = \frac{C_{n,m}}{T_0} \int_{-\infty}^{+\infty} \prod_{j=1}^{n} p^{(*)_j}(t + \tau_j) e^{-i2\pi\beta t} dt$$
$$e^{i2\pi\beta t_0} e^{i(n-2m)\phi_0} \prod_{k=1}^{n} [e^{i2\pi f_c \tau_k}]^{(*)_k} \qquad (6.4)$$

for $\gamma = (n - 2m) f_c + \beta$, where $\beta = q/T_0$ and $C_{n,m}$ is the n^{th}-order cumulant of the symbol a_k using m conjugation.

Taking into consideration the additive Gaussian white noise w, the received signal can be modeled as following:

$$y(t) = x(t) + w(t)$$

where $w(t)$ is independent of $x(t)$

The n^{th}-order cumulant is defined by Equation (6.4) for digital QAM modulated signal. To estimate the n^{th}order cumulant, it is important to know the impure cycle frequencies of all signal of interests. Thus by knowing the cycle frequencies, estimation of the cumulants become easier, and less complex. However, in case we don't have prior knowledge of such information a procedure called blind cyclic cumulant estimation must be performed a priori. The estimation of the cyclic frequencies can be done from the data and it is specifically used in "blind" modulation classification which is out of scope of this report, thus we suppose we are given knowledge of one or more of the signal's modulation frequencies, such as a symbol rate or carrier frequency. In particular, the cycle frequencies for a digital QAM are given analytically by the formula: $(n-2m)f_c + k/T_0$ for n even. This was highlighted in [29]. We follow the theory in [28] to implement the estimation of the cyclic cumulant. The estimation of the cyclic cumulant, in this case, is based on cumulants. Cumulants estimates are based on the moments according to Shiryaev-Leonov Moment-Cumulant Formula or simply the Moment-Cumulant(M-C) formula.

6.3 Results

Using the M-C formula written for the n^{th}-order time-varying cumulants for $n = 2, 4, 8$, $m = 0, n/2$, at $\tau = 0$, we compute the cumulant at order $n = 2, 4, 8$. The n^{th}order cumulants obtained are taken as feature for QPSK, 16-QAM, 64-QAM classification. The magnitude of the cumulants versus the $C_{2,1}$ is shown in Figure 6.2 for $n = 2, 4, 8$. It is clear from Figure 6.2 that the higher the degree of the cyclic cumulant the better it is. In fact, using large n as in 8^{th} order, the different modulations belonging to the same family give rise to distinct features as the degree become larger. For example, 16-QAM and QPSK have identical cyclic autocorrelation functions ($C_{2,1}$) with reference to $C_{2,0}$ as it can be seen from Figure 6.2.(a). Meanwhile, have different fourth-order cumulant functions ($C4, 0$) with reference to $C_{2,0}$ as highlighted by 6.2.(b). Figure 6.2.(b) shows that even at n=4 16-QAM and 64-QAM still have the same cumulant but as we move to Figure 6.2.(c) feature $C_{8,4}$ have better separability. This demonstrate that Fourth-order

CCs are appropriate statistics to distinguish between the two classes QPSK and 16-QAM or QPSK and 64-QAM, while eighth-order CCs are suitable to discriminate signals within each class of the three QPSK,16-QAM,64-QAM. Even though, the separability increases for higher-order modulations with the degree n, we cannot keep increasing n to infinity because of the prohibitive burden increasing with it. Instead, we look at the lowest degree value that gives considerable separation between any modulation. Mostly this can be reached at 8^{th} degree or even less. For this reason we combine all possible cumulants with order $n \leq 8$ and use the mangitude to obtain such a vector $F = |C_{n,m}|, n = 2, 4, 8, m.$ the classifier used in this case is SVM and Figure 6.2 depicts the performance defined by the confusion matrix(CM) and the true positive rate under a non-ideal awgn channel.

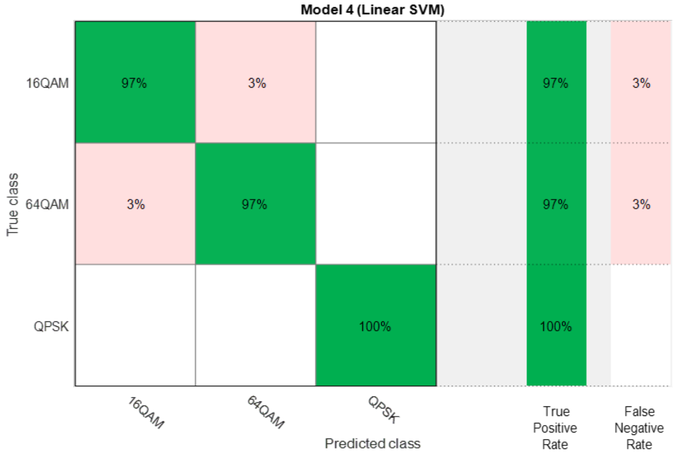


Figure 6.2: CM for predicition model using cumulants

Table 6.1: Theoretical higher order cumulant values for selected modulations schemes.

Order (n)	# of Conjugations	QPSK	16-QAM	64-QAM
2	0	0	0	0
2	1	1	1	1
4	0,4	1	-0.68	-0.619
4	2	-1	-0.68	-0.619
6	1,5	-4	2.08	1.7972
6	3	4	2.08	1.7972
8	0 ,4,8	-34	-13.9808	-11.5022
8	2,6	34	-13.9808	-11.50222

Chapter 7: Conclusion

A complete framework is designed to be used in detecting how many users in the system and what type of modulation they are using. The major use is in both military and civilian applications. We showed that using SCF as a characterization of cyclostationarity of communication signals, we can design an effective with reasonable computational cost features for AMC. First, using supervised learning and convolutional neural networks, we perform the classification on both SCF and IQ and show that SCF based classification achieves better accuracy than IQ under Multipath fading channel and especially under higher values of paths delay and lower values of gains. Second, we show that SCF has much faster convergence and training time. Finally, to reduce the cost furthermore, we establish a compressed learning scheme using a small number of measurements obtained by linear projections of the input features. We perform the linear sensing and non-linear inference stages. The framework includes channelization stage, ED stage, Supervised learning stage with improvement to be implemented on real-time system. In future work, we are looking forward to optimize the results by working on the hyper-parameters optimization and wider range of parameters.

Bibliography

[1] M. W. Aslam, Z. Zhu, and A. K. Nandi, "Automatic modulation classification using combination of genetic programming and knn," *IEEE Transactions on wireless communications*, vol. 11, no. 8, pp. 2742–2750, 2012.

[2] L. Han, F. Gao, Z. Li, and O. A. Dobre, "Low complexity automatic modulation classification based on order-statistics," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, pp. 400–411, 2016.

[3] E. Like, V. Chakravarthy, R. Husnay, and Z. Wu, "Modulation recognition in multipath fading channels using cyclic spectral analysis," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*. IEEE, 2008, pp. 1–6.

[4] S. E. El-Khamy, H. A. Elsayed, and M. M. Rizk, "Classification of multiuser chirp modulation signals using higher order cumulant features and four types of classifiers," in *2011 28th National Radio Science Conference (NRSC)*. IEEE, 2011, pp. 1–10.

[5] A. Smith, M. Evans, and J. Downey, "Modulation classification of satellite communication signals using cumulants and neural networks," in *2017 Cognitive Communications for Aerospace Applications Workshop (CCAA)*. IEEE, 2017, pp. 1–8.

[6] T. Nawaz, L. Marcenaro, and C. S. Regazzoni, "Stealthy jammer detection algorithm for wide-band radios: A physical layer approach," in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2017, pp. 79–83.

[7] T. J. OShea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *International conference on engineering applications of neural networks*. Springer, 2016, pp. 213–226.

[8] T. J. OShea, 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.

[9] E. April, “On the implmentation of the strip spectral correlation algorithm for cyclic spectrum estimation.” DEFENCE RESEARCH ESTABLISHMENT OTTAWA (ONTARIO), Tech. Rep., 1994.

[10] C. Spooner, “Csp estimators: The time smoothing method,” Dec. 2015. [Online]. Available: https://cyclostationary.blog/

[11] A. N. M. Chad Spooner, “Wideband cyclostationary signal processing using sparse subsets of narrowband subchannels,” *IEEE transactions on cognitive communications and networks, VOL. 4, NO. 2*, Jun. 2018.

[12] M. Schmidt, D. Block, and U. Meier, “Wireless interference identification with convolutional neural networks,” in *2017 IEEE 15th International Conference on Industrial Informatics (INDIN).* IEEE, 2017, pp. 180–185.

[13] Z. Zhang and M. Sabuncu, “Generalized cross entropy loss for training deep neural networks with noisy labels,” in *Advances in neural information processing systems*, 2018, pp. 8778–8788.

[14] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[15] R. Calderbank, S. Jafarpour, and R. Schapire, “Compressed learning: Universal sparse dimensionality reduction and learning in the measurement domain,” *preprint*, 2009.

[16] S. Lohit, K. Kulkarni, and P. Turaga, “Direct inference on compressive measurements using convolutional neural networks,” in *2016 IEEE International Conference on Image Processing (ICIP).* IEEE, 2016, pp. 1913–1917.

[17] M. A. Davenport, M. F. Duarte, M. B. Wakin, J. N. Laska, D. Takhar, K. F. Kelly, and R. G. Baraniuk, “The smashed filter for compressive classification and target recognition,” in *Computational Imaging V*, vol. 6498. International Society for Optics and Photonics, 2007, p. 64980H.

[18] W. B. Johnson and J. Lindenstrauss, “Extensions of lipschitz mappings into a hilbert space,” *Contemporary mathematics*, vol. 26, no. 189-206, p. 1, 1984.

[19] A. Adler, M. Elad, and M. Zibulevsky, “Compressed learning: A deep neural network approach,” *arXiv preprint arXiv:1610.09615*, 2016.

[20] A. Kabán, "New bounds on compressive linear least squares regression," in *Artificial intelligence and statistics*, 2014, pp. 448–456.

[21] M. Shoaib, N. K. Jha, and N. Verma, "Signal processing with direct computations on compressively sensed data," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 1, pp. 30–43, 2014.

[22] K. Kulkarni and P. Turaga, "Reconstruction-free action inference from compressive imagers," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 4, pp. 772–784, 2015.

[23] A. Değerli, S. Aslan, M. Yamac, B. Sankur, and M. Gabbouj, "Compressively sensed image recognition," in *2018 7th European Workshop on Visual Information Processing (EUVIP)*. IEEE, 2018, pp. 1–6.

[24] O. A. Dobre, Y. Bar-Ness, and W. Su, "Higher-order cyclic cumulants for high order modulation classification," in *IEEE Military Communications Conference, 2003. MILCOM 2003.*, vol. 1. IEEE, 2003, pp. 112–117.

[25] ——, "Robust qam modulation classification algorithm using cyclic cumulants," in *2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No. 04TH8733)*, vol. 2. IEEE, 2004, pp. 745–748.

[26] A. Swami and B. M. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Transactions on communications*, vol. 48, no. 3, pp. 416–429, 2000.

[27] W. A. Gardner and C. M. Spooner, "The cumulant theory of cyclostationary time-series. i. foundation," *IEEE Transactions on signal processing*, vol. 42, no. 12, pp. 3387–3408, 1994.

[28] C. M. Spooner and W. A. Gardner, "The cumulant theory of cyclostationary time-series. ii. development and applications," *IEEE Transactions on Signal Processing*, vol. 42, no. 12, pp. 3409–3429, 1994.

[29] C. M. Spooner, "Classification of co-channel communication signals using cyclic cumulants," in *Conference Record of the Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1. IEEE, 1996, pp. 531–536.