Over-the-Air Deep Learning Based Radio Signal Classification

Timothy James O'Shea, *Senior Member, IEEE*, Tamoghna Roy, *Senior Member, IEEE*, and T. Charles Clancy, *Senior Member, IEEE*

***Abstract*—We conduct an in depth study on the performance of deep learning based radio signal classification for radio communications signals. We consider a rigorous baseline method using higher order moments and strong boosted gradient tree classification, and compare performance between the two approaches across a range of configurations and channel impairments. We consider the effects of carrier frequency offset, symbol rate, and multipath fading in simulation, and conduct over-the-air measurement of radio classification performance in the lab using software radios, and we compare performance and training strategies for both. Finally, we conclude with a discussion of remaining problems, and design considerations for using such techniques.**

***Index Terms*—Cognitive radio, deep learning, modulation, neural networks, pattern recognition, sensor systems and applications, wireless communication.**

I. Introduction

The ability to rapidly understand and label the radio spectrum in an autonomous way is a key enabling factor for spectrum interference monitoring, radio fault detection, dynamic spectrum access, opportunistic mesh networking, and numerous regulatory and defense applications. Boiling down a complex high-data rate flood of RF information to precise and accurate labels which can be acted on and conveyed compactly is a critical component today in numerous radio sensing and communications systems. For many years, radio signal classification and modulation recognition have been accomplished by carefully hand-crafting specialized feature extractors for specific signal types and properties and by and deriving compact decision bounds from them using either analytically derived decision boundaries or statistical learned boundaries within low-dimensional feature spaces.

In the past five years, we have seen rapid disruption precipitated by the improved neural network architectures, algorithms and optimization techniques collectively known as deep learning (DL) [1]. DL has recently replaced the machine learning (ML) state-of-the-art in computer vision (CV), voice and natural language processing; in both of these fields, feature engineering and pre-processing were once critically important topics, allowing cleverly designed feature extractors and transforms to extract pertinent information into a manageable reduced dimension representation from which labels or decisions could be readily learned with tools like support vector machines or decision trees. Among these widely used front-end features were the scale-invariant feature transform (SIFT) [2], the bag of words [3], Mel-frequency Cepstral coefficients (MFCC) [4] and others which were widely relied upon only a few years ago, but are no longer needed for state-of-the-art performance today.

DL greatly increased the capacity for feature learning directly on raw high dimensional input data based on high level supervised objectives due to the newly found capacity for learning of very large neural network models with high numbers of free parameters. This was made possible by the combination of strong regularization techniques [5], [6], greatly improved methods for stochastic gradient descent (SGD) [7], [8], low-cost, high-performance graphics-card processing power, as well as combining of key neural network architecture innovations such as convolutional neural networks [9], and rectified linear units [10]. It was not until Alexnet [11] that many of these techniques were used together to realize an increase of several orders of magnitude in the practical model size, parameter count, and target dataset and task complexity, which subsequently turned the process of feature learning from raw image data into the state-of-the-art. At this point, the trend in ML has been relentless towards the replacement of rigid simplified analytic features and approximate models with much more accurate high degrees-of-freedom (DOF) models derived from data using end-to-end feature learning. This trend has been demonstrated in vision, text processing, and voice, but has yet to be widely applied or fully realized on radio time series data sets.

We showed in [12] and [13] that these methods can be readily applied to simulated radio time series sample data in order to classify emitter types with excellent performance, obtaining equivalent accuracies several times more sensitive than existing best practice methods using feature based classifiers on higher order moments. In this work we provide a more extensive dataset of additional radio signal types, a more realistic simulation of the wireless propagation environment, over-the-air (OTA) measurement of the new dataset (i.e. real propagation effects), new methods for signal classification which drastically outperform those we initially introduced, and an in-depth analysis of many practical engineering design and system parameters that impact the performance and accuracy of the radio signal classifier.

Manuscript received July 14, 2017; revised November 16, 2017; accepted January 15, 2018. Date of publication January 23, 2018; date of current version February 16, 2018. The guest editor coordinating the review of this paper and approving it for publication was Dr. George Stantchev. *(Corresponding author: Timothy James O'Shea.)*

The authors are with the Bradley Department of Electrical and Computer Engineering, Blacksburg, VA 24061 USA, , and also with the DeepSig Inc., Arlington, VA 22201 USA (e-mail: oshea@vt.edu; tamoghna@vt.edu; tcc@vt.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2018.2797022

II. Background

A. Baseline Classification Approach

1) Statistical Modulation Features: For digital modulation techniques, higher order statistics and cyclo-stationary moments [14]–[18] are among the most widely used features to compactly sense and detect signals with strong periodic components such as are created by the structure of the carrier, symbol timing, and symbol structure for certain modulations. By incorporating precise knowledge of this structure, expected values of peaks in auto-correlation function (ACF) and spectral correlation function (SCF) surfaces have been used successfully to provide robust classification for signals with unknown or purely random data. For analog modulation where symbol timing does not produce these artifacts, other statistical features are useful in performing signal classification.

For our baseline features in this work, we leverage a number of compact higher order statistics (HOSs). To obtain these we compute the higher order moments (HOMs) using the expression given below:

$$M_{pq} = E[x^{p-q}(x^*)^q] \tag{1}$$

From these HOMs we can derive a number of higher order cumulantss (HOCs) which have been shown to be effective discriminators for many modulation types [17]. HOCs can be computed combinatorially using HOMs, each expression varying slightly; below we show one example such expression for the C_{40} HOC.

$$C_{40} = M_{40} - 3M_{20}^2 \tag{2}$$

In practice quadratically scaled HOCs (e.g. $\sqrt{C_{40}}$) can be used to improve feature scaling. Additionally we consider a number of analog features which capture other statistical behaviors which can be useful, these include mean, standard deviation and kurtosis of the normalized centered amplitude, the centered phase, instantaneous frequency, absolute normalized instantaneous frequency, and several others which have shown to be useful in prior work. [19].

2) Decision Criterion: When mapping our baseline features to a class label, a number of compact machine learning or analytic decision processes can be used. Probabilistically derived decision trees on expert modulation features were among the first to be used in this field, but for many years such decision processes have also been trained directly on datasets represented in their feature space. Popular methods here include support vector machines (SVMs), decision trees (DTrees), neural networks (NNs) and ensembling methods which combine collections of classifiers to improve performance. Among these ensembling methods are Boosting, Bagging [20], and Gradient tree boosting [21]. In particular, XGBoost [22] has proven to be an extremely effective implementation of gradient tree boosting which has been used successfully by winners of numerous Kaggle data science competitions [23]. In this work we opt to use the XGBoost approach for our feature classifier as it outperforms any single decision tree, SVM, or other method evaluated consistently as was the case in [13].

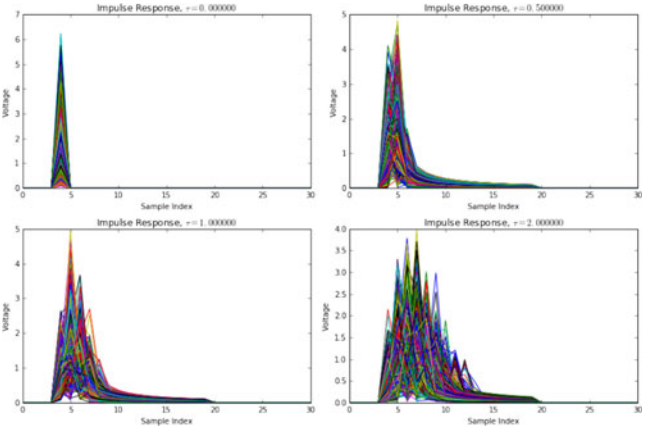

Fig. 1. Fading power delay profile examples.

B. Radio Channel Models

When modeling a wireless channel there are many compact stochastic models for propagation effects which can be used [24]. Primary impairments seen in any wireless channel include:

- *Carrier frequency offset (CFO):* carrier phase and frequency offset due to disparate local oscillators (LOs) and motion (Doppler).
- *Symbol rate offset (SRO):* symbol clock offset and time dilation due to disparate clock sources and motion.
- *Delay Spread:* non-impulsive delay spread due to delayed reflection, diffraction and diffusion of emissions on multiple paths.
- *Thermal Noise:* additive white-noise impairment at the receiver due to physical device sensitivity.

Each of these effects can be compactly modeled well and is present in some form on any wireless propagation medium. There are numerous additional propagation effects which can also be modeled synthetically beyond the scope of our exploration here.

C. Deep Learning Classification Approach

DL relies today on back-propagation with SGD to optimize large parametric neural network models. Since Alexnet [11] and the techniques described in Section I, there have been numerous architectural advances within CV leading to significant performance improvements. Neural networks are comprised of a series of layers which map each layer input h_0 to output h_1 using parametric dense matrix operations followed by non-linearities. This can be expressed simply as follows, where weights, W, have the dimension $|h_0 \times h_1|$, bias, b, has the dimension $|h_1|$ (constituting the layer parameters θ), and an activation is applied element-wise $|h_1|$ (e.g. $max(0, h_0 W + b)$ for rectified linear unit (ReLU)).

$$h_1 = \max(0, h_0 W + b) \tag{3}$$

Convolutional layers replicate weights W at strides across the input to reduce parameter count and enforce translation invariance. A loss function ($\mathscr{L}$), commonly categorical cross-entropy for classification tasks, between class labels y_i and predicted

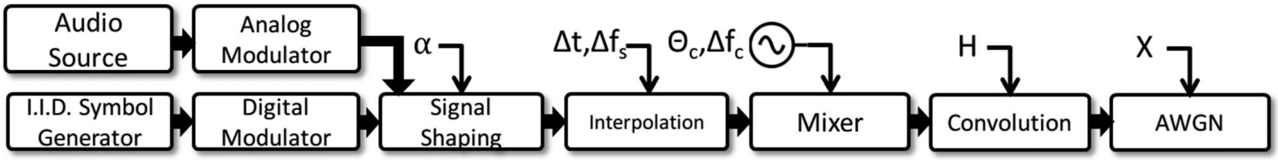

Fig. 2. System for dataset signal generation and synthetic channel impairment modeling.

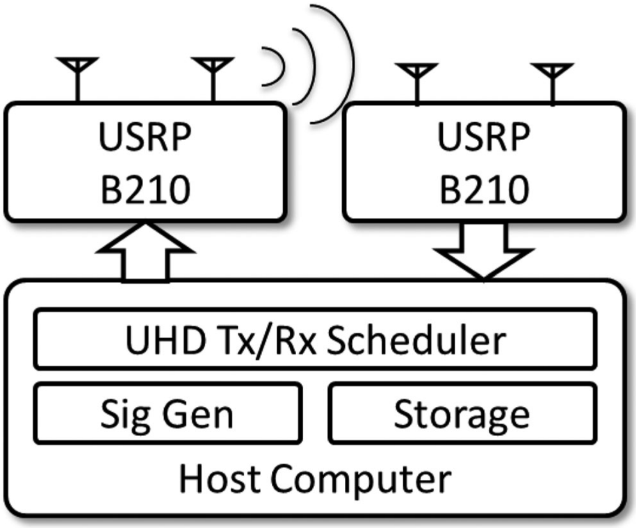

Fig. 3. Over-the-air test configuration.

class values $\hat{y}_i$ is used to compute error gradients.

$$\mathscr{L}(y, \hat{y}) = \frac{-1}{N} \sum_{i=0}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (4)$$

Back propagation of loss gradients fits layer weights (θ) of the network $f(x, \theta)$ until convergence (In our work, the Adam optimizer [8]).

$$\theta_{n+1} = \theta_n - \eta \frac{\partial \mathscr{L}(y, f(x, \theta_n))}{\partial \theta_n} \quad (5)$$

To reduce over-fitting to training data, regularization is used. We use batch normalization [6] to regularize convolutional layers and Alpha Dropout [25] for fully connected layers. Detailed descriptions of additional layers used including Soft-Max, Max-Pooling, etc are can be found in [1].

III. Dataset Generation Approach

We generate new datasets for this investigation by building upon an improved version of the tools described in [26]. 24 different analog and digital modulators are used which cover a wide range of single carrier modulation schemes. We consider several different propagation scenarios in the context of this work; first are several simulated wireless channels generated from the model shown in Fig. 2, and second we consider OTA transmission channel of clean signals as shown in Figs. 3 and 4 with no synthetic channel impairments. Digital signals are shaped with a root-raised cosine pulse shaping filter [27] with a range of roll-off values (α).

Fig. 4. Configuration for over-the-air transmission of signals.

TABLE I
Random Variable Initialization

Random Variable	Distribution
α	$U(0.1, 0.4)$
Δ_t	$U(0, 16)$
Δf_s	$N(0, \sigma_{clk})$
θ_c	$U(0, 2\pi)$
Δf_c	$N(0, \sigma_{clk})$
H	$\Sigma_i \delta(t - \text{Rayleigh}_i(\tau))$

For each example in the synthetic data sets, we independently draw a random value for each of the variables shown below in Table I. This results in a new and uncorrelated random channel initialization for each example.

Fig. 1 illustrates several random values for H, the channel impulse response envelope, for different delay spreads, $\tau = [0, 0.5, 1.0, 2.0]$, relating to different levels of multi-path fading in increasingly more difficult Rayleigh fading environments. Fig. 24 illustrate examples from the training set when using a simulated channel at low SNR (0 dB E_s/N_0).

We consider two different compositions of the dataset, first a "Normal" dataset, which consists of 11 classes that are all relatively low information density and are commonly seen in impaired environments. These 11 signals represent a relatively simple classification task at high SNR in most cases, somewhat comparable to the canonical MNIST digits. Second, we introduce a "Difficult" dataset, which contains all 24 modulations. These include a number of high order modulations (QAM256 and APSK256), which are used in the real world in very high-SNR, low-fading channel environments such as those associated with impulsive satellite links [28] (e.g. DVB-S2X). We however, apply impairments which are beyond that which one would expect to see in such a scenario and consider only relatively short-time observation windows for classification, where the number of samples (ℓ) is $= 1024$. Short-time classification is challenging but could be unavoidable when decision processes cannot wait to acquire more data to increase certainty. This is the case in many real world systems when dealing with short observations (such as when rapidly scanning a receiver) or short signal bursts in the environment. Under these effects, with low SNR examples (from -20 dB to $+30$ dB E_s/N_0), one would not expect to be able to achieve anywhere near 100% classification rates on the full dataset, making it a good benchmark for comparison and future research comparison.

The specific modulations considered within each of these two dataset types are as follows:

- *Normal Classes:* OOK, 4ASK, BPSK, QPSK, 8PSK, 16QAM, AM-SSB-SC, AM-DSB-SC, FM, GMSK, OQPSK
- *Difficult Classes:* OOK, 4ASK, 8ASK, BPSK, QPSK, 8PSK, 16PSK, 32PSK, 16APSK, 32APSK, 64APSK, 128APSK, 16QAM, 32QAM, 64QAM, 128QAM, 256QAM, AM-SSB-WC, AM-SSB-SC, AM-DSB-WC, AM-DSB-SC, FM, GMSK, OQPSK

The raw datasets will be made available on the RadioML website [1] shortly after publication and are shown in Figs. 23 and 24.

A. Over-the-Air Data Capture

In additional to simulating wireless channel impairments, we also implement an OTA test-bed in which we modulate and transmit signals using a universal software radio peripheral (USRP) [29] B210 software defined radio (SDR). We use a second B210 (with a separate free-running LO) to receive these transmissions in the lab, over a relatively benign indoor wireless channel on the 900 MHz ISM band. These radios use the Analog Devices AD9361 [30] radio frequency integrated circuit (RFIC) as their radio front-end and have an LO that provides a frequency (and clock) stability of around 2 parts per million (PPM). We off-tune our signal by around 1 MHz to avoid DC signal impairment associated with direct conversion, but store signals at base-band (offset only by LO error). Received test emissions are stored off unmodified along with ground truth labels for the modulation from the emitter.

TABLE II
FEATURES USED

Feature Name
M_{20}, M_{21}
$M_{40}, M_{41}, M_{42}, M_{43}$
$M_{60}, M_{61}, M_{62}, M_{63}$
C_{20}, C_{21}
$C_{40}, C_{41}, C_{42},$
$C_{60}, C_{61}, C_{62}, C_{63}$
Additional analog II-A

IV. SIGNAL CLASSIFICATION MODELS

In this section we explore the radio signal classification methods in more detail which we will use for the remainder of this paper.

A. Baseline Method

Our baseline method leverages the list of higher order moments and other aggregate signal behavior statistics given in Table II. Here we can compute each of these statistics over each 1024 sample example, and translate the example into feature space, a set of real values associated with each statistic for the example. This new representation has reduced the dimension of each example vector from $1024 * 2$ to 28, making the classification task much simpler but also discarding the vast majority of the data. We use an ensemble model of gradient boosted trees (XGBoost) [22] to classify modulations from these features, which outperforms a single decision tree or support vector machine (SVM) significantly on the task.

B. Convolutional Neural Network

A typical convolutional neural network (CNN) architecture, widely used in CV, comprises a number of convolutional layers followed by fully connected layers (FC) in classifiers [9], [11]. In [31], the question of how to structure such networks is explored, and several basic design principles for network architectures, developed by the Visual Geometry Group (VGG) at Oxford University, are introduced (e.g. filter size is minimized at 3 × 3, smallest size pooling operations are used at 2 × 2). Following this approach has generally led to straightforward way to construct CNNs with good performance. We adapt the VGG architecture principles to a 1D CNN as shown in Table III, improving upon the similar networks in [12] and [13]. This represents a simple DL CNN design approach which can be readily trained and deployed to effectively accomplish many small radio signal classification tasks.

Of significant note here is that the features into this CNN are the raw I/Q samples of each radio signal example which have been normalized to unit variance. We do not perform any expert feature extraction or other pre-processing on the raw radio signal, instead allowing the network to learn raw time-series features directly on the high dimension data. Real valued networks are used, as complex valued auto-differentiation is not yet mature enough for practical use.

[1] https://radioml.org

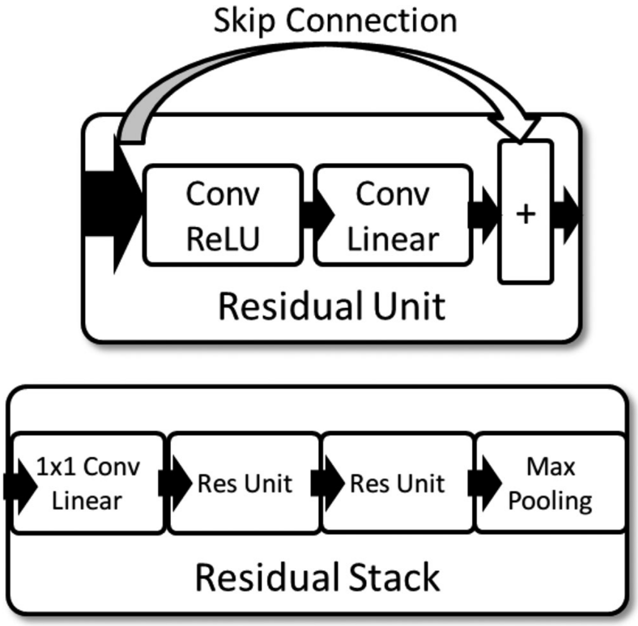

Fig. 5. Hierarchical layers used in network.

C. Residual Neural Network

As network algorithms and architectures have improved since Alexnet, they have made the effective training of deeper networks using more and wider layers possible, and leading to improved performance. In our original work [12] we employ only a small convolutional neural network with several layers to improve over the prior state-of-the-art. However in the CV domain, the idea of deep residual networks (RNs) has become increasingly effective [32]. In a deep RN, as is shown in Fig. 5, the notion of skip or bypass connections is used heavily, allowing for features to operate at multiple scales and depths through the network. This has led to significant improvements for both CV and time-series audio tasks [33]. In [34], RNs for time-series radio classification is investigated, and seen to train in fewer epochs, but not to provide significant performance improvements in terms of accuracy. We revisit the problem of modulation recognition with a modified RN and obtain improved performance as compared to the CNN. The residual unit and stack of residual units used are shown in Fig. 5, while the network architecture for our best architecture for ($\ell = 1024$) is shown in Table IV. We also employ self-normalizing neural networks [25] in the fully connected layers, using scaled exponential linear unit (SELU) activation functions, mean-response scaled initialization (MRSA) [35], and Alpha Dropout [25], to provide an improvement over the conventional ReLU+Dropout approach.

For the network layouts shown, with $\ell = 1024$ and $L = 5$, the RN has 236,344 trainable parameters, while the CNN/VGG network has a close 257,099.

V. Sensing Performance Analysis

There are numerous design, deployment, training, and data considerations which can significantly affect the performance of a DL based approach to radio signal classification which must be considered when designing a solution. We explore several of the most common factors which impact accuracy including propagation effects, model size/depth, data set sizes, observation window, and modulation type.

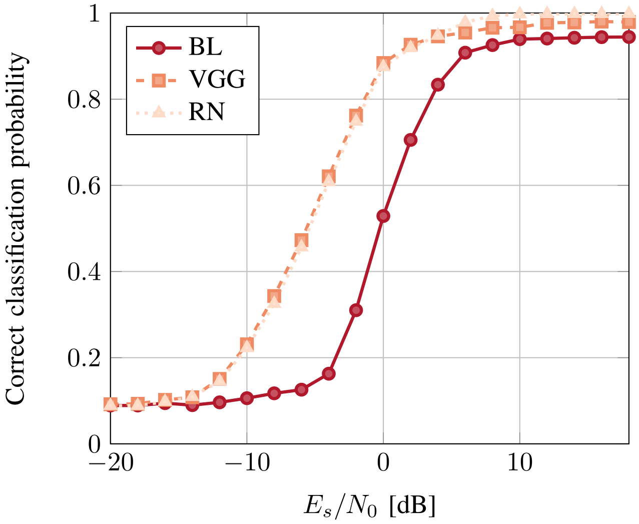

Fig. 6. 11-modulation AWGN dataset performance comparison (N = 1 M)

TABLE III
CNN Network Layout

Layer	Output dimensions
Input	2 × 1024
Conv	64 × 1024
Max Pool	64 × 512
Conv	64 × 512
Max Pool	64 × 256
Conv	64 × 256
Max Pool	64 × 128
Conv	64 × 128
Max Pool	64 × 64
Conv	64 × 64
Max Pool	64 × 32
Conv	64 × 32
Max Pool	64 × 16
Conv	64 × 16
Max Pool	64 × 8
FC/Selu	128
FC/Selu	128
FC/Softmax	24

TABLE IV
ResNet Network Layout

Layer	Output dimensions
Input	2 × 1024
Residual Stack	32 × 512
Residual Stack	32 × 256
Residual Stack	32 × 128
Residual Stack	32 × 64
Residual Stack	32 × 32
Residual Stack	32 × 16
FC/SeLU	128
FC/SeLU	128
FC/Softmax	24

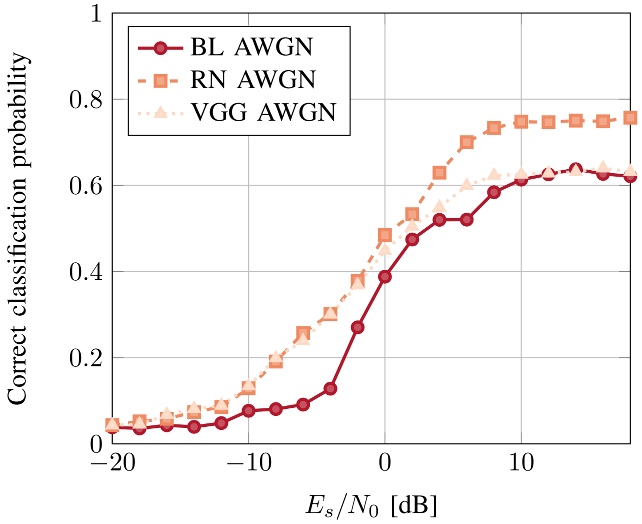

Fig. 7. Comparison models under AWGN (N = 240 k).

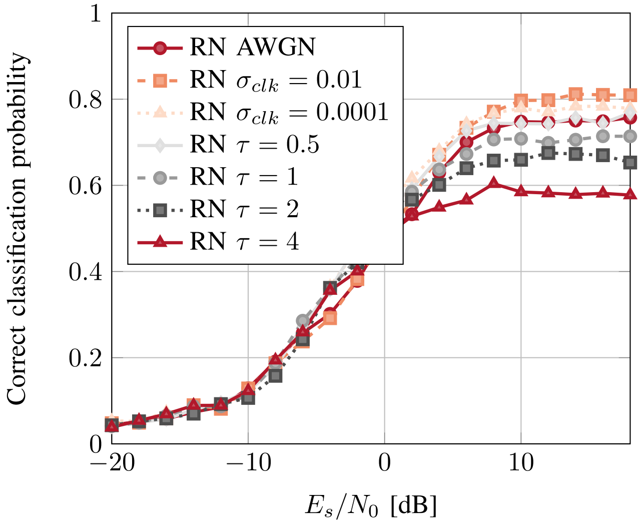

Fig. 8. ResNet performance under various channel impairments (N = 240 k).

A. Classification on Low Order Modulations

We first compare performance on the lower difficulty dataset on lower order modulation types. Training on a dataset of 1 million example, each 1024 samples long, and obtaining excellent performance at high SNR for both the VGG CNN and the RN as shown in Fig. 6.

The RN achieves roughly 5 dB higher sensitivity for equivalent accuracy compared with the baseline. At high SNR a maximum classification accuracy rate of 99.8% is achieved by the RN, while the VGG network achieves 98.3% and the baseline method achieves a 94.6% accuracy. At lower SNRs, the performance of VGG and RN networks is similar, but at high-SNR performance improves considerably using the RN.

For the remainder of the paper, we consider the harder task of classification on 24 high order modulations at higher information rates which are traditionally easier to confuse (e.g. high order PSK, QAM, APSK).

B. Classification Under AWGN Conditions

Signal classification under additive white Gaussian noise (AWGN) is the canonical problem which has been explored for many years in communications literature. It is a simple starting point, and it is the condition under which analytic feature extractors should generally perform its best (matching their analytic model assumptions). In Fig. 7 we compare the performance of the RN, VGG network, and the baseline (BL) method on the full dataset for $\ell = 1024$ samples, $N = 239,616$ examples, and $L = 6$ residual stacks. Here, the residual network provides the best performance at both high and low SNR on the difficult dataset by a margin of 2-6 dB in sensitivity.

C. Classification Under Impairments

In any real world scenario, wireless signals are impaired by a number of effects. While AWGN is widely used in simulation and modeling, the effects described above are present almost universally. It is important to consider how well learned classifiers perform under such impairments and understand how these effects degradation their performance in comparison to traditional methods.

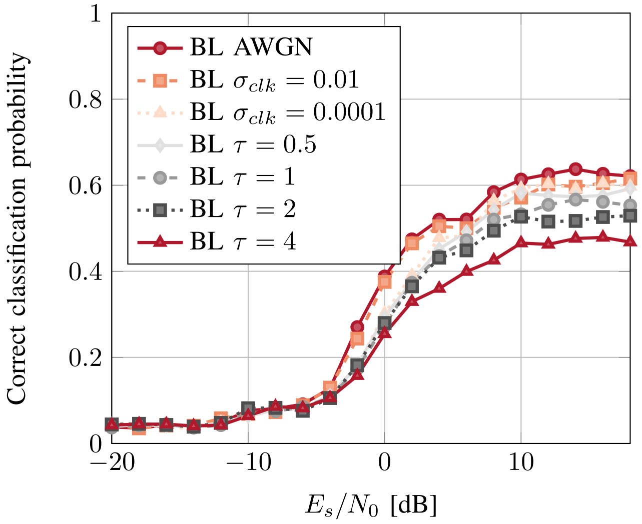

Fig. 9. Baseline performance under channel impairments (N = 240 k).

In Fig. 8 we plot the performance of the RN based classifier under each considered impairment model. This includes AWGN, $\sigma_{clk} = 0.0001$ - minor LO offset, $\sigma_{clk} = 0.01$ - moderate LO offset, and several fading models ranging from $\tau = 0.5$ to $\tau = 4.0$. Under the fading models, moderate LO offset is assumed as well. Interestingly in this plot, RN performance improves under LO offset rather than degrading. Additional LO offset which results in spinning or dilated versions of the original signal, appears to have a positive regularizing effect on the learning process which provides quite a noticeable improvement in performance. At high SNR performance ranges from around 80% in the best case down to about 59% in the worst case.

In Fig. 9 we show the degradation of the baseline classifier under impairments. In this case, LO offset never helps, but the performance instead degrades with both LO offset and fading

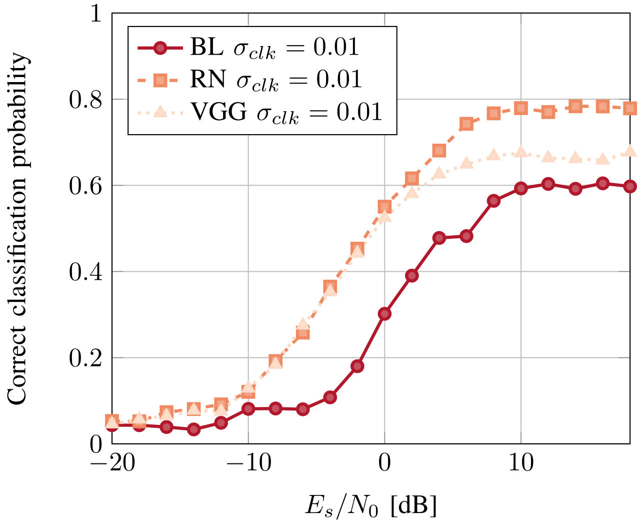

Fig. 10. Comparison models under LO impairment.

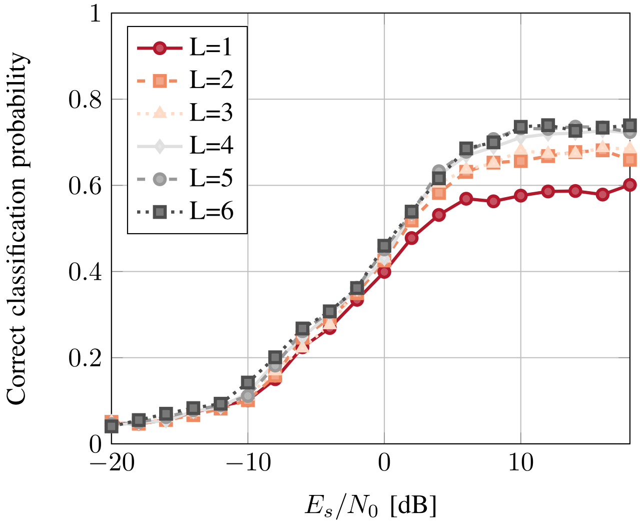

Fig. 11. ResNet performance vs depth (L = number of residual stacks).

effects, in the best case at high SNR this method obtains about 61% accuracy while in the worst case it degrades to around 45% accuracy.

Directly comparing the performance of each model under moderate LO impairment effects, in Fig. 10 we show that for many real world systems with unsynchronized LOs and Doppler frequency offset there is nearly a 6 dB performance advantage of the RN approach vs the baseline, and a 20% accuracy increase at high SNR. In this section, all models are trained using $N = 239,616$ and $\ell = 1024$ for this comparison.

D. Classifier Performance by Depth

Model size can have a significant impact on the ability of large neural network models to accurately represent complex features. In CV, convolutional layer based DL models for the ImageNet dataset started around 10 layers deep, but modern state-of-the-art networks on ImageNet are often over 100–200 layers deep [36]. Initial investigations of deeper networks [34] did not show significant gains from deeper architectures, but with the use of deep RNs on this larger dataset, we begin to see quite a benefit to additional depth. This is likely due to the significantly larger number of examples and classes used. In Fig. 11 we show the increasing validation accuracy of deep RNs as we introduce more residual stack units within the network architecture (i.e. making the network deeper). We see that performance steadily increases with depth in this case with diminishing returns as we approach around 6 layers. When considering all of the primitive layers within this network, when $L = 6$ the RN has 121 layers and 229 k trainable parameters, when $L = 0$ it has 25 layers and 2.1 M trainable parameters. Results are shown for $N = 239,616$ and $\ell = 1024$.

E. Classification Performance by Modulation Type

In Figs. 12–14 we show the performance of the classifier for individual modulation types. Equivalent classification accuracy for each modulation type varies about 18 dB of signal to noise ratio (SNR). Some signals with lower information rates and unique structure such as AM and FM are more readily identified at low SNR, while high-order modulations require higher SNRs for robust performance. AM modulations with a carrier tend to be easier to classify, and in general higher order modulations require higher signal to noise ratios to achieve equivalent classification accuracy. All modulation types reach rates 80%+ accuracy by 10 dB SNR. In Fig. 15 we show a confusion matrix for the classifier across all 24 classes for AWGN validation examples where SNR is greater than or equal to zero. The largest sources of error are between high order phase shift keying (PSK) (16/32-PSK), between high order quadrature amplitude modulation (QAM) (64/128/256-QAM), as well as between AM modes (confusing with-carrier (WC) and suppressed-carrier (SC)). For short-time observations, with substantial noise, especially with higher order QAM and PSK, significant error is expected simply due to lack of information and similar symbol structure using this or any other known prior method. For AM

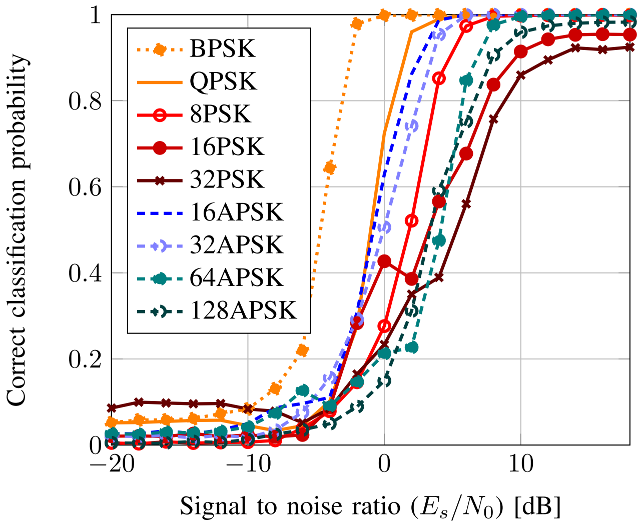

Fig. 12. PSK-style modulation performance (ResNet, Synthetic N = 1 M, $\sigma_{clk} = 0.0001$)

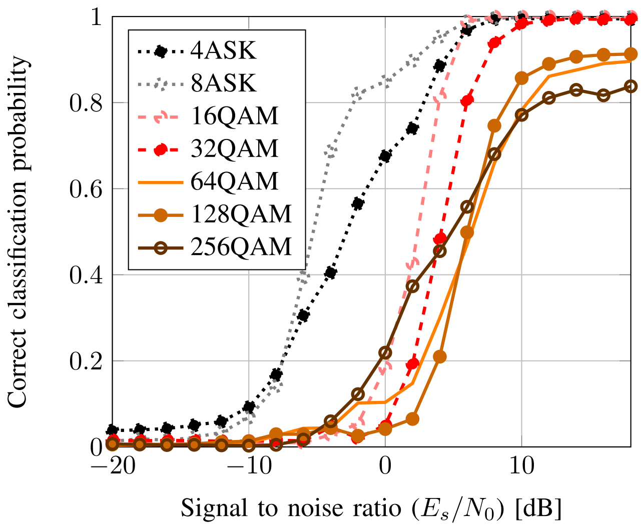

Fig. 13. QAM/ASK-style modulation performance (ResNet, synthetic N = 1 M, σ_{clk} = 0.0001).

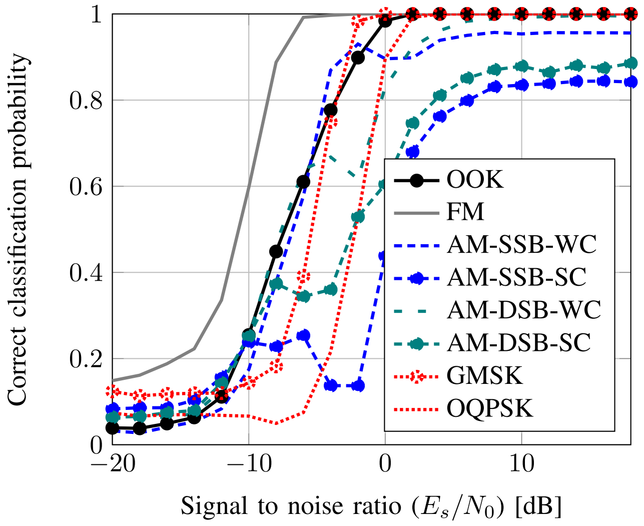

Fig. 14. Low order & analog modulation performance (ResNet, synthetic N = 1 M, σ_{clk} = 0.0001).

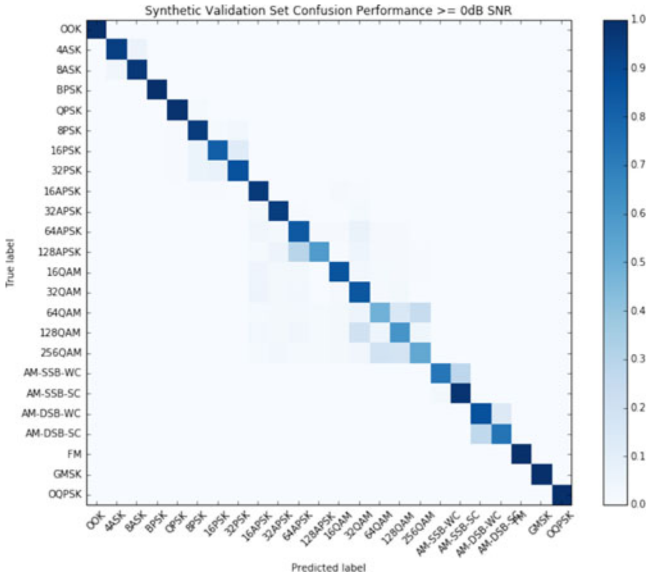

Fig. 15. 24-modulation confusion matrix for ResNet trained and tested on synthetic dataset with N = 1 M and AWGN.

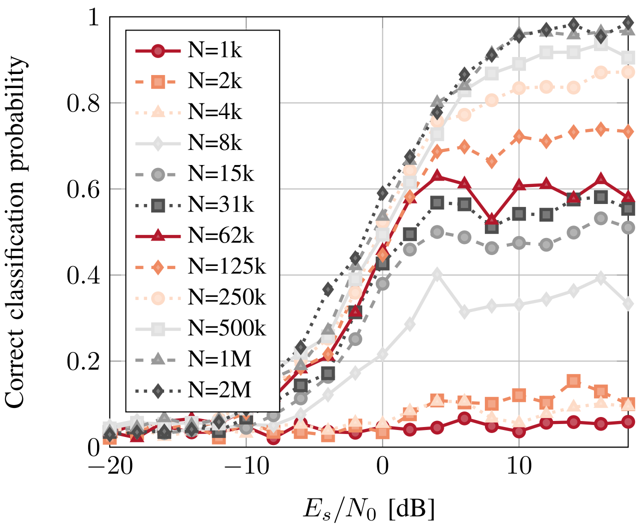

Fig. 16. Performance vs training set size (N) with $\ell = 1024$.

modulations, we suspect additional voice data set size might improve performance. ASK modulations provide surprisingly good performance, as we generally observe that amplitude variation seems to provide better performance than phase variation (8ASK is one of the best modes, while 32PSK is quite poor). Representation of inputs in Cartesian form may be related, it would be interesting for example to consider relative classification performance when using polar representation as well.

F. Classifier Training Size Requirements

When using data-centric machine learning methods, the dataset often has an enormous impact on the quality of the model learned. We consider the influence of the number of example signals in the training set, N, as well as the time-length of each individual example in number of samples, ℓ.

In Fig. 16 we show how performance of the resulting model changes based on the total number of training examples used. Here we see that dataset size has a dramatic impact on model training, high SNR classification accuracy is near random up to 4–8 k examples and improves by 5–20% with each doubling up to around 1 M. These results illustrate that having sufficient training data is critical for performance. For the largest case, with 2 million examples, training on a single state-of-the-art NVIDIA V100 graphics processing unit (GPU) (with approximately 125 TFLOPS) takes around 16 hours to converge, making significant experimentation at these dataset sizes cumbersome. We do not see significant improvement going from 1 M to 2 M examples, indicating a point of diminishing returns for number of examples around 1 M with this configuration. With either 1 M or 2 M examples we obtain roughly 95% test set accuracy at high SNR. The class-confusion matrix for the best performing mode with ℓ

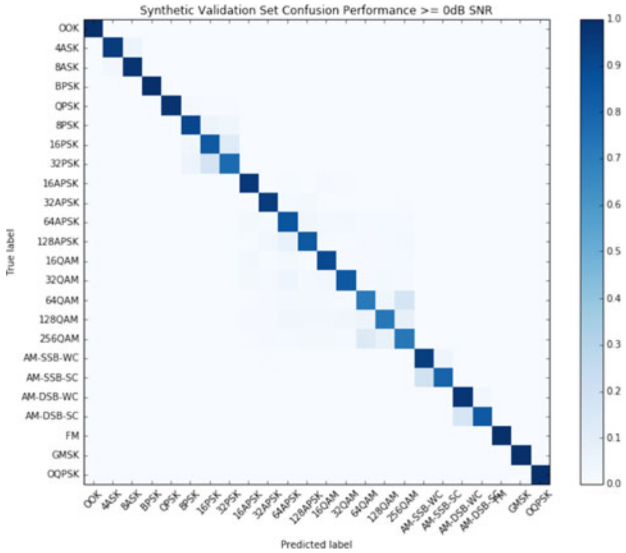

Fig. 17. 24-modulation confusion matrix for ResNet trained and tested on synthetic dataset with N = 1 M and $\sigma_{clk} = 0.0001$.

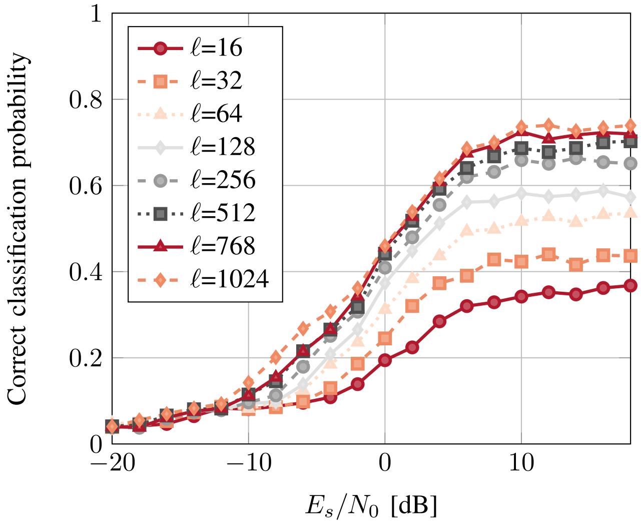

Fig. 18. Performance vs example length in samples (ℓ).

$= 1024$ and N = 1 M is shown in Fig. 17 for test examples at or above 0 dB SNR, in all instances here we use the $\sigma_{clk} = 0.0001$ dataset, which yields slightly better performance than AWGN.

Fig. 18 shows how the model performance varies by window size, or the number of time-samples per example used for a single classification. Here we obtain approximately a 3% accuracy improvement for each doubling of the input size (with N = 240 k), with significant diminishing returns once we reach $\ell = 512$ or $\ell = 1024$. We find that CNNs scale very well up to this 512–1024 size, but may need additional scaling strategies thereafter for larger input windows simply due to memory requirements, training time requirements, and dataset requirements.

G. *Over-the-Air Performance*

We generate 1.44 M examples of the 24 modulation dataset OTA using the USRP setup described above. Using a partition of 80% training and 20% test, we can directly train a RN for classification. Doing so on an NVIDIA V100 in around 14 hours, we obtain a 95.6% test set accuracy on the OTA dataset, where all examples are roughly 10 dB SNR. A confusion matrix for this OTA test set performance based on direct training is shown in Fig. 19.

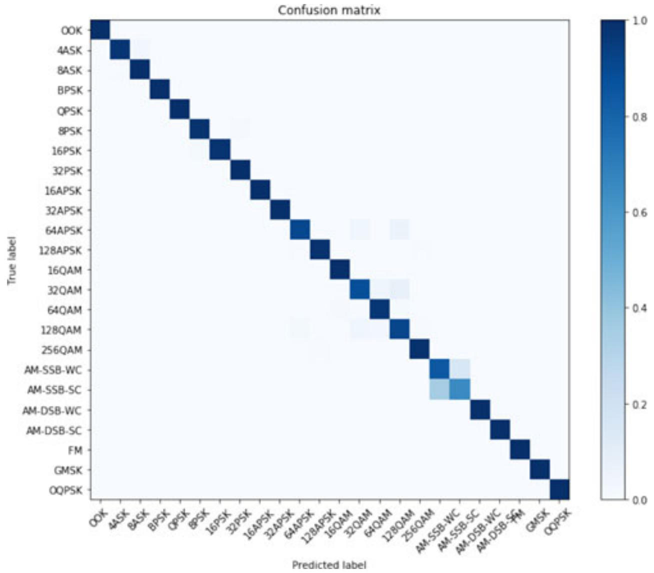

Fig. 19. 24-modulation confusion matrix for ResNet trained and tested on OTA examples with SNR ∼ 10 dB.

H. *Transfer Learning Over-the-Air Performance*

We also consider OTA signal classification as a transfer learning [37], [38] problem, whereby the model is trained on synthetic data and then only evaluated and/or fine-tuned on OTA data. Because full model training can take hours on a high-end GPU and typically requires a large dataset to be effective, transfer learning is a convenient alternative for leveraging existing models and updating them on smaller computational platforms and target datasets. We consider transfer learning, where we freeze network parameter weights for all layers except the last several fully connected layers (last three layers from Table IV) in our network while updating. This is commonly done today with CV models where it is common start by using pre-trained VGG or other model weights for ImageNet or similar datasets and perform transfer learning using another dataset or set of classes. In this case, many low-level features work well for different classes or datasets, and do not need to change during fine tuning. In our case, we consider several cases where we start with models trained on simulated wireless impairment models using RNs and then evaluate them on OTA examples. The accuracies of our initial models (trained with N = 1 M) on synthetic data shown in Fig. 8, and these ranged from 84% to 96% on the hard 24-class dataset. Evaluating performance of these models on OTA data, without any model updates, we obtain classification accuracies between 64% and 80%. By fine-tuning the last two layers of these models on the OTA data using transfer learning, we and can recover approximately 10% of additional

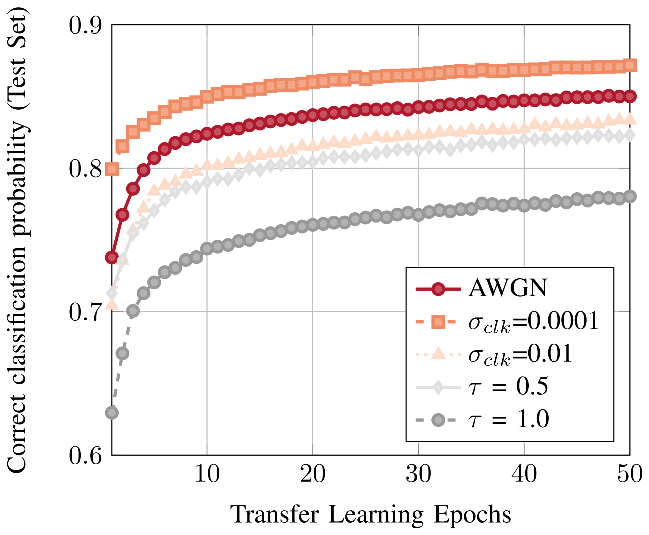

Fig. 20. ResNet transfer learning OTA performance (N = 120 k).

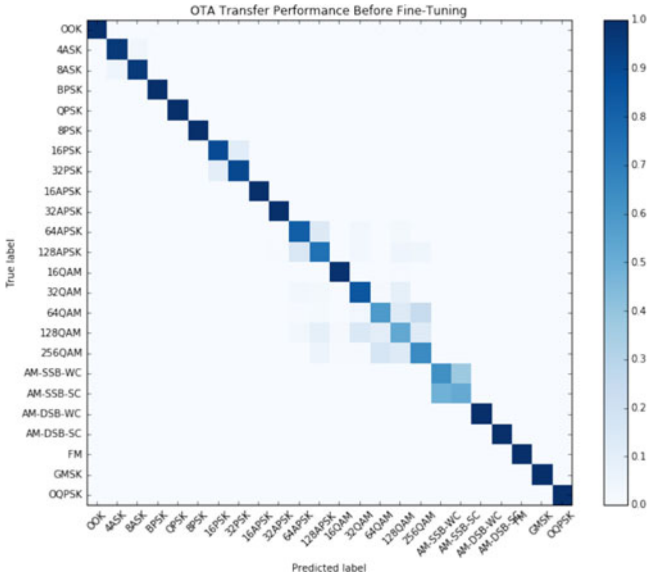

Fig. 21. 24-modulation confusion matrix for ResNet trained on synthetic $\sigma_{clk} = 0.0001$ and tested on OTA examples with SNR ∼10 dB (prior to fine-tuning)

accuracy. The validation accuracies are shown for this process in Fig. 20. This transfer learning process updates only the final dense layers' weights using 120 k examples and take roughly 60 seconds per epoch on an NVIDIA Titan X GPU to execute whereas to update weights for all layers of the RN model even on a faster NVIDIA V100 GPU takes approximately ∼500 seconds per epoch due to the increased number and depth of gradient computations required.

Ultimately, the model trained on just moderate LO offset (σ_{clk} = 0.0001) performs the best on OTA data. The model obtained 94% accuracy on synthetic data, and drops roughly 7% accuracy when evaluating on OTA data, obtaining an accuracy of 87% corresponding confusion matrix performance can be seen in Figs. 21 and 22. The primary confusion cases prior to training seem to be dealing with suppressed or non-suppressed carrier analog signals, as well as the high order QAM and APSK modes.

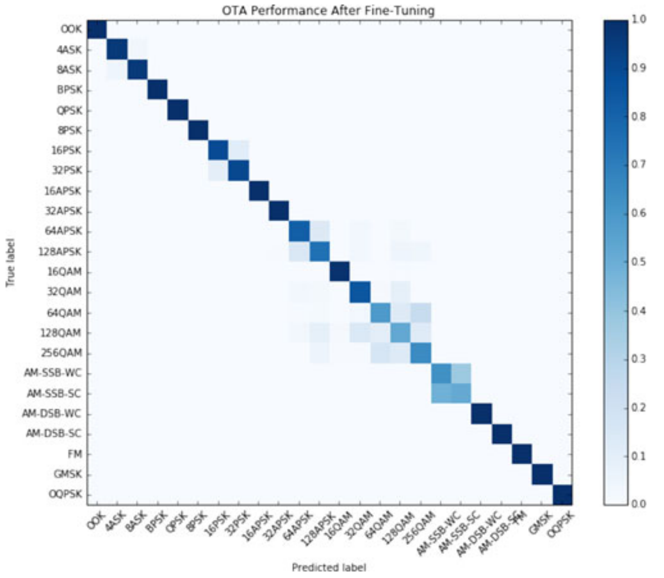

Fig. 22. 24-modulation confusion matrix for ResNet trained on synthetic $\sigma_{clk} = 0.0001$ and tested on OTA examples with SNR ∼ 10 dB (after fine-tuning).

This seems like perhaps the best suited among our models to match the OTA data for several reasons. First, only very small LO deviation is present in the OTA data because the radio oscillators (unlocked GPSDO units) provide stable $\tilde{7}$5 PPB clocks over short example lengths of 1024 samples. And second, due to the near proximity of the radios indoors and clean line-of-sight dominated impulse response with comparably little energy contained at higher delay indirect paths, a very low (near impulsive) delay spread is appropriate. Training on harsher impairments seemed to degrade performance of the OTA data significantly.

We suspect as we evaluate the performance of the model under increasingly harsh real world scenarios, our transfer learning will favor synthetic models which are similarly impaired and most closely match the real wireless conditions (e.g. matching LO distributions, matching fading distributions, etc). In this way, it will be important for this class of systems to train either directly on target signal environments, or on very good impairment emulation models under which well suited models can be derived. A possible mitigation option is to include domain-matched attention mechanisms such as the radio transformer network [26] in the network architecture to improve generalization to varying wireless propagation conditions.

VI. Discussion

In this work we have extended prior work on using deep convolutional neural networks for radio signal classification by carefully tuning deep RNs for the same task. We have also conducted a much more thorough set of performance evaluations on how this type of classifier performs over a wide range of design parameters, channel impairment conditions, and training dataset parameters. This RN approach achieves state-of-the-art modulation classification performance on a difficult new signal database both synthetically and in OTA

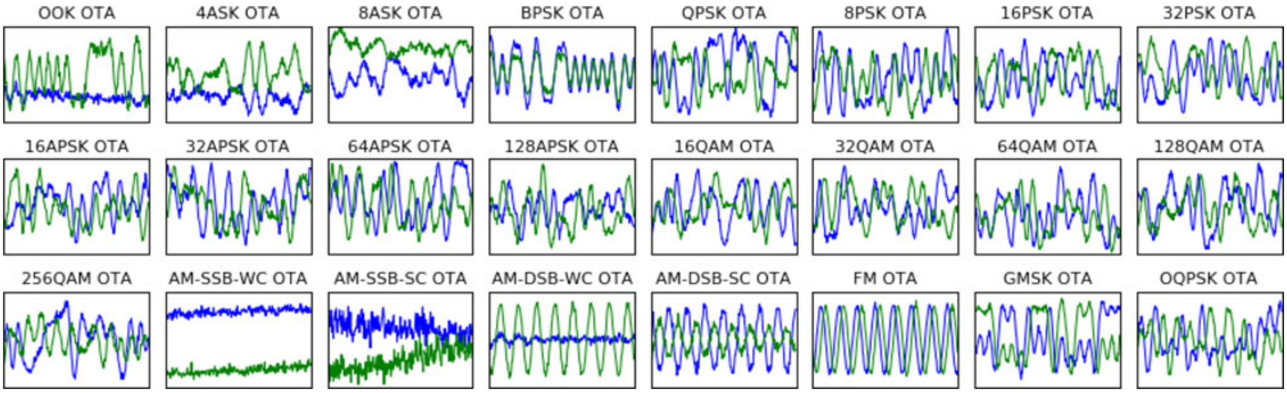

Fig. 23. I/Q time domain examples of 24 modulations over-the-air at roughly 10 dB E_s/N_0 ($\ell = 256$).

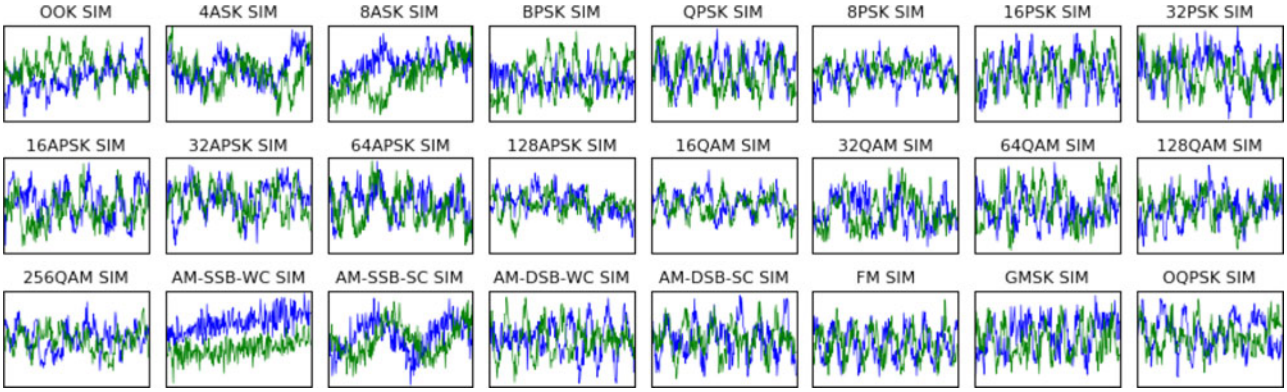

Fig. 24. I/Q time domain examples of 24 modulations from synthetic $\sigma_{clk} = 0.01$ dataset at 2 dB E_s/N_0 ($\ell = 256$).

performance. Other architectures, such as radio transformer networks, recurrent units, still hold significant potential, however they will also need to be adapted to the domain, tuned and quantitatively benchmarked against the same dataset in the future. Other works have explored these to some degree, but generally not with sufficient hyper-parameter optimization to be meaningful.

We have shown that, contrary to prior work, deep networks do provide significant performance gains for time-series radio signals where the need for such deep feature hierarchies was not apparent, and that RNs are a highly effective way to build these structures in cases where more traditional CNNs, such as VGG networks, tend to struggle to achieve the same performance, or to make effective use of the network depth. We have also shown that simulated channel effects, especially moderate LO impairments improve the effect of transfer learning to OTA signal evaluation performance, a topic which will require significant future investigation to optimize the synthetic impairment distributions used for training.

VII. Conclusion

DL methods continue to show enormous promise in improving radio signal identification sensitivity and accuracy, especially for short-time observations. We have shown deep networks to be increasingly effective when leveraging deep residual architectures and have shown that synthetically trained deep networks can be effectively transferred to OTA datasets with (in our case) a loss of around 7% accuracy or directly trained effectively on OTA data if enough training data is available. While large, well labeled datasets can often be difficult to obtain for such tasks today, and channel models can be difficult to match to real-world deployment conditions, we have quantified the real need to do so when training such systems and helped quantify the resulting performance impact.

We still have much to learn about how to best curate datasets and training regimes for this class of systems. However, we have demonstrated in this work that our approach provides roughly the same performance on high SNR OTA datasets as it does on the equivalent synthetic datasets, a major step towards real world use. We have demonstrated that transfer learning can be effective and fast, especially when limited transfer-training data is available, but in our results does not obtain equivalent accuracy/sensitivity performance as compared with direct training of a full model on a sufficiently large dataset. As simulation methods become better, and our ability to match synthetic datasets to real world data distributions improves, this gap will close and both transfer learning and data augmentation will become increasingly important tools when large scale real data capture and labeling is difficult, expensive, or time-consuming. The performance trade-offs shown in this work help shed light on these key parameters in data generation and training, increase understanding, and focus future efforts on the optimization of such systems.

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[3] M. Vidal-Naquet and S. Ullman, "Object recognition with informative features and linear classification," in *Proc. Int. Conf. Comput. Vis.*, 2003, vol. 3, pp. 281–288.

[4] S. Imai, "Cepstral analysis synthesis on the mel frequency scale," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1983, vol. 8, pp. 93–96.

[5] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[6] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[7] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, 2012.

[8] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Proc. 3rd Int. Conf. Learning Representation*, 2014.

[9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[10] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[12] T. J. OShea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. Neural Netw.*, 2016, pp. 213–226.

[13] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, 2017.

[14] W. A. Gardner and C. M. Spooner, "Signal interception: performance advantages of cyclic-feature detectors," *IEEE Trans. Commun.*, vol. 40, no. 1, pp. 149–159, Jan. 1992.

[15] C. M. Spooner and W. A. Gardner, "Robust feature detection for signal interception," *IEEE Trans. Commun.*, vol. 42, no. 5, pp. 2165–2173, May 1994.

[16] C. M. Spooner, A. N. Mody, J. Chuang, and J. Petersen, "Modulation recognition using second-and higher-order cyclostationarity," in *Proc. 2017 IEEE Int. Symp. Dyn. Spectr. Access Netw.*, 2017, pp. 1–3.

[17] A. Abdelmutalab, K. Assaleh, and M. El-Tarhuni , "Automatic modulation classification based on high order cumulants and hierarchical polynomial classifiers," *Phys. Commun.*, vol. 21, pp. 10–18, 2016.

[18] A. Fehske, J. Gaeddert, and J. H. Reed, "A new approach to signal classification using spectral correlation and neural networks," in *Proc. 2005 1st IEEE Int. Symp. New Frontiers Dyn. Spectr. Access Netw.*, 2005, pp. 144–150.

[19] A. K. Nandi and E. E. Azzouz, "Algorithms for automatic modulation recognition of communication signals," *IEEE Trans. Commun.*, vol. 46, no. 4, pp. 431–436, Apr. 1998.

[20] J. R. Quinlan *et al.*, "Bagging, boosting, and c4. 5," in *Proc. AAAI/IAAI*, 1996, vol. 1, pp. 725–730.

[21] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, pp. 1189–1232, 2001.

[22] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 785–794.

[23] A. Goldbloom, "Data prediction competitions—Far more than just a bit of fun," in *Proc. 2010 IEEE Int. Conf. Data Mining Workshops*, 2010, pp. 1385–1386.

[24] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[25] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," *Adv. Neural Inform. Proc. Sys.*, vol. 30, pp. 972–981, 2017.

[26] T. J. O'Shea and N. West, "Radio machine learning dataset generation with GNU radio," in *Proc. GNU Radio Conf.*, 2016, vol. 1, no. 1.

[27] J. G. Proakis, *Digital Communications*. New York, NY, USA: McGraw-Hill, 1995.

[28] S. Cioni *et al.*, "Transmission parameters optimization and receiver architectures for DVB-S2X systems," *Int. J. Satell. Commun. Netw.*, vol. 34, no. 3, pp. 337–350, 2016.

[29] M. Ettus and M. Braun, "The universal software radio peripheral (USRP) family of low-cost SDRD," *Opportunistic Spectrum Sharing and White Space Access: The Practical Reality*. Hoboken, NJ, USA: Wiley, 2015, pp. 3–23.

[30] Analog Devices, RF Agile Transceiver AD9361. [Online]. Available: http://www.analog.com/static/imported-files/data_sheets/ad9361.pdf, Vis ited on: September 14, 2008.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556, 2014.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[33] A. V. D. Oord *et al.*, "WaveNet: A generative model for raw audio," arXiv:1609.03499, 2016.

[34] N. E. West and T. J. O'Shea, "Deep architectures for modulation recognition," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw.*, 2017, pp. 1–6.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[36] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[37] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1717–1724.

[38] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2014, pp. 806–813.

Timothy James O'Shea (S'05–M'08–SM'13) received the B.S./M.S. degrees in electrical and computer engineering from North Carolina State University, Raleigh, NC, USA, in 2007, and the Ph.D. degree in electrical engineering from Virginia Tech, Blacksburg, VA, USA, in 2017.

He is currently a Research Associate with Virginia Tech and the CTO with DeepSig Inc., Arlington, VA, USA. Prior to joining Virginia Tech, he was a Research Engineer with a UMIACS affiliated research center at the University of Maryland. He has been a core contributor and technical advisor to the FSF GNU Radio project since 2006. His research interests include leveraging machine learning advances to develop new and improved solutions to radio signal processing, data security, and other information processing problems.

Tamoghna Roy (SM'12) received the B.S. degree from Jadavpur University, Kolkata, India, in 2009, and the M.S. and Ph.D. degrees from Virginia Tech, Blacksburg, VA, USA, in 2014 and 2017, respectively, all in electrical engineering. He is currently a Principal Engineer with DeepSig Inc., Arlington, VA, USA. His research interests include statistical signal processing, machine learning, stochastic modeling, and deep learning.

T. Charles Clancy (S'03–M'06–SM'10) received the B.S. degree in computer engineering from Rose-Hulman Institute of Technology, Terre Haute, IN, USA, the M.S. degree in electrical engineering from the University of Illinois, Champaign, IL, USA, and the Ph.D. degree in computer science from the University of Maryland, College Park, MD, USA.

He directs Virginia Tech's Hume Center for National Security and Technology and is currently a Professor in electrical and computer engineering. With more than 70 faculty and staff, the Hume Center engages 350 students annually in research and experimental learning focused in national security and technology. He is an internationally recognized expert in wireless cybersecurity. Prior to joining Virginia Tech in 2010, he was a Researcher with the Laboratory for Telecommunications Sciences, a federal research lab at the University of Maryland. He has contributed to more than 200 peer-reviewed technical publications and patents, is the coauthor of four books, and the cofounder of four venture-backed startup companies.