# Hybrid Noise-Resilient Deep Learning Architecture for Modulation Classification in Cognitive Radio Networks

Antoni Ivanov[1](✉), Krasimir Tonchev[1], Vladimir Poulkov[1], Hussein Al-Shatri[2], and Anja Klein[2]

[1] Faculty of Telecommunications, Technical University of Sofia, 1000 Sofia, Bulgaria
{astivanov,k_tonchev,vkp}@tu-sofia.bg

[2] Department of Electrical Engineering and Information Technology,
Technical University Darmstadt, 64283 Darmstadt, Germany
{h.shatri,a.klein}@nt.tu-darmstadt.de

**Abstract.** The increasing maturity of the concepts which would allow for the operation of a practical Cognitive Radio (CR) Network require functionalities derived through different methodologies from other fields. One such approach is Deep Learning (DL) which can be applied to diverse problems in CR to enhance its effectiveness by increasing the utilization of the unused radio spectrum. Using DL, the CR device can identify whether the signal comes from the Primary User (PU) transmitter or from an interferer. The method proposed in this paper is a hybrid DL architecture which aims at achieving high recognition rate at low signal-to-noise ratio (SNR) and various channel impairments including fading because such are the relevant conditions of operation of the CR. It consists of an autoencoder and a neural network structure due to the good denoising qualities of the former and the recognition accuracy of the latter. The autoencoder aims to restore the original signal from the corrupted samples which would increase the accuracy of the classifier. Afterwards its output is fed into the NN which learns the characteristics of each modulation type and classifies the restored signal correctly with certain probability. To determine the optimal classification DL model, several types of NN structures are examined and compared for input comprised of the IQ samples of the reconstructed signal. The performance of the proposed DL architecture in comparison to similar models for the relevant parameters in different channel impairments scenarios is also analyzed.

**Keywords:** Cognitive Radio · Deep Learning · Modulation classification · Spectrum sensing

V. Poulkov (Ed.): FABULOUS 2019, LNICST 283, pp. 214–227, 2019.
https://doi.org/10.1007/978-3-030-23976-3_20

## 1 Introduction

The increasing effort to improve the applicability of modern telecommunications and introduce new generations of networks and services has significantly accelerated the maturity of all concepts and functionalities of CR systems. This tendency has also influenced the development of the spectrum sensing function which allows for dynamic access to the spectrum that is not by default allocated for the CR device but has incumbent (primary) users, the transmission of which must be protected from intolerable levels of interference. For this reason it is required for this function to be fast and accurate enough in order to determine whether the frequency band is available (the incumbent users' signals are not present in it) or not [10]. In addition, it is desirable to have the possibility to predict the amount of time during which the spectrum is expected to remain unused by the primary users (PU) to avoid hampering their transmission. In order to increase the utilization of the spectrum by the CR network it can be beneficial for the equipment not only to detect the signal present in the spectrum but also to identify its type. That is because the CR device is required to be able to detect very weak signals to ensure that the primary user is distinguished from the noise even at the edge of the PU network's coverage [25]. As a result, the cognitive equipment may confuse an interfering signal from the point of view of the PU for the actual PU signal. In this case, the CR device will miss the opportunity to utilize a portion of the spectrum in which the incumbent user (IU) is not in fact present. This is why studies on recognizing the type and source of the received signal, are necessary. Consequently there have been multiple works which attempted to achieve this [1,4,6,10,11,18,19,22,30]. Other applications of signal recognition include identifying illegal transmitters or malfunctioning equipment, TV white space access planning, improving the capabilities of Emergency and Public Safety services, mapping spectrum occupancy, electrosmog monitoring, location identification for military purposes, demodulation without overhead information, reconnaissance and satellite relay selection [22,28].

Extensive research in the field of signal interception has led to the introduction of modulation classification (MC) as the most commonly employed technique for recognition of the received signal's type [4]. There have been a large variety of proposed algorithms for achieving efficient signal recognition but they can in general be separated into two groups - likelihood [4,29] or feature [18,22] based MC. The advantages of the former group of methods is the possibility to recognize a large variety of signals with very little or none a priori data. Their implementation in realistic receivers, however, could be problematic considering that they might need intolerable amount of time to estimate the necessary parameters for signal classification [29]. The receiver devices themselves might not be designed with the computational power to handle these complex operations. In contrast the concept behind the feature-based methods is the possibility for near-instantaneous recognition of the modulation type of a received signal. These algorithms use a supervised learning procedure like Machine Learning (ML). Only in the recent years has the modern concept of Deep Learning (DL) been applied to the MC problem [1,2,9,13,16,18,19,21,22,30–33]. The characteristic trait of these algorithms is the utilization of large amounts of preprocessed data which is used for preliminary training a deep Neural Network (NN) that afterwards will be able to recognize newly-received signals with sufficient probability under specified conditions (most often the SNR level). At this point the emerging obstacles before using DL for signal recognition are seen as following. First of all, a sufficient volume of suitable data for the training of the NN needs to be recorded or generated. Subsequently, the design of the NN itself is not trivial as there are a few parameters which require careful and most often empirical determination. Finally the training process typically has a high computational cost on the host computer and can take a lot of time depending on the amount of data and the choice of parameters of the NN. These considerations will be explored in the subsequent sections of this paper. Section 2 presents the review of available literature for MC using DL algorithms. The parameters for dataset generation and the channel and noise models are described in Sect. 3. Section 4 details the proposed architecture and the relevant parameters. In Sect. 5 the results in terms of recognition accuracy are analyzed and the conclusions are discussed in Sect. 6.

## 2 State of the Art

The current works related to the field of DL-based MC can be summarized according to the following aspects: the kinds of input given to the NN, the amount of data vectors (or realizations of the signal) used for training and testing, the DL architectures utilized for the classification and the channel model which is used for generation of the testing data.

When it comes to the data used for the training of the NN, there are mostly two types of input in regard to signal recognition - the vector of signal samples [1,18,22,30], statistical features extracted from those samples [1,2,13,16,19,21,28], or a combination of both [1]. Vast majority of the published works examine cyclostationary and other kinds of statistical features derived from the signals as inputs of the classifier. These are utilized for MC because they represent the signal components which are resilient to the effects of noise. For this reason they have gained popularity with likelihood-based algorithms and are consequently explored in the studies of MC based on DL. The most often utilized input is the high-order statistical features [1,13,19] which are easy to define mathematically and they are not very computationally intensive [17]. Other features used are the center points in the modulation constellation [1], more varied statistical features like kurtosis, peak to average power ratio, etc. [13] and the amplitude, frequency and phase of the signal which are estimated as part of the learning process of the NN [21]. When the signal itself is fed into the NN, it is most often represented as a matrix of 2 columns and $N$ rows where $N$ is the number of samples, the first column contains the in-phase while the second - the quadrature components of the input signal [18,22]. There is also a vast margin between the number of data vectors utilized for training and testing varying from a few thousand to several hundreds of thousand realizations. It is apparent, though, that whenever larger amounts of signal data vectors were used, they contained much less samples so in terms of overall volume of data, there is not a big difference. The largest database of 1.44 million signal realizations each composed of 1024 samples was studied in [19].

All of the main DL structures have so far been studied in the literature for application to the MC problem because they are all useful for recognition of data which consists of sequential samples (like the signal representation in time domain) [5]. These include Convolutional NN (CNN) [2,18,19,30], Recurrent NN (RNN) [22], autoencoder (AE) [1,30] and Restricted Boltzmann Machines (RBM) [16]. The most often utilized DL architecture is the CNN with different number of layers[1], normally consisting of from 2 to 4 [2,18,30] but can reach up to 7 [19]. CNNs employ convolution instead of multiplication in their layers and are especially interesting because they can learn the features of the input data without the need for them to be extracted separately [5]. Generally the number of nodes (filters) in each convolutional sub-layer is in the order of hundreds. As for the RNN, it normally consists of 2 long-short term memory (LSTM) [22] or gated recurrent unit (GRU) layers and a fully-connected (FC) layer. Their structure is based around cells which process the data through gates and are able to classify sequential data. The AE has also been popular in recent works [1,30] because of its property to learn and reproduce (encode and decode) the form of a given input signal to its output with sufficiently low error [5]. Structures composed of 2 hidden layers [1] or 3 to 4 convolutional AE layers [30] which combine the traditional encoder/decoder of the AE with layers which are typically used in CNNs. Similarly to the AE, the RBM models aim to reconstruct the input data into its output but each node is connected to all the nodes in the consecutive layer [5]. This model is used in [16] and it has 5 layers. The input data used is images representing the spectral correlation function (SCF) of the signals. Additionally, some papers examine more novel DL models like the hierarchical residual network [19] or Extensible NN [21].

Finally there comes the important question of what channel model should be chosen for the testing data as to determine the performance of the proposed solution in realistic environments. Some of the studies [30,33] include just additive white Gaussian noise (AWGN) in their considerations but there are those that introduce frequency and phase offset [1,19] as well as Rayleigh, Rician fading or both [13,19,21]. A few [2,18,19,22] have utilized real-world recorded signals produced using software-defined radio (SDR) transceivers and the GNU Radio [26] package which are publicly available.

As a result of the analysis done in this section, the contributions of this paper are the following: design of a multi-layered architecture which combines an AE (Denoising AE or DAE) for the purpose of recovering distorted signals

[1] The term “layer” in the context of CNNs throughout this paper, is understood in the sense described in [5], i.e. each “complex” convolutional layer is composed of sub-layers which can, because of the coherence of their functions, be denoted as a single unit.

and a NN classifier with improved optimization algorithms; comparison between the performance of a CNN and of an RNN classifier with the inclusion of three (CNN and RNN-based) DL models presented in the literature which have shown very good recognition accuracy in low SNR levels ($<5$ dB) [22,30]. In addition to testing the NN performance with signals in AWGN and Rayleigh fading channels, the effects of non-Gaussian noises and generalized fading on the precision are also explored. Some of the most often considered modulation types are used for input of the DL model.

## 3 Dataset Generation

This Section gives an outline of the way in which the input signals are generated and the channel models are simulated. This is done in MATLAB and the resulting signal data is saved in files to be later used as an input of the NNs.

There are 11 modulation types considered in this study - BPSK, QPSK, PSK8, 16QAM, 64QAM, PAM4, AM-DSB, AM-SSB, GFSK, OFDM-16 (OFDM symbols with 16QAM modulated bits) and OFDM-64 (with 64QAM). All of them are generated with the same parameters as follows. Each data vector consists of a matrix composed of two vectors, 2048 samples each, the first containing the real while the other - the imaginary component of the signal. The sampling frequency is 1 GHz, the carrier frequency is 100 MHz, each bit is represented by 8 samples, bandwidth is 25 MHz. The length of the cyclic prefix of the OFDM signals is 15% of the length of each symbol, the modulation depth for the amplitude modulations is 30% and the standard deviation for the Gaussian filter used to form the GFSK signals, is equal to 6. There are 4096 realizations of each modulation for training and 512 for testing. The test set contains the same 5632 realizations (512 per modulation type) for each SNR level in the range $[-20; 20]$ dB. There are 6 test sets which contain signals corrupted by different combinations of channel distortions.

As it was stated in Sect. 2, to the best of the authors' knowledge, signals corrupted in complex fading and non-Gaussian channels have not been studied in works examining MC based on DL. Therefore, some channel impairments studied in signal detection literature are described here.

### 3.1 Middleton Noise

The Middleton Class A is a narrowband impulse noise model which describes the "coherent" interference created by man-made sources (mostly unintended radiations by various appliances, antennas, etc.) [3,24]. This form of the noise has been used in some signal detection and reception performance studies to model impulsive distortions [3,24]. It can be presented analytically and it is thus convenient for simulations. The probability density function (PDF) of the Middleton A noise used to describe it and the relevant parameters are found in [24].

### 3.2 Cauchy Noise

The Cauchy noise has also been studied in signal detection literature [8,27] due to its attributes which make it viable for describing impulse distortions in the propagation environment. These are created by normal human activities, the mechanical operation of machines, natural phenomena and others [27]. The PDF of the Caucy noise is given in [8]. In order to define the SNR levels for both the Cauchy and Middleton noises, the generalized expression for the SNR defined in [27] is used.

### 3.3 Generalized Gamma Fading

The generalized Gamma (also known as Stacey or $\alpha - \mu$) distribution is a basis out of which several of the most commonly used fading models can be derived. It consists of a non-linear sum of the multipath components and represents the small-scale fluctuations of the received signal [7]. The distribution of this fading model as well as the applicable parameters to describe it are taken from [7].

## 4 Proposed Architecture

This Section describes the proposed DL algorithm's structure, the methodology of training and testing, and how the input which consists of signal vectors in time domain is processed. First, the DAE learns the shape of the training signal data vectors (they do not have any channel impairments added to them) and afterwards, reconstructs the signals from the corrupted testing set. For all DL models examined in this study the training dataset is randomly shuffled. The recovered test data vectors are then saved into files and the classifier NN is trained with the same training set as the DAE was. Finally the reproduced testing set is given to the classifier which gives the probability for correct recognition between all modulation types for each SNR level. All of the algorithms were run on a NVIDIA TITAN X (Pascal) graphical processor unit which was kindly donated by the NVIDIA Corporation. They were implemented using Tensorflow.

### 4.1 Denoising Autoencoder

As it was stated earlier, this study explores how the DAE's capability to reduce the effect of noise by recovering the signal's shape, influences the performance of the NN classifier. A representation of the DAE's structure is shown in Fig. 1. It is composed of two layers, the first having 64 nodes and the second - 32. Both of them operate by simple matrix multiplication (Eq. (1)) use the sigmoid activation function. Increasing the number of layers leads to degradation in the performance. Using only two layers is also established in other works which study the AE structure [1,33].

$$\boldsymbol{Y} = \boldsymbol{x}\boldsymbol{W} + \boldsymbol{b}, \tag{1}$$

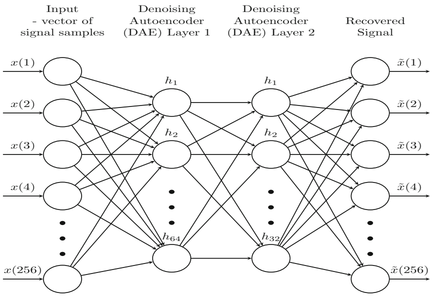

**Fig. 1.** Structure of the DAE

where the output $\boldsymbol{Y}$ is obtained by multiplying the input vector $\boldsymbol{x}$ by the matrix of weights $\boldsymbol{W}$, and adding the biases $\boldsymbol{b}$.

The Minimum mean square error (MMSE) method is used to reduce the cost function and the optimization is performed by the popular and effective Adam algorithm [12]. Empirically it was deduced that the autoencoder reconstructs shorter sequences with greater precision than long ones and therefore the input is divided into signals composed of 256 samples[2] (it is for this reason that the DAE has 256 input nodes). After the reconstruction is done the signals $\tilde{x}$ are reshaped again into series of 2048 samples. Another experimental observation is that the DAE provides the lowest error when the input signals are normalized in the $[0, 1]$ interval, whereas if the data is not normalized, the error becomes too high for viable reconstruction to be achieved. Faster convergence (in this case, obtaining the lowest cost function possible) is achieved by utilizing low learning rate (0.001) and batch size (11264 for the DAE, which means that the number of batches into which the training set will be divided is 64). Combining low values of both of these parameters is also commonly used in DL algorithms [5,18,20,22]. With this configuration, the training is performed in about 30 min.

### 4.2 Convolutional Neural Network

The first deep NN classifier the performance of which is analyzed in this paper, is the CNN. It learns from the training set and afterwards takes the reconstructed test sets which include the same signal data vectors for all SNR levels in the $[-20, 20]$ dB interval and all fading and noise scenarios as described in Sect. 3. Then, the CNN gives the classification accuracy for each SNR level and each scenario. The structure of the model is illustrated in Fig. 2

As it is seen in Fig. 2, the CNN contains four 2-dimensional convolutional layers, each one composed of convolution sub-layer, rectified linear unit (ReLU) and a 2-dimensional max-pooling sub-layer (using the terminology described in [5]), a ReLU layer and a class-prediction layer. Essentially, all convolution

[2] Consequently, the number of data vectors in both the training and test sets will be 8 times higher than it was originally.

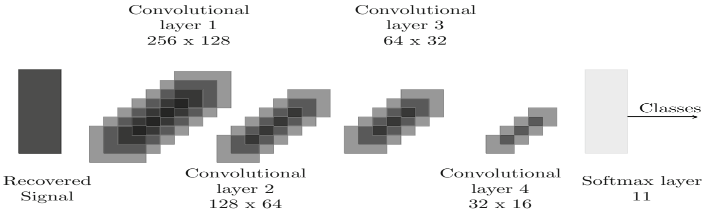
Convolutional layer 1 256 x 128
Convolutional layer 3 64 x 32
Classes
Recovered Signal
Convolutional layer 2 128 x 64
Convolutional layer 4 32 x 16
Softmax layer 11

**Fig. 2.** Structure of the CNN

sub-layers and the prediction layer implement Eq. (1). The input of the CNN is 4 dimensional tensor with dimensions [Batch Size , 2 , 2048 , 1] because as explained in Sect. 3, each signal realization is composed of a [2 × 2048] matrix. As seen in Fig. 2, the first convolution sub-layer has a kernel with dimensions [256 × 128], the second with [128 × 64], third - [64 × 32] and fourth with [32 × 16]. All max-pooling sub-layers have kernel sizes of [2 × 2] and strides with dimensions of [2 × 2]. At the end of the CNN there is a softmax layer which is a widely-used activation function. Such depth of the NN structure is employed to increase the capability of the model to learn sophisticated functions such as signals corrupted by noise and fading. As it was the case for the DAE, the Adam optimizer is utilized but it is also enhanced with a weight-decay algorithm which adapts the weights with small gradients [15]. This modification allows for separation of the weight decay and the gradients' update in order for the regularization to be applied properly for the Adam optimizer. Recognition accuracy is computed using MMSE. The weight decay factor is 0.0001 [30], the exponential decay rates are $\beta_1 = 0.9 , \beta_2 = 0.999$ and the stability constant $\epsilon$ is $10^{-8}$ [15]. As for the learning rate and batch size, they are usually chosen to both have small values (around 0.001 and 40 or more times smaller than the training set's size) but in this work, the recently proposed in the field of image classification, opposite direction is followed [23]. The authors in [23] conclude that using batch size around 10 times smaller than the training set and high learning rate can produce roughly the same results as in the alternative case but reach convergence much faster. For this reason, the values chosen for the learning rate and batch size are 0.1 and 4096 (number of batches is 11), respectively. Preliminary experiments showed that very similar results are achieved if small learning rates and batch sizes are used but, as expected, after much slower training. Increasing the depth of the model does not give any significant performance gains as well. The time needed for training of the CNN is discussed in Sect. 5.

### 4.3 Recurrent Neural Network

The performance of the deep CNN is compared to a mutli-layered RNN which is described here. It is trained with the same training set but it was shown that it cannot recognize any of the reconstructed test datasets even after rigorous training. Thus, this classifier uses the noisy test data that is provided to the DAE, i.e. no denoising is performed on it.

The model takes an $[2 \times 2048]$ input and is composed of 5 LSTM layers, a ReLU and a class prediction layer at the end in the same way as the CNN. A recently developed modification of the LSTM layers based on the Independently RNN (IndRNN) is used [14]. These LSTM layers are called IndyLSTM and they differ from the traditional ones by the way their states are updated. Instead of multiplication the new input by the previous state, the Hadamard product of these two is found. This alteration introduces some important advantages to the training of the RNN, namely, independence of the nodes in each layer, efficient training due to LSTMs being enhanced for robustness against gradient decay and ability to handle much longer series of data. Additionally, constructing multi-layered RNNs is also much more viable [14]. As it is recommended in the reference, the IndRNNs require low learning rate in order to converge, however large batch size can still be used for faster training. After preliminary experiments, optimal results were achieved for the following parameters. The learning rate is 0.001, batch size is 4096, the number of nodes in each layer is 128 and the forget bias of the IndyLSTMs is 1. Again, the Adam optimizer with weight decay is employed in this classifier with the same parameters. Training is performed in about an hour.

## 5 Results

This Section presents a thorough analysis of the results in terms of recognition accuracy for the two proposed DL classifiers in six channel scenarios:

Scenario 1: AWGN
Scenario 2: Additive white Gaussian, Middleton and Cauchy noises
Scenario 3: AWGN and generalized Gamma fading
Scenario 4: AWGN and Rayleigh fading
Scenario 5: Additive white Gaussian, Middleton and Cauchy noises, and generalized Gamma fading
Scenario 6: Additive white Gaussian, Middleton and Cauchy noises, and Rayleigh fading.

In addition, three NN classifiers from [22,30] are used for reference. They follow the structure described in the respective papers that proposed them and are indicated as "Reference RNN" [22], "Reference Convolutional AE" and "Reference CNN" [30]. For better clarity, the graphical representations of the results for every two scenarios in the case in which the input is the signal vectors in time domain, are shown in a common plot in Figs. 3, 4 and 5. The effective SNR range of $[-20; 20]$ dB which is relevant for CR applications, is considered in all experiments.

The most notable characteristics exposed in all graphics are those of the CNN classifiers (the proposed CNN, the reference convolutional AE and CNN). In Figs. 3, 4 and 5 it is seen that for reference CNN the recognition accuracy is constant for the whole SNR range. The convolutional AE and proposed CNN models show some insignificant variations in the results. This effect is related

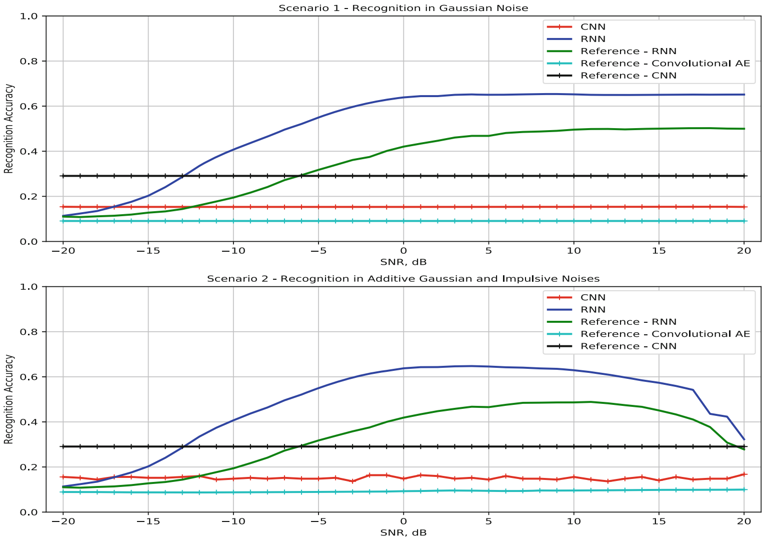

**Fig. 3.** Recognition accuracy in Scenarios 1 and 2 with signal input

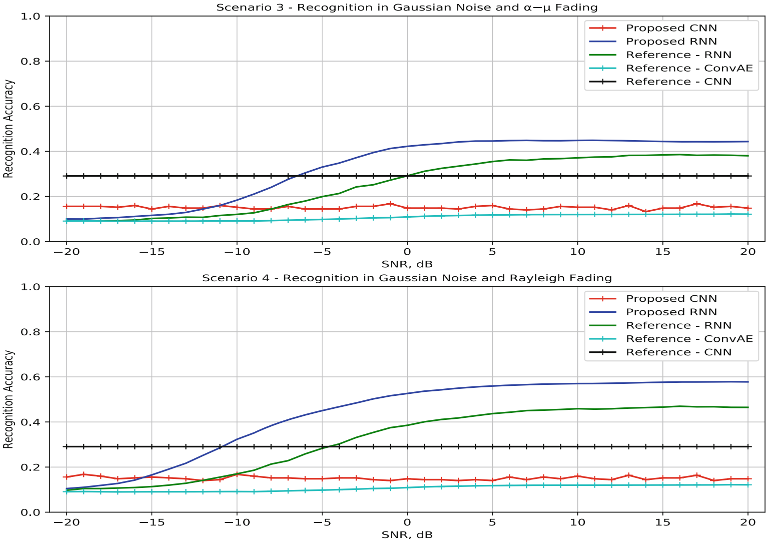

**Fig. 4.** Recognition accuracy in Scenarios 3 and 4 with signal input

to the training of the CNNs which showed that at every run, the NN gets to a certain accuracy for the noiseless test sets very quickly and at that point, it does not show any alternation in its learning process. Consequently, all test datasets have the same classification accuracy as the one obtained during training. For that reason, the results are collected in the following manner. Each CNN is

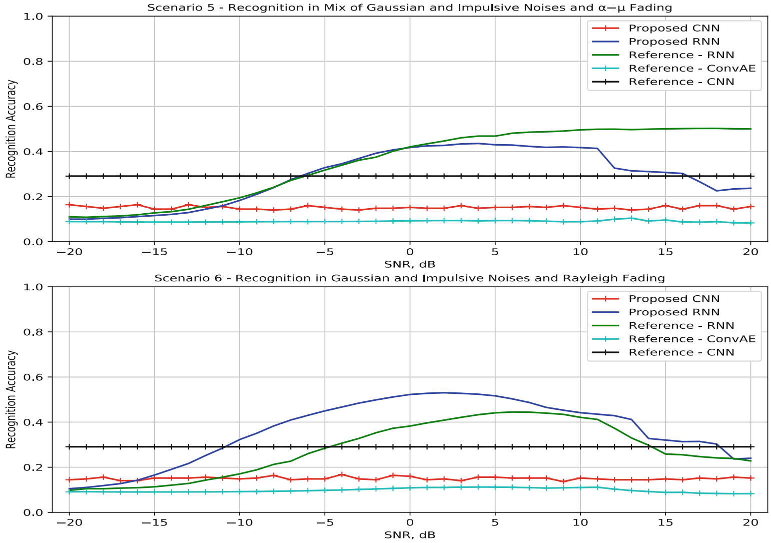

**Fig. 5.** Recognition accuracy in Scenarios 5 and 6 with signal input

trained 13 times and the average of the achieved accuracy values, is taken for all test sets and in all scenarios. Thus, the model is trained in about two hours which combines all 13 runs of the classifier. It is evident that the "Reference Convolutional AE" has more noticeable fluctuations than the proposed CNN and the reference CNN and they demonstrate a linearly-ascending trend. However, this model still shows the worst performance. In the results for the proposed CNN, there is some insignificant variation where as the "Reference CNN" has the best accuracy but it is yet poor.

The effects of the channel on the classification accuracy can be explored in much greater depth using the curves of the proposed and reference RNNs. In almost all channel scenarios, the RNN presented in this study shows much greater performance gains in comparison to its alternative. Figures 3 and 5 illustrate an interesting trend in that the accuracy experiences a significant decline in high SNR levels (>10 dB). This tendency is not present in the scenarios which exclude impulse noises so naturally, it can be attributed to them. The reason can be found in the distortions that these noise components introduce into the signal. Their impulsive nature reduces the classification efficiency of the RNN because it dramatically changes the shape of the signal. The deterioration in high SNR rates can be ascribed to the way that the SNR is calculated for the impulse noises (as explained in Sect. 3). As a consequence, the peaks added to the signal in high SNR, even though they are much smaller than the signal's amplitude, still have significant presence when it comes to recognition accuracy. In contrast, when the impulses are comparable to the signal's amplitude (around SNR = 0 dB), the classifier shows better performance. Thus, it is evident that the influence of the noise is much greater than that of the fading. However, as seen from Figs. 4 and 5, there is a considerable degradation in the classification effectiveness when the generalized Gamma fading model is adopted. The performance decline in high SNR levels is not observed in the scenario which combines impulsive noises and generalized Gamma fading for the reference RNN. This can be attributed to the model having less layers and thus, being able to process the particular test dataset more efficiently.

## 6 Conclusions and Future Work

This paper presents a study on the signal recognition capabilities of a hybrid DL framework composed of an AE and a NN classifier. The input data is composed of a large volume of signal vectors in time domain. The test datasets contain signals corrupted with complex generalized channel fading and non-Gaussian impulse noise models. The influence of these impairments on the classification performance is explored and the comparison between the proposed CNN and RNN classifiers and three other reference NNs. On the basis of the results obtained during the simulations there are a number of aspects which pertain to the type of input data and the DL architecture, that are important and may guide further steps in the development of algorithms for MC. When it comes to the input data, an important consideration is how the particular model performs depending on whether the data is normalized or not. A significant difference in recognition accuracy is not observed in the proposed CNN and RNN models but it is a crucial factor in the efficiency of the AE. As for the CNN and RNN classifiers it is shown that the RNN has better performance even though it is not tested on the denoised testset of signals. As for the parameters of each NN model, there are useful guidelines in the recent studies but the need for empirical adjustment during training and testing is ever-present. In view of this fact, the adaptation of the learning rate and batch size during training has real potential.

**Acknowledgment.** The paper is published with the support of the project No BG05M2OP001-2.009-0033 “Promotion of Contemporary Research Through Creation of Scientific and Innovative Environment to Encourage Young Researchers in Technical University - Sofia and The National Railway Infrastructure Company in The Field of Engineering Science and Technology Development” within the Intelligent Growth Science and Education Operational Programme co-funded by the European Structural and Investment Funds of the European Union.

## References

1. Ali, A., Yangyu, F., Liu, S.: Automatic modulation classification of digital modulation signals with stacked autoencoders. Digit. Signal Process. **71**, 108–116 (2017)
2. Arumugam, K.S.K., Kadampot, I.A., Tahmasbi, M., Shah, S., Bloch, M., Pokutta, S.: Modulation recognition using side information and hybrid learning. In: 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), pp. 1–2. IEEE (2017)

3. Bhatti, S.A., et al.: Impulsive noise modelling and prediction of its impact on the performance of WLAN receiver. In: 2009 17th European Signal Processing Conference, pp. 1680–1684. IEEE (2009)
4. Dobre, O.A., Abdi, A., Bar-Ness, Y., Su, W.: Blind modulation classification: a concept whose time has come. In: 2005 IEEE/Sarnoff Symposium on Advances in Wired and Wireless Communication, pp. 223–228. IEEE (2005)
5. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016). http://www.deeplearningbook.org
6. Gouldieff, V., Palicot, J., Daumont, S.: Blind automatic modulation classification in multipath fading channels. In: 2017 22nd International Conference on Digital Signal Processing (DSP), pp. 1–5. IEEE (2017)
7. Gurugopinath, S.: Energy-based bayesian spectrum sensing over $\alpha$-$\mu$/stacy/generalized gamma fading channels. In: 2016 8th International Conference on Communication Systems and Networks (COMSNETS), pp. 1–6. IEEE (2016)
8. Gurugopinath, S., Muralishankar, R., Shankar, H.: Spectrum sensing in the presence of cauchy noise through differential entropy. In: Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), pp. 201–204. IEEE (2016)
9. Hong, D., Zhang, Z., Xu, X.: Automatic modulation classification using recurrent neural networks. In: 2017 3rd IEEE International Conference on Computer and Communications (ICCC), pp. 695–700. IEEE (2017)
10. Ivanov, A., Mihovska, A., Tonchev, K., Poulkov, V.: Real-time adaptive spectrum sensing for cyclostationary and energy detectors. IEEE Aerosp. Electron. Syst. Mag. **33**(5–6), 20–33 (2018). https://doi.org/10.1109/MAES.2018.170098
11. Jang, W.M.: Blind cyclostationary spectrum sensing in cognitive radios. IEEE Commun. Lett. **18**(3), 393–396 (2014)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
13. Lee, J.H., Kim, J., Kim, B., Yoon, D., Choi, J.W.: Robust automatic modulation classification technique for fading channels via deep neural network. Entropy **19**(9) (2017). https://doi.org/10.3390/e19090454, http://www.mdpi.com/1099-4300/19/9/454
14. Li, S., Li, W., Cook, C., Zhu, C., Gao, Y.: Independently recurrent neural network (IndRNN): building a longer and deeper RNN. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5457–5466 (2018)
15. Loshchilov, I., Hutter, F.: Fixing weight decay regularization in adam. arXiv preprint arXiv:1711.05101 (2017)
16. Mendis, G.J., Wei, J., Madanayake, A.: Deep learning-based automated modulation classification for cognitive radio. In: 2016 IEEE International Conference on Communication Systems (ICCS), pp. 1–6. IEEE (2016)
17. Orlic, V.D., Dukic, M.L.: Automatic modulation classification: sixth-order cumulant features as a solution for real-world challenges. In: 2012 20th Telecommunications Forum (TELFOR), pp. 392–399. IEEE (2012)
18. O'Shea, T.J., Corgan, J., Clancy, T.C.: Convolutional radio modulation recognition networks. In: Jayne, C., Iliadis, L. (eds.) EANN 2016. CCIS, vol. 629, pp. 213–226. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44188-7_16
19. O'Shea, T.J., Roy, T., Clancy, T.C.: Over-the-air deep learning based radio signal classification. IEEE J. Sel. Top. Signal Process. **12**(1), 168–179 (2018)
20. Peng, S., Jiang, H., Wang, H., Alwageed, H., Yao, Y.D.: Modulation classification using convolutional neural network based deep learning model. In: 2017 26th Wireless and Optical Communication Conference (WOCC), pp. 1–5. IEEE (2017)

21. Qing Yang, G.: Modulation classification based on extensible neural networks. Math. Probl. Eng. **2017** (2017)
22. Rajendran, S., Meert, W., Giustiniano, D., Lenders, V., Pollin, S.: Distributed deep learning models for wireless signal classification with low-cost spectrum sensors. arXiv preprint arXiv:1707.08908 (2017)
23. Smith, S.L., Kindermans, P.J., Le, Q.V.: Don't decay the learning rate, increase the batch size. arXiv preprint arXiv:1711.00489 (2017)
24. Spaulding, A., Middleton, D.: Optimum reception in an impulsive interference environment - Part I: coherent detection. IEEE Trans. Commun. **25**(9), 910–923 (1977). https://doi.org/10.1109/TCOM.1977.1093943
25. Stevenson, C.R., Chouinard, G., Lei, Z., Hu, W., Shellhammer, S.J., Caldwell, W.: IEEE 802.22: the first cognitive radio wireless regional area network standard. IEEE Commun. Mag. **47**(1), 130–138 (2009)
26. The GNU Radio Foundation: GNU Radio, the free and open software radio ecosystem, October 2018. https://www.gnuradio.org/
27. Tsakalides, P., Nikias, C.L.: Maximum likelihood localization of sources in noise modeled as a stable process. IEEE Trans. Signal Process. **43**(11), 2700–2713 (1995)
28. Xiong, X., Feng, J., Jiang, L.: Automatic digital modulation classification for ors satellite relay communication. In: 2015 International Conference on Wireless Communications & Signal Processing (WCSP), pp. 1–5. IEEE (2015)
29. Xu, J.L., Su, W., Zhou, M.: Likelihood-ratio approaches to automatic modulation classification. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **41**(4), 455–469 (2011)
30. Xu, Y., Li, D., Wang, Z., Guo, Q., Xiang, W.: A deep learning method based on convolutional neural network for automatic modulation classification of wireless signals. Wireless Networks. Springer, New York (2017). https://doi.org/10.1007/s11276-018-1667-6
31. Zhang, D., et al.: Automatic modulation classification based on deep learning for unmanned aerial vehicles. Sensors **18**(3), 924 (2018)
32. Zhang, Z., Hua, Z., Liu, Y.: Modulation classification in multipath fading channels using sixth-order cumulants and stacked convolutional auto-encoders. Wirel. Netw. **11**(6), 910–915 (2017)
33. Zhu, X., Fujii, T.: A novel modulation classification method in cognitive radios using higher-order cumulants and denoising stacked sparse autoencoder. In: 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), pp. 1–5, December 2016. https://doi.org/10.1109/APSIPA.2016.7820860