ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10094907

TOWARDS A ROBUST AND EFFICIENT CLASSIFIER FOR REAL WORLD RADIO SIGNAL MODULATION CLASSIFICATION

Dancheng Liu, Kazim Ergun, Tajana Šimunić Rosing

University of California San Diego
Department of Computer Science and Engineering

ABSTRACT

Automatic modulation classification for radio signals is an important task in many applications, including cognitive radio, radio spectrum monitoring and signal decoding in non-cooperative communications. Recent studies in this area apply various deep learning methods to achieve accurate classification. However, due to the nature of radio signals, distortions during transmission are often unforeseen and unpredictable, which poses a need for robust learning models. At the same time, there is the need for fast real-time modulation classification to meet strict timing requirements. In this work, we propose a lightweight deep learning model that accurately and quickly classifies the modulation of signals having different types of distortions, without the need to be trained using distorted signals. Our model trains 25% faster and classifies 36% faster compared to the state-of-the-art [1], with smaller accuracy degradation on datasets generated using distortion parameters that do not appear in the training set.

***Index Terms*—** artificial intelligence, modulation classification, deep learning

1. INTRODUCTION

Radio signal automatic modulation classification (AMC) is useful for many applications, including radio spectrum monitoring and signal decoding in non-cooperative communications. However, existing work on automatic modulation classification often requires assumptions of the environment, including prior knowledge of the shape and parameters of incoming waves. Without prior knowledge of incoming signals, if the deep learning (DL) architecture can not learn a generalized representation of different modulation types, such architecture is impractical to be used in real life. Also, some modulation classification tasks, e.g., in non-cooperative communications, have strict time constraints, and thus a complex DL model with slower execution speed might not provide services in a timely manner. In this paper, we identify and address the following two practical challenges.

In real world modulation classification, signals are distorted through wireless channels (fading, thermal noise) and other impairment sources (carrier frequency offset, sampling frequency offset). Explicitly training the DL model to be robust to an exhaustive list of all (virtually infinite) possible distortions is infeasible. On the other hand, without acquiring knowledge of these distortions during the training stage, DL models suffer from significant performance degradation. For example, the effects of carrier frequency offset and sampling frequency offset [2], and channel noise and multi-path fading [3] were analyzed by prior work, both resulting in significant degradation in classification accuracy under these conditions. [2] shows that 5% carrier frequency offset leads to around 15% accuracy drop. [3] shows that different channel types bring up to 30% difference in classification accuracy.

Thus, one challenging assumption one should make in developing DL solutions for the AMC task is that the testing dataset is unknown to the model, meaning it does not share the same distortion parameters with the training dataset. However, most existing works fail to address this challenge. The authors of [4, 5, 6, 7, 8, 9, 10] all proposed different models that are trained and tested on the same dataset, making their models vulnerable to producing incorrect classification when facing new signals with unknown distortions. In [11], an additional neural network (NN) module is proposed which can recover signals to their "undistorted" state to boost the performance of modulation classification. Similarly, the authors of [12] proposed a model to correct carrier frequency offsets before classification. However, neither of the works consider signals beyond a single dataset. [13] identified this challenge of generalization, showing that sample length, upsampling factor, and dataset complexity all significantly degrade classification accuracy of existing models. Their solution still requires knowledge on a small portion of the new, unseen signals, which is often impractical to obtain.

On the other hand, the authors of [1, 14, 15] identified the need for real-time modulation classification. The study in [14] claims the attacks on wireless transmission can be avoided by allowing the sender to dynamically switch parameters, including modulation type, without prior agreement with the receiver. Both [14, 15] proposed some hardware-friendly NN model and its hardware implementation, but their

This work was supported in part by PRISM and CoCoSys, centers in JUMP 2.0, an SRC program sponsored by DARPA, and NSF grants #2003279, #1826967, #2100237, #2112167, and #2112665.

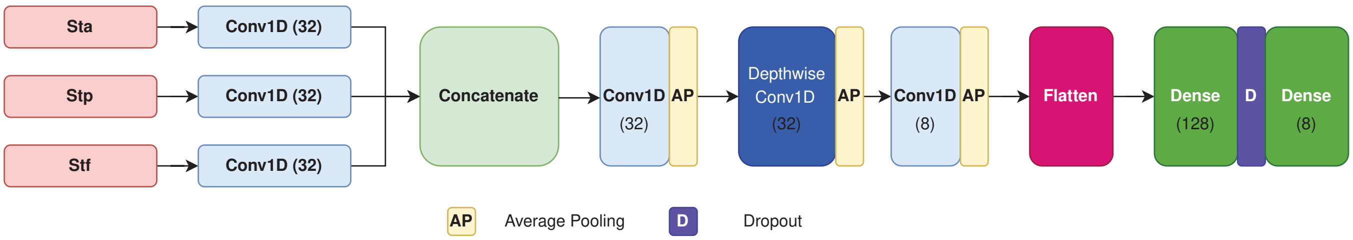

Fig. 1: The detailed view of CNN architecture. The number represents the output channels.

models have low classification accuracy. On the other hand, [1] is the fastest and a recent DL model to run on general purpose computers.

In summary, there are two main challenges in automatic modulation classification tasks that existing works inadequately address. Without the generalization ability to datasets representing unseen distortions during the training stage, a DL model is impractical to be used in real scenarios. Without efficient execution, a DL model can not meet the need of real-time tasks. In this paper, with these two challenges in mind, we come up with a solution that enables robust and quick modulation classification that is practical in real-world scenarios under different types of distortions.

To the best of our knowledge, our work is the first to explore the effects of data transformation on improving the robustness of DL models towards unseen signal samples. We propose *FastMC*, a lightweight DL radio signal modulation classifier that has higher robustness towards unseen distortions compared to existing DL models [1, 5, 4]. *FastMC* transforms raw quadrature (IQ) signal data into phase, angular, and frequency representations. It also uses a depth-wise convolutional layer, for the first time in the AMC domain, to improve inference accuracy on previously unseen distortions. *FastMC* has up to 4.77% higher robustness towards distortions on the test datasets compared to [1]. Additionally, it trains up to 25% faster and classifies 36% faster compared to [1], which is the state-of-the-art method on speed. We also build ten new datasets to simulate a diverse set of real world scenarios, being the most comprehensive set of data used for AMC experiments up to date to the best of our knowledge. The training and testing datasets have been released to the public for future research needs.

2. *FastMC* DESIGN

2.1. Data Transformation

Different from many existing works that use raw IQ data as the input, we identify the need for signal transformation method for automatic modulation classification. With the help of expert knowledge on data transformation, DL networks are able to capture various characteristics of training data that are important for generalization to unseen test data. While unseen distortions change the shape of the incoming radio waves, there are some characteristics that are less affected compared to other ones. Although a very complex network might be able to automatically capture such characteristics, they are computationally expensive. Hence, when a modulation classification task requires real-time classification, it is best to manually identify those less affected characteristics based on expert knowledge for the simpler NN models.

Among the existing transformations, [16] proposed to use cyclic-moment features, and they showed that networks based on those features are less accurate compared to CNNs. [1, 7, 8] proposed to use $r - \theta$ transformation, and they claim that this data transformation can slightly improve the separability between certain classes. However, we have found that existing works all lack the awareness of data transformation's ability to help a simpler model generalize better to data of unseen distortions. In our work, we propose to transform the data from the raw IQ form into its amplitude, phase, and frequency forms. We denote samples of data as S with an in-phase component of S_I and a quadrature component of S_Q, amplitude transformation as $S_{ta} = \sqrt{S_Q^2 + S_I^2}$; phase transformation as $S_{tp} = \arctan(\frac{S_Q}{S_I})$; and frequency transformation as $S_{tf} = \mathcal{F}(IQ)$, where $\mathcal{F}$ is the Fast Fourier Transform of the real part of raw signal. These three transformations aim to explicitly tell a simple DL network about the three types of modulation schemes, which are amplitude-shift-keying (ASK), phase-shift-keying (PSK), and frequency-shift-keying (FSK). In Section 3, we also provide a comparison of *FastMC* with input of transformed data and raw IQ input, and we will show that transformation brings 5.94% less maximum accuracy drop on the test datasets compared to IQ.

2.2. Neural Network Architecture

We design a convolutional neural network (CNN) that functions with the designed transformation as illustrated in Figure 1. Different from classical sequential CNNs that takes in one input of interrelated multiple channels, this assumption of interrelation is not necessarily true with radio signals. Although the three transformed data (S_{ta}, S_{tp}, S_{tf}) could be seen as three channels of input, they might not directly relate to each other, and thus, it is less favorable to apply the same convolutional filter to all of them. Thus, we change the classical

Name	Description	Name	Description
D1	Hypothetical Urban Area	D6	NLOS, UMi Street Canyon, Long Delay Spread
D2	Indoor Office	D7	NLOS, Indoor, Nominal Delay Spread
D3	Outdoor to Indoor and Pedestrian	D8	NLOS, UMa, Very Long Delay Spread
D4	Vehicular and High Antenna	D9	LOS, RMa, Short Delay Spread
D5	Hypothetical Software-Defined Radio	D10	Indoor, Short Delay Spread

Table 1: This table provides a short description of each test dataset that simulates a potential real world communication scenario. The first five datasets (D1-D5) contain data transmitting through multi-path with simpler power-delay-profiles, and the later five (D6-D10) contain data transmitting through multi-path that resembles 5G communications that are more complex.

structures of CNN to fit our needs by splitting the first part of the proposed model into three separated convolutional layers.

As the next step, we concatenate the results of the three convolutional layers and form a classical CNN architecture from there. The convolutional part after concatenation is constructed by two convolutional layers and one depth-wise convolutional layer. Depth-wise convolutional layer is included because it is able to extract useful information with much less parameters. Previous works in other areas reported that compared to traditional convolutional layers, depth-wise convolutional layer is able to learn spatial filters that represent frequency-related information [17], and is useful in multi-domain tasks [18]. Thus, with this depth-wise convolutional layer, *FastMC* is able to capture some characteristics which are useful for the generalization task. Each of the layers uses Rectified Linear Units (ReLU) as the activation function. Additionally, the architecture uses “dropout” as the regularization technique. To be consistent with other models, our proposed architecture uses 50% as the dropout rate. To reduce the parameter size for faster training and classification speed, we apply average pooling after each convolutional layer. After the convolutional layers, the architecture contains two dense layers, with the first one having 128 outputs, and the second one having 8 outputs to match the number of classes in the datasets. Finally, the architecture uses categorical cross-entropy as the loss function.

3. EXPERIMENTS AND EVALUATIONS

3.1. Datasets Simulating Real World Communications

For the base training dataset, we want to consider a dataset without any distortions. However, the widely used *RadioML2016.10a* [16] has some default distortions, namely, moderate carrier frequency offset of 0.01, moderate sampling rate offset of 0.01, and a hypothetical multi-path fading scenario. Since we do not want these distortions in the training dataset, we create our own dataset with the above distortions set to 0 and the hypothetical multi-path fading scenario removed. We create the training dataset with eight digital modulation types: {BPSK, 8PSK, CPFSK, GFSK, PAM4, QAM16, QAM64, QPSK}, with each data being 128 samples long. For each signal-to-noise ratio (SNR) between -20 to 18 with a step size of 2, we generate 1000 data points. There are total 80,000 data points, each having the shape of (128,2).

In addition to the base dataset, we create ten test datasets, in the same dimensions as the training dataset, that represent signals of unforeseen distortions during the training stage. These datasets characterize real-world power-delay-profiles (PDP). Compared to the datasets generated by [16] and [11], these ten datasets comprehensively consider the real-world scenarios of communication including different environments, delay spreads, and movement speeds. The datasets are created in accordance with 3GPP’s technical report on channel models, which many companies refer to for their tests [19]. The simulated events of each dataset are summarized in Table 1. Except for PDP parameters, we generate these datasets with the same parameters as *RadioML2016.10a* [16]. The datasets can be accessed at [1]

3.2. Evaluation Metrics and Experiments Setup

The accuracy of a DL model on a dataset is measured by the maximum accuracy on data of each signal-to-noise-ratio (SNR), which is in the range {-20, 18} with each increment by 2 SNR. This is consistent with the most of the existing works in this area [1, 5, 16]. In this work, we are interested in a DL model’s ability to generalize to unseen data that are not in the distribution of training data, as well as the training and classification speed of the model. Thus, we define the main evaluation metrics to be two parts. First, we evaluate the drop in accuracy for a model’s performance on the unseen test datasets compared to its performance on validation dataset. We also use the maximum drop in accuracy (MD) as an evaluation metric for robustness on all of the datasets. Second, we evaluate the training and classification time of each model.

We compare the proposed *FastMC* to a ResNet-26 based structure (ResNet) [4], a LSTM-CNN model (MCLDNN) [5], a recently proposed pure LSTM-based auto-encoder model (LSTM) [1]. In addition to these baselines, we also evaluate the effect of data transformation by measuring the performance of a modified version of the proposed model taking raw IQ data as input (*FastMC*_IQ). We split the training

[1] https://github.com/DanchengLiu/Modulation_Classification_Dataset

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	MD
MCLDNN [5]	30.0%	0.2%	9.5%	22.3%	30.4%	19.0%	21.2%	14.3%	16.4%	11.1%	30.4%
ResNet [4]	26.8%	0.9%	12.7%	16.0%	26.4%	20.1%	21.6%	14.6%	17.6%	9.8%	26.8%
LSTM [1]	24.2%	0.6%	11.0%	13.2%	25.5%	17.9%	20.5%	15.3%	15.1%	9.7%	25.5%
FastMC_IQ	26.7%	2.1%	13.8%	13.9%	23.2%	21.2%	22.4%	15.4%	18.7%	9.8%	26.7%
FastMC	19.9%	-0.2%	8.8%	10.2%	20.8%	16.9%	18.2%	12.6%	13.7%	7.3%	**20.8%**

Table 2: This table presents drop in the classification accuracy on the test datasets with unseen distortions by each baseline and *FastMC*. The last column MD is the maximum drop in accuracy on all of the ten datasets.

dataset into training and validation set with a 50:50 ratio to be consistent with [16]. We set the maximum epoch of each to be 200, the batch size to be 1024, the early stopping tolerance to be 5 epochs, and the optimizer to be the Adam optimizer with a default learning rate.

We implement each model using the Keras API in Python3. Training and testing are run on Google Colab using NVIDIA T4 as the GPU and Xeon (2.2GHz) as the CPU. The virtual environment has 13GB RAM and around 100 GB space. Since the speed of some models are very fast, to minimize the errors in measurements, when we compare the classification speed of each model, we turn off GPU acceleration and use only CPU.

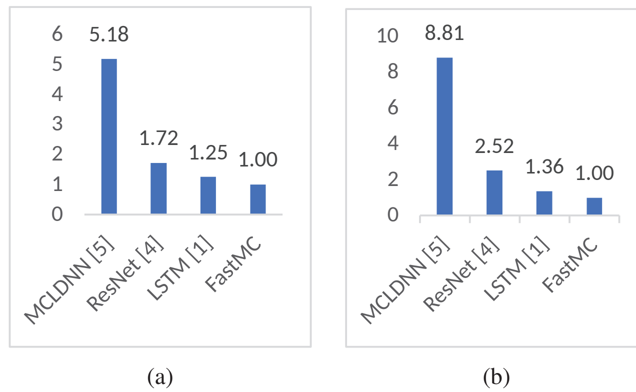

Fig. 2: Normalized (a) training and (b) classification speed of baseline models compared to *FastMC*.

3.3. Classification Accuracy and Accuracy Drop

Table 2 lists the drop in accuracy of each model on each of the test datasets. The last column is the maximum drop in accuracy. As we can see, all DL models experience different levels of performance degradation against unseen signals. Among them, our model is the most robust one with at most 20.76% accuracy drop on the test datasets, being 4.77% better than LSTM [1]. There are three important things needed to be noticed. First, we show that data transformation leads to better generalization to unseen signals by evaluating the performance of our method with raw IQ input denoted as *FastMC*_IQ. It has similar accuracy drops compared to ResNet [4], which is around 6% worse compared to *FastMC*. Second, the most complex architecture, MCLDNN [5] has the best accuracy on the validation dataset. Due to the simplicity of *FastMC*, we think this lower accuracy on validation dataset is reasonable because simpler networks do not have enough parameters to capture all characteristics of each modulation type. Third, the performance of LSTM [1] is slightly better than the other two baseline models. This further validates our claims that data transformation based on expert knowledge is useful for generalization to unseen signals because as we mentioned above, this LSTM [1] uses $r - \theta$ transformation that transforms raw IQ data into amplitude and phase domain.

3.4. Training and Classification Performance

During some real-time communication tasks with resource constraints, a model might need to quickly classify the modulation type of incoming data because the sender might have the need of constantly switching modulation types to avoid wireless attacks [14]. In Figure 2, we compare the time for the four models in 1. training on the training dataset; and 2. classifying the validation dataset. *FastMC* performs the training in 102 seconds on GPU and the classification in 33 seconds on CPU, being 1.25x faster speed in training and 1.36x faster speed in classification compared to the LSTM model [1], which was the previous state-of-the-art in execution speed on general purpose computers.

4. CONCLUSION

In this paper, we present *FastMC*, a novel neural network-based approach to the automatic signal modulation classification task. This model aims to address two important challenges in this area, namely generalization to incoming data of unknown distortions and the need for real-time classification. *FastMC* is a new CNN architecture that combines expert knowledge with neural network, and it shows higher robustness compared to existing models on modulation classification. On the test datasets that represent signals of unseen distortions, *FastMC* achieves a 4.77% lower drop in accuracy compared to LSTM [1] while having 1.25x faster speed in training and 1.36x faster speed in classification.

5. REFERENCES

[1] Ziqi Ke and Haris Vikalo, "Real-time radio modulation classification with an lstm auto-encoder," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 4935–4939.

[2] Steven C Hauser, William C Headley, and Alan J Michaels, "Signal detection effects on deep neural networks utilizing raw iq for modulation classification," in *MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM)*. IEEE, 2017, pp. 121–127.

[3] Lei Li, Qihang Peng, Pamela C Cosman, and Laurence B Milstein, "Deep modulation recognition in an unknown environment," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2019, pp. 1045–1048.

[4] Timothy James O'Shea, Tamoghna Roy, and T Charles 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.

[5] Jialang Xu, Chunbo Luo, Gerard Parr, and Yang Luo, "A spatiotemporal multi-channel learning framework for automatic modulation recognition," *IEEE Wireless Communications Letters*, vol. 9, no. 10, pp. 1629–1632, 2020.

[6] Tianyao Yao, Yuan Chai, Shuai Wang, Xiaqing Miao, and Xiangyuan Bu, "Radio signal automatic modulation classification based on deep learning and expert features," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, 2020, vol. 1, pp. 1225–1230.

[7] Dehua Hong, Zilong Zhang, and Xiaodong Xu, "Automatic modulation classification using recurrent neural networks," in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, 2017, pp. 695–700.

[8] Sreeraj Rajendran, Wannes Meert, Domenico Giustiniano, Vincent Lenders, and Sofie Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.

[9] Rui Ding, Hao Zhang, Fuhui Zhou, Qihui Wu, and Zhu Han, "Data-and-knowledge dual-driven automatic modulation recognition for wireless communication networks," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 1962–1967.

[10] Shahab Hamidi-Rad and Swayambhoo Jain, "Mcformer: A transformer based deep neural network for automatic modulation classification," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 1–6.

[11] Samer Hanna, Chris Dick, and Danijela Cabric, "Signal processing-based deep learning for blind symbol decoding and modulation classification," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 82–96, 2022.

[12] Kumar Yashashwi, Amit Sethi, and Prasanna Chaporkar, "A learnable distortion correction module for modulation recognition," *IEEE Wireless Communications Letters*, vol. 8, no. 1, pp. 77–80, 2018.

[13] Erma Perenda, Sreeraj Rajendran, Gerome Bovet, Sofie Pollin, and Mariya Zheleva, "Learning the unknown: Improving modulation classification performance in unseen scenarios," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 2021, pp. 1–10.

[14] Francesco Restuccia and Tommaso Melodia, "Polymorphic wireless receivers," *Communications of the ACM*, vol. 65, no. 9, pp. 83–91, 2022.

[15] Stephen Tridgell, David Boland, Philip H.W. Leong, Ryan Kastner, Alireza Khodamoradi, and Siddhartha, "Real-time automatic modulation classification using rfsoc," in *2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, 2020, pp. 82–89.

[16] Timothy J O'Shea, Johnathan Corgan, and T Charles Clancy, "Convolutional radio modulation recognition networks," in *International conference on engineering applications of neural networks*. Springer, 2016, pp. 213–226.

[17] Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance, "Eegnet: A compact convolutional neural network for eeg-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, pp. 056013, 2018.

[18] Yunhui Guo, Yandong Li, Liqiang Wang, and Tajana Rosing, "Depthwise convolution is all you need for learning multiple visual domains," 2019, AAAI'19/IAAI'19/EAAI'19, AAAI Press.

[19] "5G; Study on channel model for frequencies from 0.5 to 100 GHz (3GPP TR 38.901) ," https://www.3gpp.org/, [Online].