UNIVERSITY *of* York

This is a repository copy of *Deep Learning-Based Detector for OFDM-IM*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/151899/

Version: Accepted Version

Article:

White Rose
university consortium
Universities of Leeds, Sheffield & York

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

Deep Learning-Based Detector for OFDM-IM

Thien Van Luong, Youngwook Ko, *Senior Member, IEEE,* Ngo Anh Vien, Duy H. N. Nguyen, *Member, IEEE,*
and Michail Matthaiou, *Senior Member, IEEE*

Abstract—This letter presents the first attempt of exploiting deep learning (DL) in the signal detection of orthogonal frequency division multiplexing with index modulation (OFDM-IM) systems. Particularly, we propose a novel DL-based detector termed as DeepIM, which employs a deep neural network with fully-connected layers to recover data bits in an OFDM-IM system. To enhance the performance of DeepIM, the received signal and channel vectors are pre-processed based on the domain knowledge before entering the network. Using datasets collected by simulations, DeepIM is first trained offline to minimize the bit error rate (BER) and then the trained model is deployed for the online signal detection of OFDM-IM. Simulation results show that DeepIM can achieve a near-optimal BER with a lower runtime than existing hand-crafted detectors.

Index Terms—DeepIM, deep learning, deep neural network, index modulation, low-complexity detector, OFDM-IM.

I. INTRODUCTION

Orthogonal frequency division multiplexing with index modulation (OFDM-IM) [1] has emerged as a promising multicarrier system to replace the classical OFDM technology. In OFDM-IM, only a subset of sub-carriers are activated to convey information bits via not only active sub-carriers but also their indices. This enables OFDM-IM to enjoy higher reliability and energy efficiency than OFDM since no extra power or bandwidth resources are required when using indices of active sub-carriers to carry data bits. In addition, OFDM-IM introduces an attractive balance between spectral efficiency and reliability when adjusting the number of active sub-carriers. Due to these benefits, OFDM-IM has received a great deal of research attention recently. For example, frameworks of analyzing the symbol error probability and bit error rate (BER) of OFDM-IM with uncertain channel state information (CSI) were presented in [2] and [3], respectively. Various studies aimed to improve the performance of OFDM-IM can be found in [4]–[6] and the references therein. A low-complexity greedy detector (GD) based on energy detection was proposed in [7], while its BER was analyzed in [8].

Despite having better performance, OFDM-IM suffers from high detection complexity compared to conventional OFDM systems. In fact, current detectors are unable to address this complexity issue while maintaining a near-optimal performance. For instance, maximum likelihood (ML) detection based on exhaustive search achieves optimal BER performance with exponentially high complexity. A low-complexity log-likelihood ratio (LLR) detector was proposed in [1] to achieve a near-ML performance, however, it requires precise knowledge of the receive noise power spectral density. The GD has complexity comparable to that of OFDM detector, however, it suffers from a notable performance loss in comparison with the ML detector. In this context, deep learning (DL) will be exploited to deal with the fundamental drawbacks of hand-crafted detectors.

DL [9] combined with deep neural networks (DNNs) has been successfully applied in various fields, such as computer vision and natural language processing. Recently, it also has been applied to communication systems, especially in physical layer problems. For example, in [10], a DL autoencoder was proposed to replace both the transmitter and receiver of communication systems with DNNs. In OFDM systems, DL also was utilized to reduce the peak-to-average ratio [11] and design a joint channel estimation and signal detection receiver [12]. In [13], an OFDM autoencoder-based system was investigated in presence of hardware impairments. To the best of our knowledge, none of existing works has investigated the potential of applying DL in an OFDM-IM system, especially in reducing its detection complexity.

In this letter, we first propose a DL-based detector for OFDM-IM, termed as DeepIM, which can significantly reduce the complexity over existing hand-designed detectors while maintaining a near-optimal performance. In DeepIM, the received signal and channel information are pre-processed based on the domain knowledge of OFDM-IM prior to being fed to the DNN. Meanwhile, the proposed DNN structure only needs two fully-connected (FC) nonlinear layers to perform the detection task efficiently under Rayleigh fading channels. Moreover, the number of nodes of the hidden layer can be adaptively adjusted to strike a trade-off between the performance and complexity. DeepIM is trained offline to minimize the BER using the simulated data and then the trained model can be used as an online detector with very low runtime. It is shown via simulation results that DeepIM can provide a near-optimal BER under both perfect and imperfect CSI with lower runtime compared with current hand-crafted schemes.

The rest of this paper is organized as follows. Section II describes the system model, while Section III presents the proposed DL-based detector. Simulation results are provided in Section IV. Finally, Section V concludes the paper.[1]

T. V. Luong, N. A. Vien, M. Mathhaiou and Y. Ko are with the ECIT Institute, Queen's University Belfast, Belfast, BT3 9DT, UK, (e-mail: {tluong01, v.ngo, m.matthaiou, y.ko}@qub.ac.uk).

D. H. N. Nguyen is with the Department of Electrical and Computer Engineering, San Diego State University, San Diego, CA, USA 92182 (email: duy.nguyen@sdsu.edu).

[1]*Notation:* Upper-case bold and lower-case bold letters are used for matrices and vectors, respectively; $\mathcal{CN}\left(0, \sigma^2\right)$ represents the complex Gaussian distribution with zero mean and variance σ^2, while $\lfloor . \rfloor$ denotes the floor function and $C\left(.\right)$ stands for the binomial coefficient.

II. System Model

Consider an OFDM-IM system with N_c sub-carriers that are split into G groups of N sub-carriers, i.e., $N_c = NG$. At the transmitter, the signal processing of every OFDM-IM group is the same and independent of each other. Thus, we address only one group for simplicity. Particularly, in every transmission of each group, only K out of N sub-carriers are activated to send a total of p data bits that include $p_1 = K \log_2 M$ bits carried by K complex data symbols and $p_2 = \lfloor \log_2 C(N, K) \rfloor$ bits carried by indices of active sub-carriers, i.e., $p = p_1 + p_2$. Here, M is the M-ary modulation size. Notice that the mapping from p_1 bits to a combination of K active indices can be implemented using either combinatorial methods or a look-up table [1]. As a result, based on p incoming bits, the transmitted vector $\mathbf{x} = [x_1, ..., x_N]$ is formed by assigning K non-zero data symbols to corresponding K active sub-carriers, i.e., x_i is non-zero if sub-carrier i is active and $x_i = 0$ otherwise, $i = 1, ..., N$. We denote such bit-to-symbol mapping by a function $\mathbf{x} = f_{\text{OFDM-IM}}(\mathbf{b})$, where $\mathbf{b}$ stands for a sequence of p incoming bits in one group.

At the receiver, the received signal in the frequency domain is expressed by

$$\mathbf{y} = \mathbf{h} \odot \mathbf{x} + \mathbf{n}, \tag{1}$$

where $\odot$ stands for the element-wise multiplication, $\mathbf{h} = [h_1, ..., h_N]$ is assumed to be the Rayleigh fading channel with $h_i \sim \mathcal{CN}(0, 1)$ and $\mathbf{n}$ represents the additive white Gaussian noise (AWGN) with its entries $n_i \sim \mathcal{CN}(0, \sigma^2)$, $i = 1, ..., N$. The average received signal-to-noise ratio (SNR) is given by $\bar{\gamma} = E_s/\sigma^2$, where E_s is the average energy of a transmitted M-ary symbol.

For signal detection, as discussed in Section I, all existing detection schemes of OFDM-IM such as ML, GD or LLR have drawbacks regarding either high complexity or performance loss. In the next section, DL appears to be an ideal approach to strike the best balance between performance and complexity.

III. Proposed DL-based Detector

In this section, the network structure of the proposed DeepIM detector is first presented and then we describe the training procedure and online deployment of DeepIM.

A. Structure of DeepIM

The structure of the proposed DeepIM is depicted in Fig. 1. Similar to the current detection schemes of OFDM-IM, the channel information is assumed to be known at the receiver. Thus, the received signal $\mathbf{y}$ and the channel $\mathbf{h}$ are considered as the coarse inputs of the DNN. To achieve a better detection performance, $\mathbf{y}$ and $\mathbf{h}$ will be pre-processed based on the domain knowledge of OFDM-IM before entering the DNN model. In particular, firstly, the well-known zero-forcing (ZF) equalizer is employed to get an equalized received signal vector as follows $\bar{\mathbf{y}} = \mathbf{y} \odot \mathbf{h}^{-1}$. Intuitively, this is expected to improve the reconstruction of the M-ary symbols at the active sub-carriers. Secondly, the energy of the received signal, i.e., $\mathbf{y}_m = \left[|y_1|^2, ..., |y_N|^2\right]$ is computed and then combined with $\bar{\mathbf{y}}$ to create the input of the DNN. Notice that in the
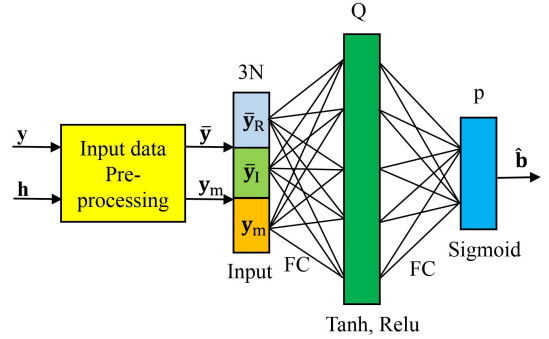

Fig. 1. The structure of the proposed DeepIM detector for OFDM-IM.

GD detector, $\mathbf{y}_m$ is also used to decode indices of active sub-carriers, thus such the proposed pre-processing of the input data based on the domain knowledge of GD can improve the index detection compared to just using the coarse inputs. As shown in Fig. 1, the real and imaginary parts of $\bar{\mathbf{y}}$, i.e., $\bar{\mathbf{y}}_R$ and $\bar{\mathbf{y}}_I$, and the received energy vector $\mathbf{y}_m$ are concatenated to form the $3N$-dimensional input vector $\mathbf{z} = [\bar{\mathbf{y}}_R, \bar{\mathbf{y}}_I, \mathbf{y}_m]$.

The proposed DNN structure consists of two fully-connected (FC) layers including one hidden FC layer of Q nodes and one FC output layer of p nodes as illustrated in Fig. 1. At the hidden layer, either the rectifier linear unit (Relu), $f_{\text{Relu}}(x) = \max(0, x)$, or the hyperbolic tangent (Tanh) function, $f_{\text{Tanh}}(x) = \frac{1-e^{-2x}}{1+e^{-2x}}$, can be used as the activation function. Meanwhile, the Sigmoid function, $f_{\text{Sig}}(x) = \frac{1}{1+e^{-x}}$, is applied at the output layer to output the estimate of the transmitted data bits $\hat{\mathbf{b}}$. In particular, since the output of the Sigmoid lies between 0 and 1, we would decide one element of $\hat{\mathbf{b}}$ either to be bit 0 when its value is less than 0.5 or to be bit 1 otherwise. It is worth noting that the proposed DNN needs only two nonlinear layers to be sufficient for the highly accurate detection of both M-ary and index bits of OFDM-IM. Specifically, let us denote by $\mathbf{W}_1$, $\mathbf{b}_1$ and $\mathbf{W}_2$, $\mathbf{b}_2$ the weights and the biases of the first and second FC layers, respectively. The output vector of the DeepIM model can be expressed as

$$\hat{\mathbf{b}} = f_{\text{Sig}}\left(\mathbf{W}_2 f_{\text{Tanh/Relu}}\left(\mathbf{W}_1 \mathbf{z} + \mathbf{b}_1\right) + \mathbf{b}_2\right). \tag{2}$$

We now highlight some key insights of the proposed DNN structure. First, the input and output lengths of DeepIM are determined by the system parameters such as N, K and M, while the length of the hidden layer, i.e., Q, needs to be properly selected to achieve a desired performance for each system configuration. Intuitively, when the number of transmitted bits p per OFDM-IM group increases, we need Q large enough to guarantee a pre-determined performance. Moreover, just by adjusting Q, we can attain a satisfactory trade-off between the detection accuracy and the model complexity. Another advantage of the proposed DeepIM is that the length of the input vector $\mathbf{z}$ depends on N only, this interestingly makes the complexity of DeepIM less dependent on K and M. In contrast, the complexity of current detectors strongly relies on these parameters. All the aforementioned benefits of DeepIM will be validated by simulation results afterwards.

B. Training Procedure

Before using the proposed DeepIM detector, we need to train offline the DNN model with the data collected from simulations. Particularly, various sequences of p bits $\mathbf{b}$ are randomly generated to obtain a corresponding set of transmitted vectors as $\mathbf{x} = f_{\text{OFDM-IM}}(\mathbf{b})$. These vectors are then sent to the receiver subject to the effects of the Rayleigh fading channel and AWGN noise. Here, the channel and noise vectors are also randomly generated and changed from one bit sequence to another, based on their known statistical models. The collected received signal and channel vectors, i.e., $\mathbf{y}$ and $\mathbf{h}$ are pre-processed as described in the previous section to attain the input dataset $\mathbf{z}$ whose labels are corresponding bit sequences $\mathbf{b}$. Note that the number of training data samples $(\mathbf{z}, \mathbf{b})$ should be large enough to prevent overfitting in the training.

Using the collected data, the DeepIM model is trained to minimize the BER, or equivalently, to minimize the difference between $\mathbf{b}$ and its prediction $\hat{\mathbf{b}}$. Thus, we simply adopt the mean-squared error (MSE) loss function for the training

$$\mathcal{L}\left(\mathbf{b}, \hat{\mathbf{b}}; \theta\right) = \frac{1}{p} \left\| \mathbf{b} - \hat{\mathbf{b}} \right\|^2, \tag{3}$$

where $\theta = \{\mathbf{W}_i, \mathbf{b}_i\}_{i=1,2}$ are the weights and biases of the model. The model parameters θ can be updated for the batches randomly picked up from data samples, using the stochastic gradient descent (SGD) algorithm as follows

$$\theta^+ := \theta - \eta \nabla \mathcal{L}\left(\mathbf{b}, \hat{\mathbf{b}}; \theta\right), \tag{4}$$

where η is the learning rate that defines the step size of the SGD. In our training, we adopt an advanced update algorithm based on the SGD, known as the adaptive moment estimation (Adam) optimizer, which can be easily implemented in various off-the-shelf DL platforms, such as Tensorflow and Keras.

In order to efficiently train DeepIM, the SNR level used for training (denoted as $\bar{\gamma}_{\text{train}}$) must be properly selected since the performance of the model is very sensitive to it. More precisely, it is essential to choose the best $\bar{\gamma}_{\text{train}}$ such that the model trained by this SNR level still performs well for any other SNRs of interest. For instance, if $\bar{\gamma}_{\text{train}}$ is too small, the effect of the noise will not be taken into account in the training, leading to a poor generalization of the obtained model. The details of selecting a specific training SNR for each experiment setting will be subsequently provided in the simulation results.

C. Online Deployment

Once trained offline, the DeepIM model with the optimized parameters θ is utilized for the online deployment of the signal detection of OFDM-IM with arbitrary received SNRs and channels of interest. More specifically, the proposed scheme can be implemented in a real-time manner to estimate the data bits over various channel fading conditions with no extra training for θ. Whenever the received signal and channel information are fed to DeepIM, it will autonomously output the estimated bits in very short computation time. Most importantly, our proposed scheme performs as good as the ML detector under channel estimation errors, though it is trained with perfect CSI, i.e., we do not need to retrain the model.
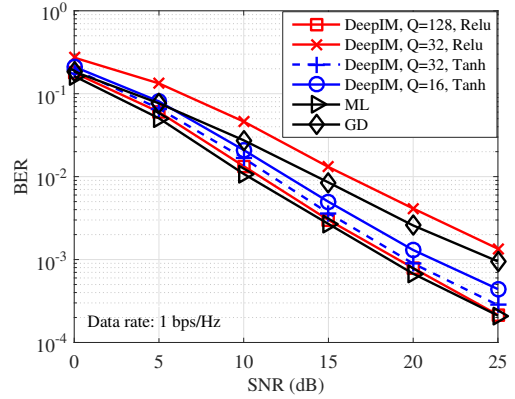

Fig. 2. BER comparison between the proposed DeepIM and the reference detectors under perfect CSI condition and $(N, K, M) = (4, 1, 4)$.

IV. SIMULATION RESULTS

We provide simulation results of the BER and computational complexity of the proposed DeepIM in comparison with existing schemes, including ML and GD detectors. In all considered experiment settings, the proposed DeepIM is trained with 10^3 epochs, each of which contains 20 batches of 10^3 data samples (also known as batch size). Since samples of each batch are randomly generated while training, there are a total of 2×10^4 batches involving in total 2×10^7 different samples for training. The learning rate η is set to 0.001, while a number of other DL parameters, such as Q and $\bar{\gamma}_{\text{train}}$ are selected based on the OFDM-IM parameters in each experiment. Although we consider Rayleigh fading channels in our experiments, the DeepIM detector can be applied to any other channel models.

A. BER Performance

Fig. 2 compares the BER among the proposed, ML and GD detectors, when $(N, K, M) = (4, 1, 4)$ and perfect CSI is assumed. In DeepIM, different values of Q and corresponding activation functions are considered, while $\bar{\gamma}_{\text{train}}$ is set to 10 dB. Particularly, at the hidden layer, for large Q, i.e., $Q = 128$, the Relu is used, while for smaller Q, i.e., $Q = 16, 32$, the Tanh is used. This is due to the fact that the output of the Tanh is not limited to be non-negative as the Relu, thus provides higher model capacity than the Relu. This makes the Tanh more appropriate for DeepIM with small Q, which does not have enough model capacity to perform the detection task. As shown in Fig. 2, DeepIM with $Q = 128$ achieves a BER very close to ML and significantly outperforms GD. When Q decreases to 32 and 16, DeepIM exhibits performance losses of only 1 dB and 2 dB, respectively, compared to ML, while still performs better than GD. Finally, as expected for the same small $Q = 32$, the Tanh has much better BER than the Relu.

The BER comparison for higher data rates, i.e., 2 bps/Hz when $(N, K, M) = (4, 3, 4)$ is illustrated in Fig. 3. Herein, higher data rate means larger p, or equivalently, larger number of classes involved in DeepIM. Hence, the DNN model needs to have higher model capacity than the previous setting; this observation justifies the selection of larger Q in Fig. 3 than that in Fig. 2. Unlike Fig. 2, $\bar{\gamma}_{\text{train}}$ is now set to 15 dB. Again,
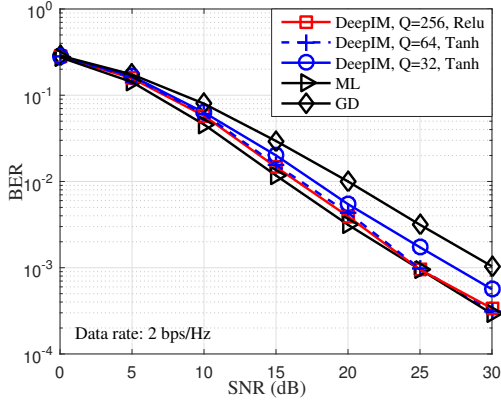
Fig. 3. BER comparison between the proposed DeepIM and the reference detectors under perfect CSI condition and $(N, K, M) = (4, 3, 4)$.
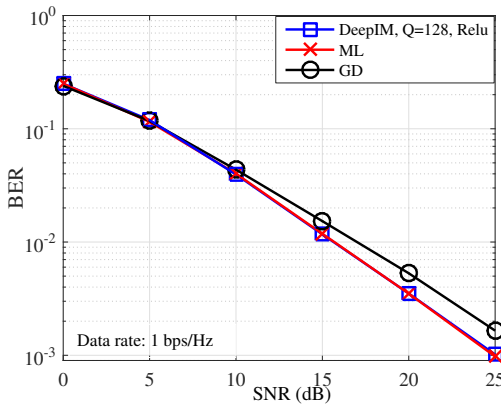

Fig. 4. BER comparison between the proposed DeepIM and the reference detectors under imperfect CSI and $(N, K, M) = (4, 1, 4)$.

there is just a slight performance gap between the proposed detector and ML, while the BER of GD is much worse than our detector. Moreover, as seen from two figures, DeepIM provides an attractive trade-off between performance and complexity when adjusting Q, which is not available in existing detectors.

In Fig. 4, we demonstrate the BER of DeepIM in comparison with competing schemes under CSI uncertainty. In particular, we adopt the minimum mean square error (MMSE)-based variable imperfect CSI model as presented in [2], where the CSI error variance ϵ^2 varies as a function of the average SNR, i.e., $\epsilon^2 = (1 + \bar{\gamma})^{-1}$. It is clear from Fig. 4 that under imperfect CSI, DeepIM achieves a performance identical to ML as it is trained with perfect CSI to have an ability to learn and remember the characteristics of the true channel.

B. Complexity Comparison

To evaluate the complexity of DeepIM, we convert the DeepIM model obtained by training on Tensorflow library to MATLAB to compute the runtime per sample. This is to ensure a fair comparison as the ML and GD detectors are also executed on MATLAB of the same computer. Table I compares the complexity of three detectors in terms of the runtime measured in milliseconds (ms). Herein, DeepIM is assumed to employ $Q = 128$ with the Relu and $Q = 64$ with

TABLE I
COMPLEXITY COMPARISON AMONG DEEPIM, ML, AND GD DETECTORS

(N, K, M)	ML	GD	DeepIM/Relu	DeepIM/Tanh
$(4, 1, 4)$	0.1	0.041	0.01	0.011
$(4, 2, 4)$	0.28	0.053	0.011	0.012
$(4, 3, 4)$	1.6	0.057	0.012	0.013

the Tanh for all cases. As observed in Table I, the runtimes of DeepIM with the Relu and Tanh are quite similar and much less than that of both ML and GD, especially when K gets larger. For example, when $K = 3$, the runtime of DeepIM is about 130 and 5 times less than ML and GD, respectively. As expected in Section III.A, unlike ML, increasing K has a negligible impact on the complexity of DeepIM.

V. CONCLUSION

We have illustrated the effectiveness of the DNN in signal detection of OFDM-IM in terms of both the error performance and complexity. Specifically, the proposed detector called as DeepIM pre-processes received signal and channel vectors based on the domain knowledge such as the ZF equalizer and energy detection before the DNN with fully-connected layers is trained to efficiently recover data bits. It is shown that once trained, DeepIM can be deployed in an online manner with very low runtime while obtaining a near-optimal performance. We believe that such key benefits of DeepIM will inspire future work of applying DL to various advanced IM schemes.

REFERENCES

[1] E. Basar, U. Aygolu, E. Panayirci, and H. V. Poor, "Orthogonal frequency division multiplexing with index modulation," *IEEE Trans. Signal Process.*, vol. 61, no. 22, pp. 5536–5549, Nov. 2013.
[2] T. V. Luong and Y. Ko, "Impact of CSI uncertainty on MCIK-OFDM: tight, closed-form symbol error probability analysis," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1272 – 1279, Feb. 2018.
[3] T. V. Luong, Y. Ko, and J. Choi, "Repeated MCIK-OFDM with enhanced transmit diversity under CSI uncertainty," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4079–4088, June 2018.
[4] M. Wen, B. Ye, E. Basar, Q. Li, and F. Ji, "Enhanced orthogonal frequency division multiplexing with index modulation," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4786 – 4801, July 2017.
[5] J. Choi, "Coded OFDM-IM with transmit diversity," *IEEE Trans. Commun.*, vol. 65, no. 7, pp. 3164–3171, July 2017.
[6] T. V. Luong and Y. Ko, "Spread OFDM-IM with precoding matrix and low-complexity detection designs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 11 619–11 626, Dec. 2018.
[7] J. Crawford and Y. Ko, "Low complexity greedy detection method with generalized multicarrier index keying OFDM," in *Proc. IEEE PIMRC*, Aug. 2015, pp. 688–693.
[8] T. V. Luong and Y. Ko, "A tight bound on BER of MCIK-OFDM with greedy detection and imperfect CSI," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2594 – 2597, Dec. 2017.
[9] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85 – 117, 2015.
[10] 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, Dec. 2017.
[11] M. Kim, W. Lee, and D. Cho, "A novel PAPR reduction scheme for OFDM system based on deep learning," *IEEE Commun. Lett.*, vol. 22, no. 3, pp. 510–513, Mar. 2018.
[12] H. Ye, G. Y. Li, and B. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
[13] A. Felix, S. Cammerer, S. Dorner, J. Hoydis, and S. T. Brink, "OFDM-autoencoder for end-to-end learning of communications systems," in *Proc. IEEE SPAWC*, June 2018, pp. 1–5.