Demodulation of Faded Wireless Signals using Deep Convolutional Neural Networks

Ahmad Saeed Mohammad [1,2], Narsi Reddy [1], Fathima James [1], Cory Beard [1]
[1]School of Computing and Engineering, University of Missouri-Kansas City
Kansas City, Missouri 64110 USA
Email: asm2x9@mail.umkc.edu, sdhy7@mail.umkc.edu, fjmb7@mail.umkc.edu, BeardC@umkc.edu
[2]Collage of Engineering, Mustansiriyah University, Baghdad, Iraq.

***Abstract*—This paper demonstrates exceptional performance of approximately 10.0 dB learning-based gain using the Deep Convolutional Neural Network (DCNN) for demodulation of a Rayleigh-faded wireless data signal. We simulate FSK demodulation over an AWGN Rayleigh fading channel with average signal to noise ratios (SNR) from 10 dB to 20 dB. The most recent and accurate classifier is the Deep Convolutional Neural Network (DCNN) which resulted in the lowest error bit probabilities between 0.00128 to 0.00019 for the range of SNRs. A comparative study has been applied between DCNN and other machine learning classifiers such as Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), Multi-Layer Perceptron (MLP) which give bit error probabilities between 0.021 to 0.002, and Quadratic Discriminant Analysis (QDA) which gives bit error probabilities between 0.027 to 0.003. Frequency-shift keying (FSK) demodulation using matched filtering showed bit error probabilities between 0.025 to 0.0025. We also discuss the complexity issues with the DCNN regarding decoding rates and training set sizes. This work shows how much the DCNN would provide substantial benefit as the demodulator.**

keywords-Faded Signal, Deep Learning, Convolutional Neural Network, Machine Learning, support vector machine, Multi-Layer Perceptron, Linear Discriminant Analysis, and Quadratic Discriminant Analysis.

I. INTRODUCTION

Radio communication models have to accomplish effective transfer of information over a wide variety of communications channels. These channels are not simply AWGN but consist of complex issues with multi-path fading, impulse noise, spurious or continuous jamming, and numerous other complex impairments. Modulation/demodulation schemes of radio signals play an important role in wireless communication for efficient transmission of data with low error rates. To achieve better performance many modulation/demodulation techniques have been crafted over many years.

In this paper, a learning based demodulation technique for binary frequency shift keying (BFSK) has been proposed using deep convolutional neural networks which far outperform other methods with better bit error probability (P_b) compared to other learning based classifiers such as multi-layer perceptron (MLP), support vector machines (SVM), linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) and also a non-learning correlation (CORR) based demodulation method similar to matched filtering. Our results show that DCNN has what we call a ”learning-based gain” of 10dB compared to the CORR method.

This paper is organized as follows: Section III discusses our proposed approaches to demodulating a faded signal with noise using different classifiers. Experimental evaluation processes are presented in Section IV and results are discussed in Section V. Conclusions are drawn in Section VII.

II. PREVIOUS WORK

Recent work for signal modulation and demodulation could be classified into non-learning based methods and learning based methods.

Regarding learning based methods, Oshea el at., [1] created a 128-bit modulation and demodulation technique using DCNN as auto-encoders. Their results show that the implemented technique is applicable for end-to-end learning communication systems for modulation and demodulation applications. Chen el at., [2] proposed a generic framework for signal demodulation using maximum likelihood (ML). The reported results show high SNR approximations which are derived based on the closed-form BER expressions. West el at., [3] implemented radio modulation recognition using machine learning methods with a deep neural network. Their results show the network depth does not affect the modulation recognition, and the improvement should be focus on the learned synchronization and equalization. Albaz tl at., [4] proposed a special Deep Neural Network (DNN), Long Short-Term Memory (LSTM), for FM demodulation that adopts a learning based approach which uses the prior information of a transmitted speech message in the demodulation process. Their reconstruction depended on the usage of memory therefore, their demodulation scheme may fail without powerful memory. Li el at., [5] implemented a FSK signal demodulator depends on Artificial Neural Network (ANN), also their scheme can demodulate different modulation signals such as ASK signal, and PSK signal. Mursel el at., [6] proposed a demodulator for a transmitter signal over unknown channel using neural network; their implementation showed additional complexity over traditional demodulation methods. Oden el. at., [7] proposed a neural network based SDR that took into account multiple channel impairments. The reported results showed a same performance as the correlation receiver for the additive white Gaussian noise (AWGN) channel while Amini

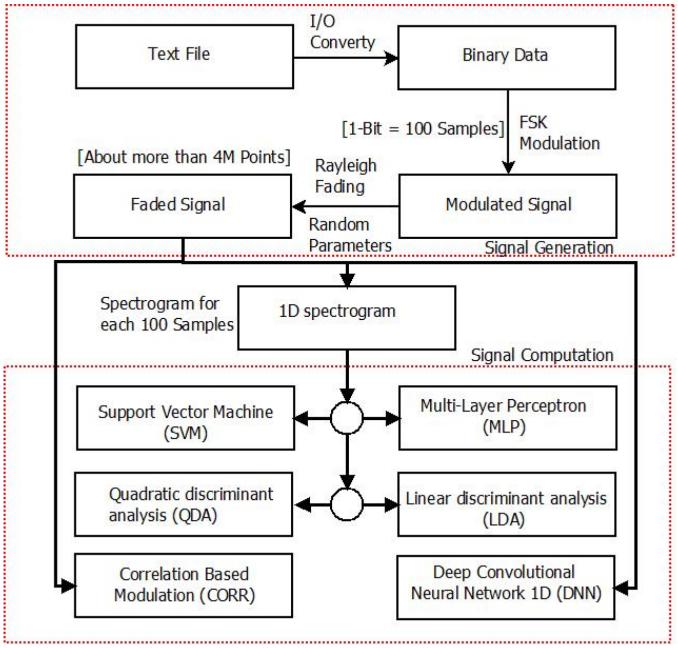

Fig. 1. The block diagram of error detection on faded signal using different classifiers.

et at., [8] implemented a probabilistic neural network for universal demodulator. Their network showed fast training and processing since their network structure has no feedback. Mitra el at., [9] proposed an adaptive Radial Based Function (RBF) neural network for demodulating multi-user communication in a Direct-Sequence Spread-Spectrum Multi-Access (DS-SSMA); their results showed a fast convergence and near optimal performance in communication environment.

As for non-learning methods, Zibar el at., [10] proposed a scalable technique for signal modulation and demodulation using orthogonal frequency-division multiplexing (OFDM). The results evaluated up to 40 Gb/s for wireless signal generation and demodulation. Li el at., [11] implemented a robust demodulation technique against DC offset in direct conversion systems. They tested a high compact 5 to 6 GHz portable radar system. Sambaraju el at., [12] proposed a novel conversion technique for signal demodulation that converts an optical baseband quadrature phase-shift keying (QPSK) signal into a millimeter-wave wireless signal. The reported results show successful demodulation of up to a 16 Gbps QPSK wireless signal in the band of 75 to 110 GHz.

This paper is the first to perform a comprehensive comparison of learning-based and traditional approaches versus DCNN. The results are very substantial.

III. Proposed Evaluation Approach

We evaluate the effectiveness of DCNN versus other approaches using two main stages, generation of test signals and signal demodulation performance analysis as shown in Figure 1.

A. Signal Generation

In the first stage, a randomly generated text file [13], which contains the characters in ASCII code has been converted into 8-bit binary code. There are a total of 5045 characters in the text of 8-bit characters, totaling 40360 bits.

Sampling of 100 sample points per bit with 1 MHz sampling frequency and 75 kHz separation frequency between symbols has been implemented using BFSK. As far a fading wireless channel, a Rayleigh fading channel model is adopted with five multipath delays, gain ranging from -30 dB to 5 dB each, and Doppler shift frequency from 1 Hz to 100 Hz. For modeling receiver noise, additive white Gaussian noise (AWGN) is adopted and noise is added to the entire signal to create an average SNR from 10 dB to 20 dB. For training and validation datasets increments of 1 dB average SNR are used. Testing datasets used increments of 2 dB average SNR.

B. Signal Computation

Three different types of signal computation (signal modulation and demodulation) have been implemented on the prepared signals which are Convolutional neural network (CNN), classical learning based classifiers (such MLP, SVM, LDA, and QDA), and non-learning based Correlation (CORR). The main proposed computation approach is the CNN, which has been compared with the other two computational approaches mentioned above.

The CNNs are a type of feed-forward artificial neural networks which groups neurons in each layer into a convolutional kernel to learn patterns from the input data. The neurons are grouped together; the number of computations and number of parameters to train is reduced compared to multi-layer perceptrons. Stacking a large number of kernels and layers together can create a Deep CNN as shown in Equation (1).

$$Y_L = W_L \circledast X_{L-1} + b_L \tag{1}$$

Except for the convolution of the weight matrix (kernel) W_L with input dataset X_{L-1}, the rest of the process is similar to MLP. The kernels and biases are trained using back propagating the loss (or error) of the predicted outputs and updating the weights using optimization function methods like stochastic gradient descent (SGD) [14], adam [15], etc. For the network used in this paper, random dropout of 25% is used to drop the neurons connected between the current layer and the next layer to reduce the problem of over fitting the network to the training data [16].With advancements in computational hardware, CNN played a major role in advancement of image [17] [18], video [19] [20] [21] and voice [22] [23] recognition applications. Figure 2 shows the convolutional neural network implemented in this paper with two convolutional hidden layers for finding patterns in the input signal followed by three fully connected layers for mapping the generated patterns out of the convolutional layers to the binary classification of 0 or 1.

Now we discuss the learning-based and non-learning based classifiers that were used for comparison. The first learning

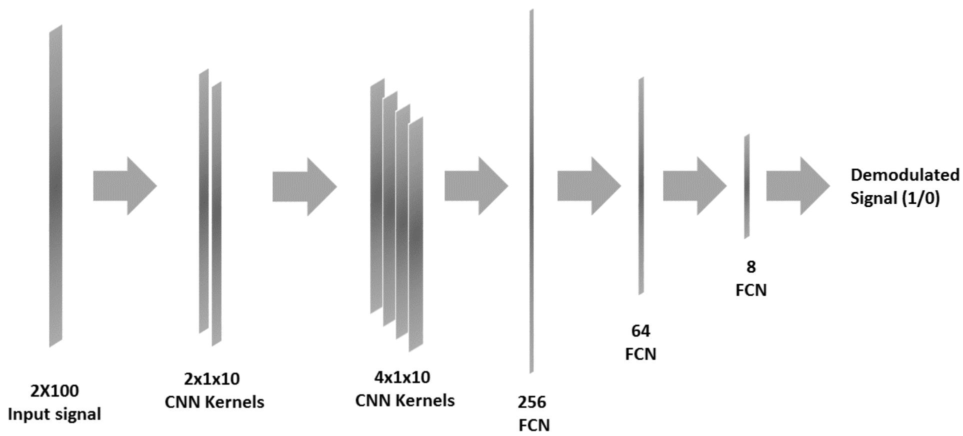

Fig. 2. Convolutional neural network with two convolutional layers followed by four fully connected layers.

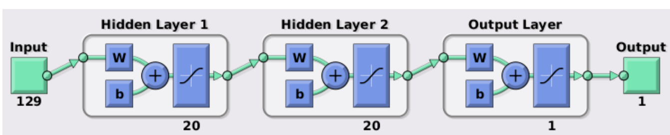

Fig. 3. The architecture of Multi-Layer Perceptron (MLP).

based classifier is MLP. This classifier is used in different applications such as detection [24], and predication [25]. Figure 3 shows the proposed structure of three layers MLP with 2 hidden layers and one output layer. Tangent sigmoid is used as the activation function with mean square error (MSE) as the loss function and adam as the optimization technique.

The SVM is a another supervised classifier that has been implemented. This classifier creates a set of hyper-planes in high-dimensional space to enable a linear dividable separation for the data. Different kernels have been used in SVM to let data be mapped into a high-dimensional space to make the problem data as a linear dividable. These kernels are the polynomial, radial, and sigmoid [26].

The third classifier, LDA which is mainly used for feature extraction, data reduction, and classification. This classifier projects the data in such a way that makes sure a least variance in its direction [27]. Three different kernels have been used with LDA. The first kernel is the linear kernel in which all classes have the same co-variance matrix, while in the second kernel, diagonal-linear, all classes have the same diagonal co-variance. In the third (Pseudo-linear) kernel, all classes have the same co-variance matrix but the kernel will invert the co-variance matrix using pseudo inverse.

The final learning based classifier that has been implement is QDA. This classifier considers a linear combination of class co-variance matrix estimation with poled co-variance matrix estimation and with scaled version of identify matrix to form a QDA classifier [28]. Two kernels have been used with QDA. The first kernel is the diagonal-quadratic; in this case the co-variance matrices are diagonal and can vary among classes while in the second kernel, pseudo-quadratic, the co-variance matrices can vary among classes and the kernel will invert the co-variance using pseudo inverse.

A non-learning based demodulation CORR method has been implemented to compare the performance of all the learning classifier models. In this method, the input signal at the receiver is correlated with the reference signals of bit-1 and bit-0. Then based on the correlation strength the received signal is decoded as either bit 1 or 0. This is identical to the approach of the matched filter receiver, which is the optimal signal-based non-learning technique in the presence of only AWGN.

IV. Experimental Evaluation

The training and validation datasets have been prepared as follows: for each average SNR case, 40360 bits are faded three different ways to generate variations in Doppler frequency and multipath delays. Therefore, there are 11 (avg SNR's)$*3$ (iterations) $*40360$ (bits per iteration)$= 1331880$ bits in total generated for training and validation datasets respectively. In case of a testing dataset, instead of 1dB increments (10 dB to 20 dB) of average SNR, 2 dB increments are used, so a total of 6(avg SNR's)$*$3(iterations) $*$40360(bits per iteration)$=$ 726480 bits are generated.

Except for the CNN based classifier, 1D spectrograms are extracted as a feature vector. In the case of CNNs, except for input normalization of the signal between $[-1, 1]$, raw signals (real and imaginary parts) are given as the inputs. This is to show that unlike other classical classifiers, CNN's are capable of extracting required features for better classification.

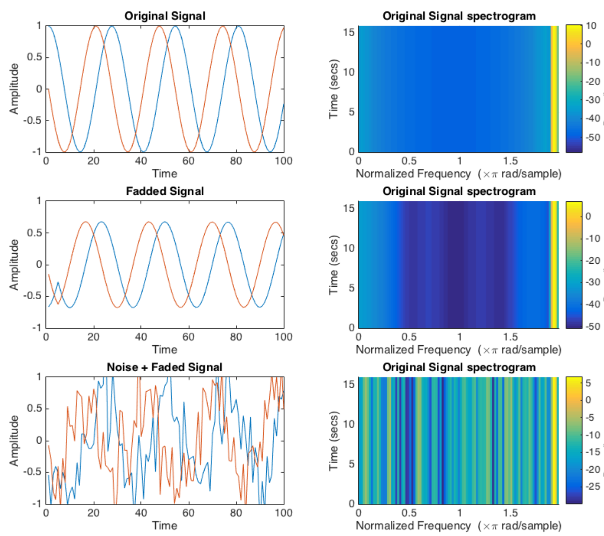

Fig. 4. The modulated signal and its spectrogram on first row.

Figure 4 shows the modulated signal and its spectrogram on the first row, followed by the faded signal and noisy faded signal.

In the second step, all of the classifiers' parameter tuning and training is done with training and validation datasets. The results of the performance of the classifiers is tested on the test dataset which remains untouched during the training process. The test dataset by itself is used to generate the results for the correlation based method (CORR) so that these results can be used for learning vs non-learning comparison.

V. Results

Figure 5 shows the bit error probability (P_b) obtained for all learning based methods along with the correlation based method. From the figure, it can be seen that there is a linear relationship between the bit error probability and average SNR on a log-log (actually log-dB) plot as is expected for Rayleigh fading.

It can also be seen that DCNN out performs all the other methods with a minimum 10 fold increase in performance. i.e., the same P_b of DCNN at 10 dB is achieved by the correlation based method (CORR) at 19.99 dB. We call this difference the "learning-based gain," similar to the use of terms like coding gain or antenna gain. Figure 6 shows DCNN gain over other methods for the same bit error probability. This is a tremendous result, especially as one considers that benefits of 3 or 4 dB are considered significant when searching for better channel coding schemes.

Table I summarizes the P_b of all suggested methods for the demodulation of faded signals with respect to bit error probability.

As mentioned before, different classical classifiers have been used in this work to demodulate the noisy faded signal.

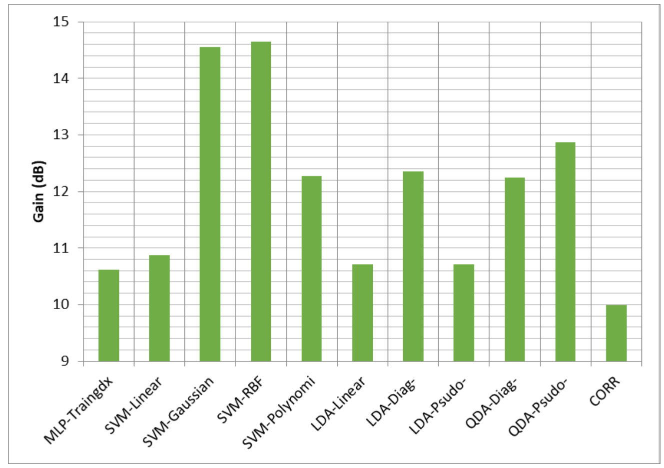

Fig. 6. Learning-based gain of DCNN over all other methods

Figure 5 shows the performance of MLP with respect to bit error probability for different average SNR. The second classical classifier that has been used for demodulation is the SVM with four different kernels (linear, Gaussian, RBF, and polynomial). The best performance is achieved with linear kernel compared to other support vector machine (SVM) kernels as shown in Figure 5.

Figure 5 shows the performance of LDA with three different kernels (linear, diagonal-linear, and pseudo-linear). In this test LDA-Linear and LDA-Pseudo-Linear are identical, and both show the highest performance over the LDA-Linear-Diagonal. The last classical classifier that has been used to demodulate the noisy faded signal is QDA. Figure 5 shows the performance of two different kernels of QDA with respect of bit error probability. Figure 5 also shows the performance of correlation based modulation with respect to bit error probability as was also seen in Figure 5; this method outperforms all learning-based methods, except of course the DCNN.

A comparative study has been established between the DCNN and other classical machine learning classifiers as shown in Figure 7. This figure shows how much lower the DCNN bit error probability compares with the correlation method and the other classical classifiers. We can also see that without DCNN, the CORR method is not improved upon by any of the other classifiers.

VI. DCNN Computational Complexity

Even given the tremendous benefits seen with DCNNs in this paper, a major concern about DCNNs might be that their computational complexity is prohibitive. But this is not the case as discussed now.

A. DCNN Computation Statistics

The deep learning model adopted in this paper has a total of six layers with the first two layers being convolutional layers for finding patterns, followed by three multi-layer perceptron layers as hidden layers and with one MLP layer as output. There are total of 31.77 Mflops per one bit prediction,

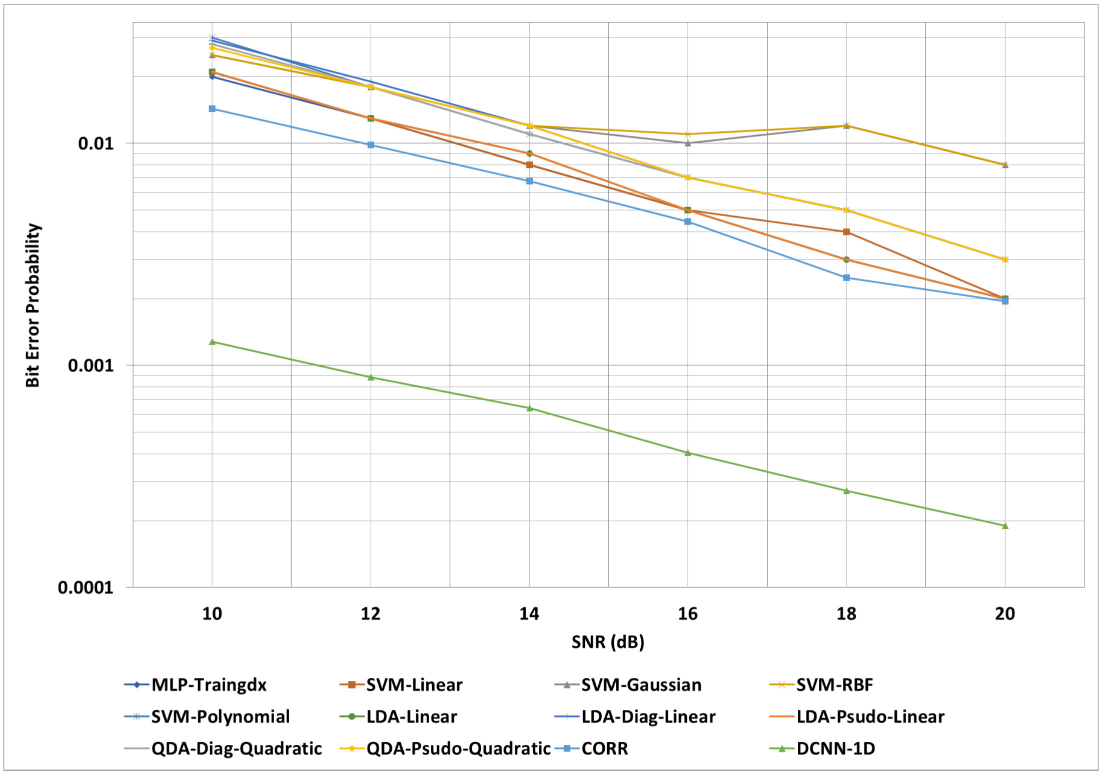

Fig. 5. The graph of all classifiers performance.

TABLE I
OVERALL BIT ERROR PROBABILITY OF THE PROPOSED METHOD FOR MODULATION OF FADED SIGNAL.

SNR (dB)	MLP Train-dgx.	SVM Linear	SVM Gauss	SVM RBF	SVM Poly.	LDA Linear	LDA Diag-Linear	LDA Pseudo-Linear	QDA Diag-Quad	QDA Pseudo-Quad	CORR	DCNN Adam
10	0.020	0.021	0.025	0.025	0.030	0.021	0.029	0.021	0.028	0.027	0.0140	0.00128
12	0.013	0.013	0.018	0.018	0.018	0.013	0.019	0.013	0.018	0.018	0.0098	0.00088
14	0.008	0.008	0.012	0.012	0.011	0.009	0.012	0.009	0.011	0.012	0.0067	0.00064
16	0.005	0.005	0.010	0.011	0.007	0.005	0.007	0.005	0.007	0.007	0.0045	0.00040
18	0.003	0.004	0.012	0.012	0.005	0.003	0.005	0.003	0.005	0.005	0.0025	0.00027
20	0.002	0.002	0.008	0.008	0.003	0.002	0.003	0.002	0.003	0.003	0.0020	0.00019

refer table II. However, this number is pure operations per second without any optimization. For example, all the tests are performed on NVIDIA's GT 840m GPU with 1.5 GB memory with peak throughput of 790.3 Gflops. This means that theoretically the proposed model is capable of predicting 25.5 kbps. But with the help of batch processing and NVIDIA's cuDNN framework, the same hardware is capable of predicting 123.33 kbps, which is an almost 5 fold increment in performance.

In the case of FPGA chips like the Xilinx KU060 where the peak throughput of 6.55 Tflops is possible with same power consumption as NVIDIA's GT 840m GPU (40 watts), the raw bit predicting power is 206.17 kbps. Using techniques such as weight pruning and weight sharing techniques, the number of operations needed to predict can be reduced drastically and along with batch prediction the number of bits per sec can be increased 10 fold to 100 fold. Note also that OFDM demodulation does not use high data rates per subcarrier.

TABLE II
DCNN THEORETICAL PERFORMANCE FOR EACH LAYER

Layer	**Weights**	**Flops**
Conv1	42	7.64K
Conv2	84	13.78K
MLP1	84224	27.54M
MLP2	16448	4.19M
MLP3	520	32.79K
MLP4	18	136
Total	**101336**	**31.79M**

B. DCNN Training Requirements

One might assume that the DCNN would need to be constantly retrained to adapt to the new wireless channel at virtually every new square meter of location. But this is not true and highlights the value of the DCNN.

We considered two scenarios. In the first scenario, the

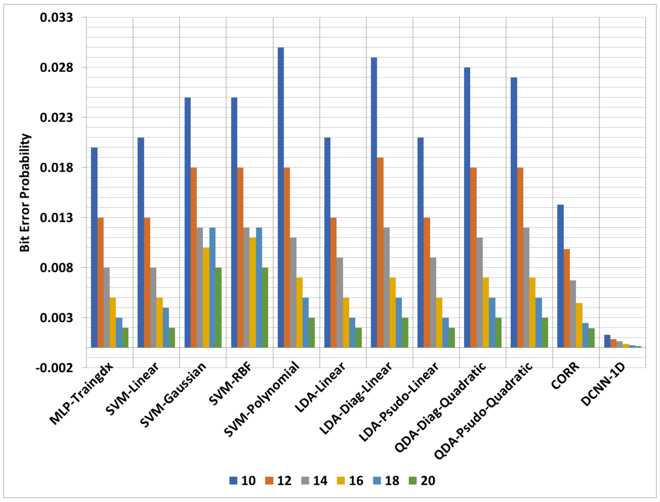

Fig. 7. Demodulation performance of DCNN and other used methods.

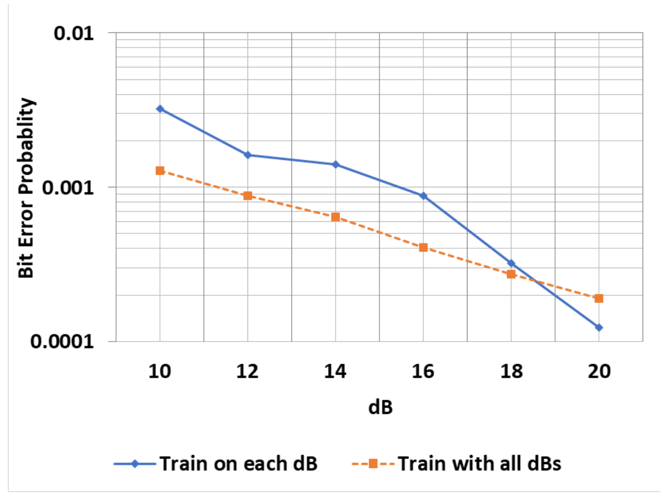

Fig. 8. The comparison of P_b for two DCNN implementation scenarios.

DCNN is trained for each average signal to noise ratio (SNR) in increments of 1 dB from 10 to 20 dB. These weights could be saved for each trained DCNN to be used for the mobile device when it moves from one point with a certain SNR to another point with different SNR. Training would not need to occur on the mobile device, but could either be performed offline or by a base station that connects with the mobile device. In the second scenario, the DCNN is trained only once for all values of average SNR. The training set would include signals that have all values of SNR from 10 to 20 dB. With these trained weights, there would be no need to change the weights of the trained model when the mobile device moves from point to point. This would be very useful, but one might assume there would be a loss of performance quality of the DCNN.

Figure 8 shows the DCNN P_b values when trained with all values of average signal to noise ratio (SNR) compared to the DCNN trained with each individual signal to noise ratio (SNR). It is still true that both scenarios of DCNN implementation have better performance compared to the other learning and non-learning based methods. But for comparison, the second scenario performs $better$ than the first scenario in all the cases except for average SNR greater than 19 dB! The reason for this is because the dataset used for training the second scenario has a wide range of diversity of fading and noise patterns compared to that of the dataset used in training the first scenario. This actually helps DCNN performance instead of hurting it. Note that in the training we also used a variety of Doppler spread values.

VII. Conclusion

The deep convolutional neural network (DCNN) is by far the best suggested learning based method to demodulate the noisy faded signal. This classifier shows the lowest bit error probability and by what we have called a "learning-based gain" of at least 10 dB over all other demodulation methods. We compared DCNN with standard modulation methods like the correlation based demodulation (CORR), and classical machine learning classifiers namely, multi-layer perceptron (MLP), support vector machine (SVM), linear discriminant analysis (LDA), and quadratic discriminant analysis (QDA) with different kernels for each classifiers.

The DCNN shows learning-based gain of at least 10 dB over other demodulation and classifiers. And DCNN is reasonable and possible to deploy in a wireless network. Several options exist for processors and chips to implement the neural networks. Mobile devices do not need to train their DCNNs, but rather can use a set of weights from training elsewhere. And the training process is most effective when signals with a variety of Doppler shifts and SNRs are used. This produces a set of weights that are usable in a wide variety of locations and that actually perform better than if the DCNN were trained for a specific location or SNR. We have shown in this paper that deep learning convolutional neural networks provide a new opportunity to substantially increase the performance of mobile communication.

References

[1] T. J. O'Shea, K. Karra, and T. C. Clancy, "Learning to Communicate: Channel Auto-encoders, Domain Specific Regularizers, and Attention," 2016. [Online]. Available: http://arxiv.org/abs/1608.06409

[2] D. Chen and J. N. Laneman, "Modulation and demodulation for cooperative diversity in wireless systems," *IEEE Trans. Wireless Commun.*, vol. 5, no. 7, pp. 1785–1794, 2006.

[3] T. West, Nathan E.O'Shea, "Deep architectures for modulation recognition," *2017 IEEE International Symposium on Dynamic Spectrum Access Networks, DySPAN 2017*, 2017.

[4] D. Elbaz and M. Zibulevsky, "End to End Deep Neural Network Frequency Demodulation of Speech Signals," *arXiv preprint*, pp. 1–6, 2017. [Online]. Available: https://arxiv.org/pdf/1704.02046.pdf

[5] M. Li and H. Zhong, "Neural Network Demodulator for Frequency Shift Keying," *2008 International Conference on Computer Science and Software Engineering*, pp. 843–846, 2008. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4722750

[6] O. Mursel, A. Akan, and H. Dogan, "Neural Network Based Receiver Design for Software Defined Radio over Unknown Channels," *8th International Conference on Electrical and Electronics Engineering (ELECO)*, pp. 297–300, 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6713848/

[7] M. Önder, A. Akan, and H. Doan, “Advanced neural network receiver design to combat multiple channel impairments,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 24, no. 4, pp. 3066–3077, 2016.

[8] M. Amini and E. Balarastaghi, “Universal Neural Network Demodulator for Software Defined Radio,” *International Journal of Machine Learning and Computing*, vol. Vol. 1, no. No. 3, pp. 305–310, 2011. [Online]. Available: http://www.ijmlc.org/papers/45-L335.pdf

[9] U. Mitra and H. V. Poor, “Neural network techniques for multi-user demodulation,” *IEEE International Conference on Neural Networks - Conference Proceedings*, vol. 1993-January, no. March, pp. 1538–1543, 1993.

[10] D. Zibar, R. Sambaraju, A. Caballero, J. Herrera, U. Westergren, A. Walber, J. B. Jensen, J. Marti, and I. T. Monroy, “High-capacity wireless signal generation and demodulation in 75- to 110-GHz band employing all-optical OFDM,” *IEEE Photonics Technology Letters*, vol. 23, no. 12, pp. 810–812, 2011.

[11] C. Li and J. Lin, “Complex signal demodulation and random body movement cancellation techniques for non-contact vital sign detection,” *IEEE MTT-S International Microwave Symposium Digest*, pp. 567–570, 2008.

[12] R. Sambaraju, D. Zibar, A. Caballero, I. T. Monroy, R. Alemany, and J. Herrera, “100-GHz Wireless-Over-Fiber Links With Up to 16-Gb / s QPSK Modulation Using Optical Heterodyne Generation and Digital Coherent Detection,” *IEE Photonics Technology Letters*, vol. 22, no. 22, pp. 1650–1652, 2010.

[13] V. Bibakis. Random text generator. [Online]. Available: http://randomtextgenerator.com/

[14] L. Bottou, “Large-scale machine learning with stochastic gradient descent,” in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[15] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[16] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting.” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[17] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[18] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Deepface: Closing the gap to human-level performance in face verification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[19] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description,” *CoRR*, vol. abs/1411.4389, 2014. [Online]. Available: http://arxiv.org/abs/1411.4389

[20] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. J. Mooney, and K. Saenko, “Translating videos to natural language using deep recurrent neural networks,” *CoRR*, vol. abs/1412.4729, 2014. [Online]. Available: http://arxiv.org/abs/1412.4729

[21] H. Nam and B. Han, “Learning multi-domain convolutional neural networks for visual tracking,” *CoRR*, vol. abs/1510.07945, 2015. [Online]. Available: http://arxiv.org/abs/1510.07945

[22] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, “End-to-end attention-based large vocabulary speech recognition,” *CoRR*, vol. abs/1508.04395, 2015. [Online]. Available: http://arxiv.org/abs/1508.04395

[23] A. Graves, A. Mohamed, and G. E. Hinton, “Speech recognition with deep recurrent neural networks,” *CoRR*, vol. abs/1303.5778, 2013. [Online]. Available: http://arxiv.org/abs/1303.5778

[24] Mohammad A Rattani A and D. R, “Eyeglasses Detection based on Learning and Non-learning based Classification Schemes,” *IEEE International Symposium on Technologies for Homeland Security, 2017*, p. (accepted), 2017.

[25] C.-F. Tsai and J.-W. Wu, “Using neural network ensembles for bankruptcy prediction and credit scoring,” *Expert Systems with Applications*, vol. 34, no. 4, pp. 2639 – 2649, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417407001558

[26] P. Wang, “SVM-Based Definition of Trust in Multi-agent System,” vol. 9, pp. 909–912, 2012.

[27] M. Welling, “Fisher Linear Discriminant Analysis,” *Science*, vol. 1, no. 2, pp. 1–3, 2009. [Online]. Available: http://www.cs.huji.ac.il/{~}csip/Fisher-LDA.pdf

[28] S. Srivastava, M. Gupta, and B. Frigyik, “Bayesian quadratic discriminant analysis,” *Journal of Machine Learning Research*, vol. 8, pp. 1277–1305, 2007.