Deep Learning-Based Channel Estimation

Mehran Soltani, Vahid Pourahmadi, Ali Mirzaei, Hamid Sheikhzadeh

***Abstract*—In this paper, we present a deep learning (DL) algorithm for channel estimation in communication systems. We consider the time-frequency response of a fast fading communication channel as a two-dimensional image. The aim is to find the unknown values of the channel response using some known values at the pilot locations. To this end, a general pipeline using deep image processing techniques, image super-resolution (SR) and image restoration (IR) is proposed. This scheme considers the pilot values, altogether, as a low-resolution image and uses an SR network cascaded with a denoising IR network to estimate the channel. Moreover, an implementation of the proposed pipeline is presented. The estimation error shows that the presented algorithm is comparable to the minimum mean square error (MMSE) with full knowledge of the channel statistics and it is better than ALMMSE (an approximation to linear MMSE). The results confirm that this pipeline can be used efficiently in channel estimation.**

***Index Terms*—Channel estimation, Deep Learning, Image Super-resolution, Image restoration**

I. Introduction

Orthogonal frequency-division multiplexing (OFDM) is a modulation method that has been widely used in communication systems to address frequency-selective fading in wireless channels. In a communication channel, the received signal is usually distorted by channel characteristics. In order to recover the transmitted symbols, the channel effect must be estimated and compensated at the receiver. Generally, the receiver estimates the channel using some symbols named pilots which their positions and values in time-frequency are known to both transmitter and receiver. Depending on these pilot arrangements, three different structures can be considered: block-type, comb-type and lattice-type [1]. In the block-type arrangement, pilots are transmitted periodically at the beginning of an OFDM block at all subcarriers while in comb-type, the pilots are present in few subcarriers of few OFDM symbols. In the lattice-type arrangement, pilots are inserted along both time and frequency axes with given periods in a diamond-shaped constellation.

The conventional pilot-based estimation methods, i.e., Least Square (LS) and Minimum Mean Square Error (MMSE) utilize the pilot values in time-frequency grids to find the unknown values of the channel response. These algorithms have been optimized in various conditions [2]. In contrast to the LS estimation which requires no information about the statistics of the channel, the MMSE estimation results in a better performance by utilizing the statistics of the channel and noise variance. To use the MMSE in practical scenarios, some approaches are presented which reduce the complexity of this scheme and use an estimation of the channel statistics instead of the exact information. In [3], an approximated linear version of the MMSE (ALMMSE), is proposed in fast fading channels which its complexity is much less than the original MMSE due to reducing the size of the correlation and the filtering matrix.

Recently, Deep Learning (DL) has gained much attention in communication systems. In DL-based communication systems, some approaches have been proposed to enhance the performance of different conventional algorithms including modulation recognition [4], signal detection [5], channel equalization [6], channel state information (CSI) feedback [7] and channel estimation [8], [9]. In [8], the communication system is considered as a black-box and an end-to-end DL architecture is used for signal transmission/reception. Encoding, decoding, channel estimation and all other functionalities of a communication link are embedded in the DL-block, implicitly. More specifically, this method is not able to explicitly find the channel time-frequency response and so not effective for applications which need to have the complete channel response. In [9], the channel matrix is considered as an image and then used a denoising network for channel estimation. This work focuses on the channel matrix along the transmitter/receiver antenna space (in multiple antenna scenario) and is not discussing the time-frequency response of the each Tx/Rx link.

Motivated by this, in this paper, we present a DL-based framework for channel estimation in OFDM systems. In this method, the time-frequency grid of the channel response is modeled as a 2D-image which is known only at the pilot positions. This channel grid with several pilots is considered as a low-resolution (LR) image and the estimated channel as a high-resolution (HR) one. A two-phase approach is presented to estimate the channel grid. First, an image super-resolution (SR) algorithm is used to enhance the resolution of the LR input. Afterwards, an image restoration (IR) method is utilized to remove the noise effects. For SR and IR networks, we have used two recently developed CNN-based (Convolutional Neural Network) algorithms, SRCNN [10] and DnCNN [11], respectively. The contributions of this paper are summarized as follows:

1) Model the channel time-frequency response as an image.
2) Consider the channel response in the pilot positions as a LR image and the estimated channel response as the proposed HR image.
3) Use DL-based image super-resolution and image denoising techniques to estimate the channel.

The remainder of the paper is organized as follows. Section II provides a brief survey of the channel estimation with conventional methods. Section III presents the structure of the proposed DL-base channel estimator. In section IV, simulation results are presented and finally section V concludes the paper.

II. Background

A. Channel Estimation

In an OFDM system, for the kth time slot and the ith subcarrier, the input-output relationship is represented as:

$$Y_{i,k} = H_{i,k} X_{i,k} + Z_{i,k}. \tag{1}$$

Considering an OFDM subframe of size $N_S \times N_D$, time slot index k is between $[0, N_D - 1]$ and the range of the subcarrier index i is $[0, N_S - 1]$. In (1), $Y_{i,k}$, $X_{i,k}$, and $Z_{i,k}$ are the received signal, transmitted OFDM symbol and white Gaussian noise, respectively. $H_{i,k}$ is the (i,k) element of $\mathbf{H} \in \mathbb{C}^{N_S \times N_D}$. $\mathbf{H}$ represents time-frequency response of the channel for all subcarriers and time slots.

To estimate the channel, specifically in the channels with fading, the time domain response is represented as $\mathbf{H} = \{\mathbf{h}[1], \mathbf{h}[2], ..., \mathbf{h}[N_D]\}$, where each $\mathbf{h}[k]$ is the channel frequency response at the kth time slot.

The LS method estimates the channel at the pilot positions. If we consider the LS estimated channel as a diagonal matrix $\mathbf{H}_p^{\text{LS}} \in \mathbb{C}^{N_P \times N_P}$, $\mathbf{H}_p^{\text{LS}}$ can be estimated by solving:

$$\hat{\mathbf{H}}_p^{\text{LS}} = \underset{\mathbf{H}_p}{\arg\min} \|\mathbf{y}_p - \mathbf{H}_p \mathbf{x}_p\|_2^2, \tag{2}$$

where $||.||_2$ is the $\ell 2$ distance and $\hat{\mathbf{H}}_p^{\text{LS}} \in \mathbb{C}^{N_P \times N_P}$ is the estimated diagonal matrix. $\mathbf{x}_p$ contains the known pilot values and $\mathbf{y}_p$ is the corresponding observations. The optimization of (2) results in $\hat{\mathbf{h}}_p^{\text{LS}} = \text{diag}(\hat{\mathbf{H}}_p^{\text{LS}}) = \mathbf{y}_p / \mathbf{x}_p$. To find the channel value at the points other than the pilot positions, we have to apply a two dimensional interpolation method.

A better choice than LS, is MMSE estimator which is obtained by multiplying the LS estimates at the pilot-symbol positions with a filtering matrix $\mathbf{A}_{\text{MMSE}} \in \mathbb{C}^{N_L \times N_P}$ [12]:

$$\hat{\mathbf{h}}_d^{\text{MMSE}} = \mathbf{A}_{\text{MMSE}} \hat{\mathbf{h}}_p^{\text{LS}}, \tag{3}$$

where $\hat{\mathbf{h}}_d^{\text{MMSE}} \in \mathbb{C}^{N_L \times 1}$ $(N_L = N_S \times N_D)$ is the vectorized MMSE estimation of the channel response $\mathbf{H}$ at subframe d. To find the filtering matrix, the mean square error(MSE),

$$\epsilon = \mathbb{E}\{\|\mathbf{h}_d - \mathbf{A}_{\text{MMSE}} \hat{\mathbf{h}}_p^{\text{LS}}\|_2^2\}, \tag{4}$$

has to be minimized. Minimizing (4) leads to

$$\mathbf{A}_{\text{MMSE}} = \mathbf{R}_{\mathbf{h}_d \mathbf{h}_p} (\mathbf{R}_{\mathbf{h}_p \mathbf{h}_p} + \sigma_n^2 (\mathbf{x}\mathbf{x}^{\text{H}})^{-1})^{-1}, \tag{5}$$

where the matrix $\mathbf{R}_{\mathbf{h}_d \mathbf{h}_p} = \mathbb{E}\{\mathbf{h}_d \mathbf{h}_p^{\text{H}}\}$ denotes the channel correlation matrix between desired subframe and pilot-symbols and the matrix $\mathbf{R}_{\mathbf{h}_p \mathbf{h}_p} = \mathbb{E}\{\mathbf{h}_p \mathbf{h}_p^{\text{H}}\}$ is the channel correlation matrix at the pilot-symbols. It is obvious that the MMSE will be useful only if the correlation matrix of the channel, denoted as $\mathbf{R}$, is completely known.

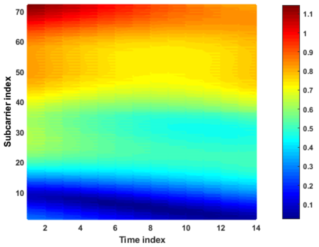

Fig. 1. An example of normalized real/imaginary 2D-image for a sample channel time-frequency grid

B. Super-resolution and Image restoration

Considering a low resolution and noisy image, several techniques have been proposed to reproduce the higher resolution and less noisy image. Image super-resolution (SR) is a class of techniques used for resolution enhancement in images. DL-based algorithms, especially with deeply and fully convolutional networks, have achieved high performance in the problem of recovering the HR images from the LR image inputs. Recently, Super-resolution convolutional neural network (SRCNN) [10] is proposed to map between LR/HR images in an end-to-end manner. Other than SR techniques, image restoration (IR) algorithms can be applied to remove/reduce the noise effect on an image. Various models have been presented for IR in the literature. For instance, in [11], a feed-forward denoising convolutional neural network (DnCNN) scheme is presented which has utilized the residual learning and batch normalization to speed up the training process.

III. ChannelNet

A. Channel Image

In this work, we focus on one link between a pair of Tx and Rx antennas, i.e., we have Single-input, Single-output (SISO) communication link. For this link, the channel time-frequency response matrix $\mathbf{H}$ (of size $N_S \times N_D$) between a transmitter and a receiver, which has complex values, can be represented as *two 2D-images* (one 2D-image for real values and another one for imaginary values). An example of the normalized real/imaginary 2D-image for a sample channel time-frequency grid with $N_D = 14$ time slots and $N_S = 72$ subcarriers (based on Long-Term Evolution (LTE) standard) is shown in Fig.1.

B. Network Structure

The overview of the proposed pipeline for DL-based channel estimation, named ChannelNet, is illustrated in Fig.2. The goal is to estimate the whole time-frequency of the channel using the transmitted pilots. Similar to LTE standard, Lattice-type pilot arrangement has been used for pilot transmission.

The estimated value of the channel at the pilot locations $\hat{\mathbf{h}}_p^{\text{LS}}$ (which might be noisy) is considered as the LR and noisy version of the channel image. To obtain the complete channel image a two stage training approach is presented:

- In the first stage, an SR network is implemented which takes $\hat{\mathbf{h}}_p^{\text{LS}}$ as the vectorized low resolution input image

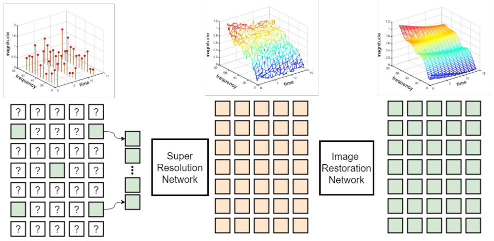

Fig. 2. The proposed pipeline for DL-based channel estimation

(once the real-part and then the imaginary-part) and estimates the unknown values of channel response $\mathbf{H}$.

- In the second stage to remove the noise effects, a denoising IR network is cascaded with the SR network.

For the *SR* and *IR*, we have used *SRCNN* [10] and *DnCNN* [11], respectively. Due to page limitation, we cannot show their structure pictorially. At a high level though, *SRCNN* first uses an interpolation scheme to find the approximate values of the high resolution image (channel) and afterwards, improves the resolution using a three-layer convolutional network. The first convolutional layer uses 64 filters of size 9×9 and the second layer uses 32 filters of size 1×1, both followed by ReLu activation. The final layer uses only one filter of size 5×5 to reconstruct the image. *DnCNN* (details in [11]) is a residual-learning based network which composed of 20 convolutional layers. The first layer uses 64 filters of size $3 \times 3 \times 1$ followed by a ReLU. Each of the succeeding 18 convolutional layers uses 64 filters of size $3 \times 3 \times 64$ followed by batch-normalization and ReLU. The last layer uses one $3 \times 3 \times 64$ filter to reconstruct the output.

C. Training

Lets denote the set of all network parameters by $\Theta = \{\Theta_S, \Theta_R\}$, where the Θ_S and Θ_R denote the set of parameter values for SR and IR networks, respectively. The input to the ChannelNet is the pilot values vector $\hat{\mathbf{h}}_p^{LS}$ and the output is the estimated channel matrix is denoted as $\hat{\mathbf{H}}$:

$$\hat{\mathbf{H}} = f(\Theta; \hat{\mathbf{h}}_p^{LS}) = f_R(f_S(\Theta_S; \hat{\mathbf{h}}_p^{LS}); \Theta_R),$$

where f_S and f_R are the SR and IR functions, respectively.

The total loss function of the network is the Mean square error (MSE) between the estimated and the actual channel responses calculated as follows:

$$C = \frac{1}{\|\mathcal{T}\|} \sum_{\mathbf{h}_p \in \mathcal{T}} \|f(\Theta; \hat{\mathbf{h}}_p^{LS}) - \mathbf{H}\|_2^2, \tag{6}$$

where $\mathcal{T}$ is the set of all training data and H is the perfect channel. In (6), $\|\mathcal{T}\|$ is the size of the training set.

To simplify the training process, we use a two stage training algorithm. Where in the first stage we minimize the the loss of the SR network, C_1 :

$$C_1 = \frac{1}{\|\mathcal{T}\|} \sum_{\mathbf{h}_p \in \mathcal{T}} \|\mathbf{Z} - \mathbf{H}\|_2^2, \tag{7}$$

where $\mathbf{Z} = f_S(\Theta_S; \hat{\mathbf{h}}_p^{LS})$ is the output of the SR network.

In the second stage, we freeze the weights of the SR network and find the parameters of the denoising network by defining $\hat{\mathbf{H}} = f_R(\mathbf{Z}; \Theta_D)$ and minimizing the loss function C_2 :

$$C_2 = \frac{1}{\|\mathcal{T}\|} \sum_{\mathbf{h}_p \in \mathcal{T}} \|\hat{\mathbf{H}} - \mathbf{H}\|_2^2, \tag{8}$$

Note that, similar to image-based techniques, the optimal weights of the network is dependent on the value of the SNR; thus, to have a complete solution we have to re-train the network for each SNR value. This approach is practically impossible to implement because the SNR value is continuous. Fortunately, however, as the results in section IV demonstrates, training networks for a few SNR values (in our case only two values) can still lead to a good performance.

IV. Simulation Results

In this section we train the network and evaluate the MSE over a range of SNRs and compare the results with the widely used baseline algorithms.

We consider a single antenna at the transmitter and at the receiver. For the channel modeling and pilot transmission, we have used widely used LTE simulator developed by university of Vienna, Vienna LTE-A simulator [13]. Keras and Tensorflow using a GPU backend are used for implementation of our proposed scheme. For SR and IR networks, the training rate is set to 0.001 with batch size of 128 and at most 500 iterations. The training, testing and validation sets consist of 32000, 4000 and 4000 channels, respectively[1]. As in LTE, in our simulations, each frame consists of 14 time slots with 72 subcarriers. For the wireless channel models of Vehicular-A (VehA) and SUI5 (a model with long delay spread) with carrier frequency of 2.1 GHz, bandwidth of 1.6 MHz and UE (user equipment) speed of 50 km/h, are considered.

To see the performance, we have compared the accuracy of channel estimation for the proposed method with that of three state-of-the-art algorithms i.e. ideal MMSE, estimated MMSE and ideal ALMMSE [3] when 48 pilots are used in each frame. The MSE between the estimated and the actual channel realization is considered as the performance metric.

The results for VehA is presented in Fig.3. Note that the ideal MMSE has the best performance and gives a lower bound of the achievable MSE as the channel correlation matrix should be known fully (without any error) which is not a valid assumption in practical applications. Estimated MMSE tries to estimate the correlation matrix based on received signals and ideal ALMMSE is an approximate counterpart of ideal MMSE (but still has the complete knowledge of the channel statistics).

In Fig.3, it is demonstrated that for low SNR values, the proposed ChannelNet trained at the SNR value of 12dB (denoted by deep low-SNR) has comparable performance with the ideal MMSE and has a better performance than the ideal ALMMSE and the estimated MMSE. Additionally, it can be observed that after around a mid SNR value, the performance

[1] Source code : https://github.com/Mehran-Soltani/ChannelNet

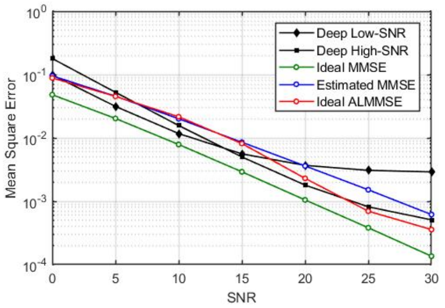

Fig. 3. Channel Estimation MSE in terms of SNR for VehA channel model.

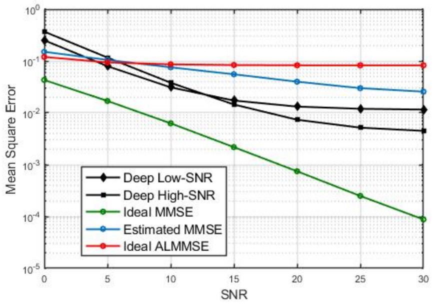

Fig. 4. Channel estimation MSE in terms of SNR for SUI5 channel model.

of the network trained at the SNR value of 22dB (denoted by deep high-SNR) is going to be better than the deep low-SNR.

So, we divide the SNR range into two regions. When the SNR value is low, we estimate the channel by deep low-SNR network, and beyond a threshold, deep high-SNR network is used. It can be observed that for SNR values higher than 23 dB, the performance of deep high-SNR is going to fail again and another network has to be trained; though as long as the SNR is below 20 dB, the two generated networks are sufficient.

MSE results associated with SUI5 model are depicted in Fig.4. In general, due to higher complexity of channel, all schemes show lower performance compared to the VehA model. More interestingly, after SNR value of 5dB, we can observe that schemes like ALMMSE and Estimated MMSE degraded significantly while the proposed deep model can still discover the underlying statistics and gets to an acceptable MSEs. As we expect, Ideal-MMSE has the best performance but it is not achievable in practical scenarios as it needs full knowledge of the correct channel statistics.

To show the performance of the proposed algorithm, considering VehA channel model, results of simulations for different number of pilots at the SNR level of 20dB are depicted in Fig.5. As can be seen, the ChannelNet, trained at that specific value of SNR, outperforms the Estimated MMSE and Ideal ALMMSE methods and it is comparable to Ideal MMSE.

V. Conclusion

In this paper, we presented ChannelNet, our initial DL-based algorithm for channel estimation in communication systems. In this method, we have considered the time-frequency response of a fading channel as a 2D-image and applied SR and IR algorithms to find the whole channel state based on the pilot values. The results show that the performance of ChannelNet is highly competitive with the MMSE algorithm. The two-step network training procedure has been presented and we also discussed how multiple ChannelNets should be used to best estimate the channel.

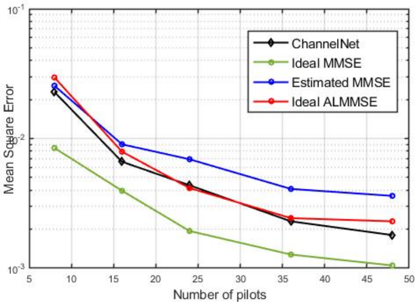

Fig. 5. Mean square error for channel estimation in terms of pilot number.

References

[1] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in ofdm systems," *IEEE Transactions on Broadcasting*, vol. 48, pp. 223–229, Sep 2002.

[2] Y. Li, L. J. Cimini, and N. R. Sollenberger, "Robust channel estimation for ofdm systems with rapid dispersive fading channels," *IEEE Transactions on Communications*, vol. 46, pp. 902–915, Jul 1998.

[3] M. imko, C. Mehlfhrer, M. Wrulich, and M. Rupp, "Doubly dispersive channel estimation with scalable complexity," in *2010 International ITG Workshop on Smart Antennas (WSA)*, pp. 251–256, Feb 2010.

[4] T. OShea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, pp. 563–575, Dec 2017.

[5] N. Samuel, T. Diskin, and A. Wiesel, "Deep mimo detection," in *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, July 2017.

[6] D. Erdogmus, D. Rende, J. C. Principe, and T. F. Wong, "Nonlinear channel equalization using multilayer perceptrons with information-theoretic criterion," in *Neural Networks for Signal Processing XI: Proceedings of the 2001 IEEE Signal Processing Society Workshop (IEEE Cat. No.01TH8584)*, pp. 443–451, 2001.

[7] C. Wen, W. Shih, and S. Jin, "Deep learning for massive mimo csi feedback," *IEEE Wireless Communications Letters*, pp. 1–1, 2018.

[8] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in ofdm systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, 2018.

[9] H. He, C. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmwave massive mimo systems," *IEEE Wireless Communications Letters*, vol. 7, pp. 852–855, Oct 2018.

[10] 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, pp. 295–307, Feb 2016.

[11] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising," *IEEE Transactions on Image Processing*, vol. 26, pp. 3142–3155, July 2017.

[12] S. Omar, A. Ancora, and D. T. M. Slock, "Performance analysis of general pilot-aided linear channel estimation in lte ofdma systems with application to simplified mmse schemes," in *2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–6, Sept 2008.

[13] C. Mehlführer, J. Colom Ikuno, M. Šimko, S. Schwarz, M. Wrulich, and M. Rupp, "The vienna lte simulators - enabling reproducibility in wireless communications research," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, p. 29, Jul 2011.