Received 11 June 2024; accepted 24 June 2024. Date of publication 1 July 2024; date of current version 25 July 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3421519

Enhancing mmWave Channel Estimation: A Practical Experimentation Approach With Modeled Physical Layer Impairments Incorporated in Deep Learning Training

RANDY VERDECIA-PEÑA [1,2] (Member, IEEE), RODOLFO OLIVEIRA [3,4] (Senior Member, IEEE), AND JOSÉ I. ALONSO [1,2] (Member, IEEE)

[1] Information Processing and Telecommunications Center, Universidad Politécnica de Madrid, 28040 Madrid, Spain

[2] ETSI de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain

[3] Departamento de Engenharia Electrotecnica, FCT, Universidade Nova de Lisboa, 2829-516 Costa da Caparica, Portugal

[4] Information Technology, Instituto de Telecomunicações, 1049-001 Lisbon, Portugal

CORRESPONDING AUTHOR: R. VERDECIA-PEÑA (e-mail: randy.verdecia@upm.es)

This work was supported in part by the Spanish Ministry of Science, Innovation and Universities funded by MCIN/AEI/10.13039/501100011033 under Project PID2020-113979RB-C21; in part by the Ministry of Economic Affairs and Digital Transformation and the European Union-NextGenerationEU (DISRADIO) under Project TSI-063000-2021-83; in part by FCT—Fundação para a Ciência e Tecnologia, I.P. under Project UIDB/50008/2020 and DOI identifier https://doi.org/10.54499/UIDB/50008/2020 and Project 2022.08786.PTDC and DOI identifier https://doi.org/10.54499/2022.08786.PTDC; and in part by the Madrid City Council through the 6GMADLab Project under Grant PC220935C247B.

ABSTRACT This paper introduces a novel methodology for wireless channel estimation in millimeter-wave (mmWave) bands, with a primary focus on addressing diverse physical (PHY)-layer impairments, including phase noise (PN), in-phase and quadrature-phase imbalance (IQI), carrier frequency offset (CFO), and power amplifier non-linearity (PAN). The key contribution centers around the innovative approach of training a convolutional neural network (CNN) using a synthetic and labeled dataset that encompasses a wide range of wireless channel conditions. The methodology involves the synthetic generation of labeled datasets, representing various types of wireless channels and PHY-layer impairments, which are subsequently employed in the CNN training stage. The resulting model-based trained CNN demonstrates exceptional adaptability to diverse operational scenarios, showcasing its capability to operate effectively under various channel conditions. To validate the efficacy of the proposed methodology, the trained CNN is deployed in a practical wireless testbed. Experimental results underscore the superiority of the proposed channel estimation methodology across different signal-to-noise ratio (SNR) regions and delay spread channel types. The trained CNN exhibits robust performance, confirming its effectiveness in mitigating the impact of PHY-layer impairments in real-world mmWave communication environments. This research not only advances reliable channel estimation techniques for mmWave systems but also provides valuable practical assessment results, with potential applications in next-generation wireless communication networks.

INDEX TERMS Wireless channel estimation, PHY-layer impairments, deep learning, performance evaluation, millimeter-wave communications.

I. INTRODUCTION

WIRELESS communication systems are evolving rapidly, driven by the increasing demand for higher data rates, lower latency, and improved spectral efficiency. One critical aspect in the optimization of wireless systems is channel estimation, which plays a pivotal role in decoding transmitted signals accurately. Traditional channel estimation methods often face challenges in dynamic and complex wireless environments. The integration of artificial intelligence (AI) and deep learning (DL) techniques has emerged as a promising solution to address these challenges, offering improved accuracy and adaptability.

Channel estimation is essential for mitigating the effects of fading, interference, and noise in wireless communication. Classical methods, such as least squares (LS) and minimum mean squared error (MMSE), have limitations in coping with the dynamic nature of wireless channels. AI techniques, including machine learning, have gained traction in wireless channel estimation due to their ability to adapt to changing channel conditions. The work in [1] proposed a support vector machine (SVM)-based approach for channel estimation in massive multiple-input multiple-output (MIMO) systems, showcasing enhanced performance in dynamic environments. Deep Learning, a subset of machine learning, leverages neural networks to automatically learn complex patterns from data. In wireless channel estimation, convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have shown promise [2]. DL techniques can also capture the channel correlation in massive MIMO systems to improve the channel estimation [3] and were also successfully adopted to address the estimation of double selective channels [4].

Millimeter-wave (mmWave) communication has emerged as a promising technology for next-generation wireless networks, offering increased data rates and capacity [5], [6], [7]. However, the challenges inherent in mmWave communication, including high attenuation, susceptibility to blockages, directional transmission characteristics, and susceptibility to atmospheric absorption, underscore the need for advanced channel estimation techniques [8]. Traditional methods often fall short of capturing the dynamic and complex nature of mmWave channels.

In recent years, parameters learning [9] and deep learning approaches [3], [10] have garnered significant attention for their ability to address the complexities of mmWave channel estimation. The work in [11] proposed a conventional neural network for mmWave channel estimation, where the network training was based on the received pilots from the base station without requiring knowledge of the true channel. Unlike conventional neural networks, CNNs automatically extract spatial features and offer advantages in capturing the intricate spatial characteristics of mmWave channels. Several works have adopted CNNs for mmWave channel estimation, including blind channel estimation based on denoising [12], and different mmWave scenarios such as large intelligent surfaces aided mmWave systems [13] and channel estimation and beamforming [14]. Considering the temporal dynamics of mmWave channels, RNNs have been employed to model the time-varying nature of channel conditions. RNN enhances the adaptability of deep learning models to dynamic mmWave environments, as considered in [15], where the authors adopted long-short term memory (LSTM) RNN models to address a scenario of nodes' mobility. Hybrid CNN and LSTM approaches were already considered for exploiting the spatial-temporal relationship of channel state information prediction [16]. More recently, DL attention learning methods were adopted in [17] to estimate the cascaded downlink channel in an intelligent reflective surface-assisted mmWave system. The authors showed that the adoption of the attention model can outperform CNN-based DL models. In [18], the authors proposed a mmWave channel estimation scheme based on generative networks, where deep generative priors adapted to the underlying channel model are identified as an advantage for one-bit estimation schemes. Model-driven channel estimation based on deep learning with feedback was considered in [19]. The adoption of transfer learning was explored in [20] to evaluate how previously trained DL modes can adapt and speed up channel estimation at different sites. A practical implementation of mmWave channel estimation through dedicated hardware was presented in [21] considering multiple DL models, to assess the latency caused by the channel estimation procedure.

The adoption of innovative deep learning methods to estimate the channel of mmWave systems has received increased attention. In [19], it is proposed a model-driven deep learning-based channel estimation and feedback scheme for wideband mmWave massive hybrid MIMO systems, demonstrating significant improvements by exploiting the sparsity of angle-delay domain channels with a jointly trained autoencoder model. In [18], it is formulated a channel estimation scheme for mmWave MIMO systems with one-bit ADCs as an inverse problem, utilizing a pre-trained deep generative model with a correlation-based loss function. The authors of [22] addressed the challenge of channel estimation in beamspace mmWave massive MIMO systems with limited RF chains by leveraging a learned denoising-based approximate message passing network, significantly outperforming state-of-the-art compressed sensing algorithms even with fewer RF chains. The work in [23] proposed a deep learning-based trainable proximal gradient descent network for mmWave channel estimation in massive MIMO systems, leveraging beamspace sparsity and integrating cross-layer feature attention fusion to outperform existing mmWave channel estimators. In [24], the authors introduced a federated learning-based channel state information estimation and feedback scheme for mmWave massive MIMO systems, where users train local models using their local datasets and exchange model parameters with the base station. However, recent works are still adopting consolidated deep learning models to address innovative wireless scenarios or more complex assumptions in terms of joint goals. In [25], the authors proposed a novel scheme using a feed-forward neural network to estimate the channel of OFDM systems with compressed-sensing-assisted index modulation over mmWave channels. The work in [10] presented a CNN-based channel estimation method for cell-free mmWave massive MIMO systems, demonstrating its effectiveness, flexibility, and efficiency. A common point in all these works is the fact that no physical impairments are considered, and their performance evaluation relies on simulated data without providing any kind of assessment in an experimental way.

Although it is known that RF impairments can cause a significant performance drop in the channel estimation methods adopted in OFDM systems [26], [27], the development of advanced channel estimation methods using deep learning techniques that leverage the information from impairments to enhance estimation efficiency has been a scarcely explored topic in the literature. While there has been a growing focus on the estimation and tracking of mmWave channels taking into account PHY-layer impairments [27], there has been comparatively little attention given to the influence of these impairments in DL-based schemes. As an example, the works in [2], [3], [4], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [22], [10], [23], [24], [25] do not consider PHY-layer impairments, which is the main topic and motivation of our work. Additionally, to the best of our knowledge, our work is the first to consider real-world physical impairments and present evaluation results obtained experimentally through an mmWave system testbed implementing the proposed methodology. Consequently, the results reported in our paper provide significant new performance benchmarks and innovative contributions to the community.

The contributions of this paper are as follows:

- The primary contribution of our paper lies in the development and implementation of a cutting-edge deep learning model designed for mmWave channel estimation. This model takes into account various PHY-layer impairments prevalent in mmWave communication systems, such as phase noise (PN), in-phase and quadrature-phase imbalance (IQI), carrier frequency offset (CFO), and power amplifier non-linearity (PAN). Our methodology introduces a novel approach for training a CNN explicitly tailored to handle these impairments. A key innovation in our methodology is the synthetic and theoretical model-driven generation of a labeled dataset that encompasses a wide range of wireless channel conditions. This dataset is then utilized for training the CNN, enhancing its adaptability to real-world scenarios.
- In addition to the novel model and training methodology, our paper contributes to the broader research community by adopting a reference theoretical model for generating training datasets. These datasets, along with the trained model, are made openly available to the community, fostering transparency and facilitating benchmarking for researchers in the field. This commitment to open access aims to promote collaboration, reproducibility, and standardization in mmWave channel estimation research.
- Our paper presents a comprehensive evaluation of the resulting model-based trained CNN. We showcase its remarkable adaptability across diverse operational scenarios, characterizing its performance through both simulations and practical testbed experiments.
- The performance evaluation includes a thorough analysis of the model's effectiveness in different signal-to-noise ratio (SNR) regions and various delay spread channel types. The findings affirm the superior performance of our proposed channel estimation methodology when compared to state-of-the-art techniques, highlighting its potential impact on advancing the reliability and efficiency of mmWave communication systems.
- Performance evaluation results are not only provided for simulated scenarios but also for realistic ones obtained experimentally through the implementation of a mmWave system testbed, where the proposed methodology was developed.

Regarding the structure of the paper, Section II delves into the system model. Section III introduces the channel estimation methodology, shedding light on the innovative approach taken to address PHY-layer impairments in millimeter-wave bands. In Section IV, the performance of the proposed performance is characterized through simulation. The practical setup adopted to evaluate the proposed method is introduced in Section V and several performance results are also presented. Finally, Section VI summarizes the key findings and contributions of the research.

Notations: In this paper, we adopt the following notations. Boldface lower and upper case symbols represent vectors and matrices, respectively. $[\cdot]_{k,i}$ is the element in k-th row and i-th column of a matrix. $\hat{\mathbf{A}}$ and $\hat{\mathbf{a}}$ denote the estimated matrix of $\mathbf{A}$ and the estimated vector of $\mathbf{a}$, respectively. Finally, $\check{\mathbf{B}}$ and $\check{\mathbf{b}}$ are the estimated channel matrix and vector in the data preparation stage, respectively.

II. MMWAVE HARDWARE IMPAIRMENT MODEL

The block diagram of a typical 5G OFDM end-to-end communication link without forward error correction codes (FEC) in the presence of hardware impairments is shown in Fig. 1. A single downlink 5G system consisting of a base station (BS) and user equipment (UE) is considered. In this section, a system model with several hardware impairments such as IQI, PN, CFO, and PAN is presented. As shown in Fig. 1, IQI, PN, and CFO exist at the transmitter and receiver sides, while PAN is only assumed at the transmitter side.

We assume a single-user (SU) downlink 5G transmission for a mmWave end-to-end communication link with a total of K subcarriers based on the 3GPP technical report in [28]. We consider the data stream $s(k)$ that is first converted to parallel streams and passed through K-points iFFT block resulting in $s(t)$. After that, a cyclic prefix (CP) of length T_{CP} ($L \leq T_{CP}$, where L is the channel duration) is then added to the signal stream and the resulting signal is up-converted using the digital to analog (D/A) block. Finally, the signal in the time domain is passed through hardware impairments (IQI, PN, CFO, and PAN), as shown in Fig. 1.

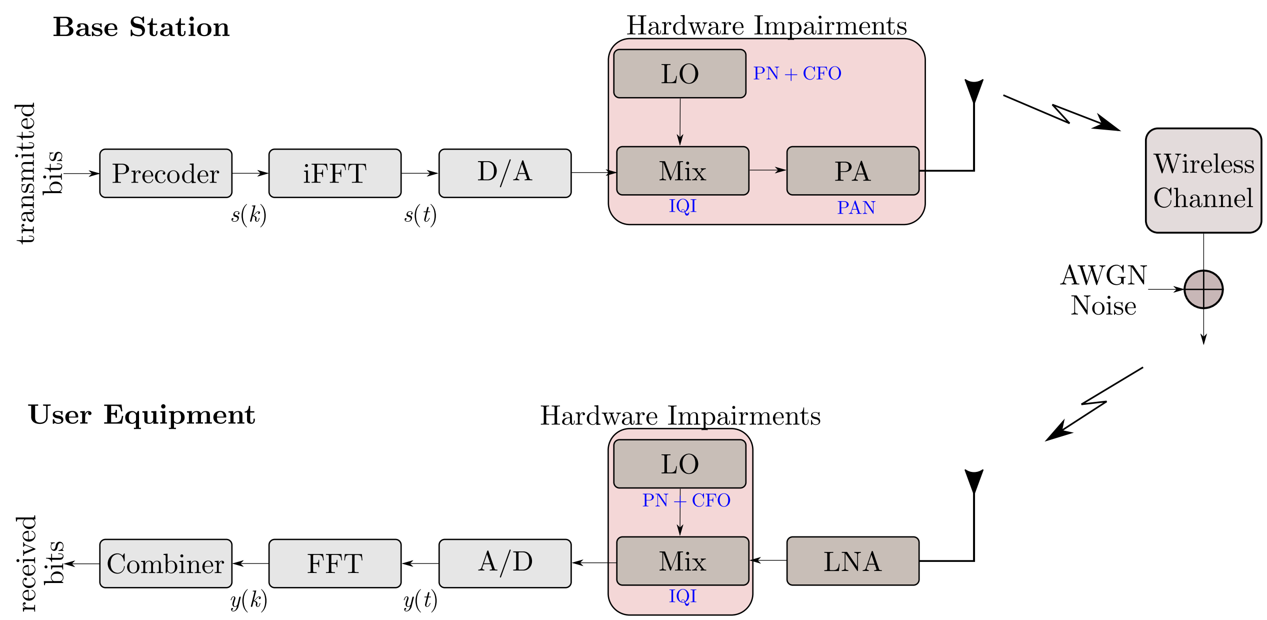

FIGURE 1. Block diagram of 5G end-to-end system model under hardware impairments.

The time-domain I/Q imbalance baseband equivalent model is given by [29]

$$x(t) \rightarrow \alpha_B s(t) + \beta_B s^*(t), \tag{1}$$

where $\alpha_B = \frac{1}{2}+\frac{1}{2}m_B e^{j\phi_B}$, $\beta_B = \frac{1}{2}-\frac{1}{2}m_B e^{j\phi_B}$, and m_B and ϕ_B denote the amplitude and phase imbalance parameters at the transmitter, respectively. The time function $s^*(t)$ represents the conjugate mirror of the data stream $s(t)$. Considering the imperfect local oscillator in the up-conversion at the transmitter, we can model the transmitted signal in (1) as [30]

$$x(t) \rightarrow \big(\alpha_B s(t) + \beta_B s^*(t)\big)e^{j\theta_B(t)}, \tag{2}$$

where $\theta_B(t)$ refers to the oscillator phase noise at the base station. The last impairment taken into account at the transmitter is the effect of the PA nonlinearity, assumed as a memoryless polynomial nonlinear model, representing the Rapp model standardized 3GPP [31]. Based on [30], the resultant signal after passing through the PA block can be written as

$$\check{x}(t) = \gamma\alpha_B s(t)e^{j\theta_B(t)} + \gamma\beta_B s^*(t)e^{j\theta_B(t)} + \xi(t), \tag{3}$$

where $\gamma = E[\check{x}(t)s^*(t)]/\sigma_x^2$ and $\xi(t)$ is the residual zero-mean complex Gaussian distributed distortion noise with power $\sigma_\xi^2 = E[|\check{x}(t) - \gamma s(t)|^2]$.

The transmitted signal in (3) undergoes a channel impulse response $h(t)$ and additive white Gaussian noise (AWGN) is added before it is received at the UE node, as shown in Fig. 1. At the receiver, the signal is amplified using the LNA block, and the resultant signal is down-converted under reception hardware impairments, including IQI, PN, and CFO. The equivalent time-domain received signal model at the m-th OFDM symbol after the FFT block can be written as

$$\begin{aligned} y_m(t) = &\,\alpha_U\Big(\gamma\alpha_B s_m(t)e^{j\theta_{B,m}(t)} * h(t)\Big)e^{j\theta_{U,m}(t)} \\ &+\alpha_U\Big(\gamma\beta_B s_m^*(t)e^{j\theta_{B,m}(t)} * h(t)\Big)e^{j\theta_{U,m}(t)} \\ &+\beta_U\Big(\gamma\alpha_B s_m(t)e^{j\theta_{B,m}(t)} * h(t)\Big)^* e^{-j\theta_{U,m}(t)} \\ &+\alpha_U(\xi_m(t) * h(t))e^{j\theta_{U,m}(t)} + \alpha_U n_m(t)e^{j\theta_{U,m}(t)}, \end{aligned} \tag{4}$$

where $\alpha_U = \frac{1}{2}+\frac{1}{2}m_U e^{j\phi_U}$ and $\beta_U = \frac{1}{2}-\frac{1}{2}m_U e^{j\phi_U}$, with m_U and ϕ_U representing the amplitude and phase imbalance parameters at the receiver, respectively. $\theta_U(t)$ refers to the oscillator phase noise at the receiver. Considering that the cyclic prefix is longer than the overall delay spread of the channel response, the time-domain model in (4) can be translated into frequency-domain as

$$\begin{aligned} \mathrm{Y}_{k,m} = &\,\hat{\mathrm{H}}_{k,m}\mathrm{S}_{k,m} + \check{\mathrm{H}}_{k,m}\mathrm{S}^*_{K\text{-}k+1,m} + \xi_{ICI,U,k,m} \\ &+\alpha_U \mathrm{H}_k \mathrm{J}_{0,U,m}\big(\xi_{ICI,B,k,m} + \Xi_{k,m}\big) + \mathrm{N}_{k,m}, \end{aligned} \tag{5}$$

where $\hat{\mathrm{H}}_{m,k} = \gamma\alpha_B\alpha_U \mathrm{J}_{0,B,m}\mathrm{J}_{0,U,m}\mathrm{H}_k$ and H_k is the channel frequency response at the k-th subcarrier. $\mathrm{J}_{0,B,m}$ and $\mathrm{J}_{0,U,m}$ are approximately $e^{j\theta_B^{CPE}}$ and $e^{j\theta_U^{CPE}}$, respectively, where θ_B^{CPE} and θ_U^{CPE} are the common phase errors caused by phase noise at the transmitter and receiver, respectively and correspond to the mean rotation during one OFDM symbol. Additionally, $\check{\mathrm{H}}_{m,k}$ is given by $\gamma\beta_B\alpha_U \mathrm{J}_{0,B,m}\mathrm{J}_{0,U,m} + \gamma\alpha_B\beta_U \mathrm{J}_{0,B,m}\mathrm{J}^*_{0,U,m}$. $\xi_{ICI,B}$ and $\xi_{ICI,U}$ represent the intercarrier interference impairments introduced by the carrier frequency offsets at the transmitter and receiver, respectively. Finally, $\Xi_{m,k}$ and n_k refer to residual nonlinear distortion noise and additive channel noise, respectively.

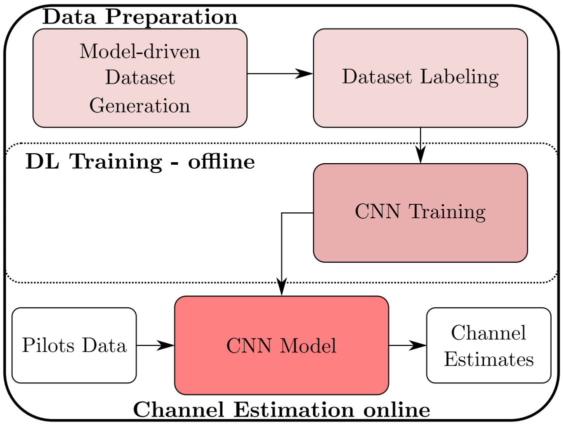

FIGURE 2. **Proposed methodology for learning and channel estimation.**

III. PROPOSED CNN-BASED ALGORITHM FOR CHANNEL AND HARDWARE IMPAIRMENTS ESTIMATION

In this section, we present the system model used in our work, which considers the effect of the major RF impairments on mmWave 5G communications that cause substantial performance degradation.

Several studies have adopted CNNs to estimate the wireless channel [32], [33]. CNNs are particularly well-suited for capturing the spatial correlation of channel state information (CSI) [33]. This is crucial when dealing with the spatial distribution of antennas or subcarriers in MIMO or OFDM systems, especially under non-time-varying channel conditions. RNNs are also commonly used in the literature but for scenarios involving time-varying channels, such as time-selective fading channels [34]. This is primarily due to RNNs' ability to model sequential dependencies, which is essential for handling the dynamic nature of such channels. In our work, we consider a quasi-static channel. Since the channel conditions are approximately constant over time, a CNN model is sufficient for capturing the subcarrier's channel pattern and the impact of the physical impairments on each one. Therefore, we have adopted a CNN DL model.

The system envisaged in this work is presented in Fig. 2.

In the first stage, data preparation is handled through the generation of a dataset that considers different PHY-layer impairments. The dataset is further labeled according to the impairments' information adopted to generate the dataset. During the offline stage, the learning procedure is implemented through the training of a CNN DL model for the different theoretical scenarios assumed in the labeled dataset. After training, the CNN is used for real-time estimation, considering the symbol pilots contained in the received frame over time, i.e., the demodulation reference signals (DM-RS) associated with the data channel are taken into account.

We first begin by introducing a proposed CNN-based algorithm description for the definition of input data and a new training label. We then present the defined CNN architecture design, the size of the convolutional filters, and the number of convolutional layers. Finally, we introduce a training algorithm that considers several channel models defined by 3GPP, as well as the 5G signal structure, which takes into account the 3GPP standard with synchronization, control, and data channels.

A. PROPOSED CNN-AIDED ESTIMATOR ALGORITHM

Considering the received signal in the frequency-domain as represented in (5) for the i-th time slot and k-th subcarrier, and taking into account an OFDM 5G subframe as standardized by the 3GPP with size $K \times N_S$, the time slot index i is in $[0, 1, \ldots, N_S - 1]$ and the range of subcarrier index k is in $[0, 1, \ldots, K - 1]$. The received signal can be written as

$$\mathrm{Y}_{k,i} = \hat{\mathrm{H}}_{k,i}\mathrm{S}_{k,i} + \mathrm{IQI}_{k,i} + \mathrm{ICI}_{k,i} + \mathrm{PAN}_{k,i} + \mathrm{N}_{k,i}, \tag{6}$$

where $\hat{\mathrm{H}}_{k,i}\mathrm{S}_{k,i}$ is the desired signal after experiencing the linear gain of PA, IQ imbalance, and CPE rotation at transmitter and receiver. $\mathrm{IQI}_{k,i}$ term is caused by the IQ imbalance effect and is given by the $\check{\mathrm{H}}_{k,m}\mathrm{S}^*_{K\text{-}k+1,m}$ term in (5). In addition, $\mathrm{ICI}_{k,i}$ describes the ICI impairment due to phase noise in transmitter and receiver, which in (5) is given as $\xi_{ICI,U,k,m} + \alpha_U \mathrm{H}_k \mathrm{J}_{0,U,m} \xi_{ICI,B,k,m}$. $\mathrm{PAN}_{k,i}$ is given by $\alpha_U \mathrm{H}_k \mathrm{J}_{0,U,m} \Xi_{k,m}$ in (5) and describes the nonlinear distortion caused by the PA behavior. The white Gaussian noise is denoted by $\mathrm{N}_{k,i}$.

Next, we introduce how the CNN model is used for channel estimation. The channel time–frequency response matrix is represented through two-dimensional (2-D), encapsulating both real and imaginary components [32]. In [35], the authors propose a novel CNN-based channel estimation algorithm in the presence of phase noise, where a linear interpolation algorithm is applied to the current slot to obtain the channel matrix input. In this work, we focus on 5G downlink end-to-end communication based on the physical structure proposed by 3GPP for the FR2 band. We propose a training label that considers different types of hardware impairments as shown in the system model in Fig. 1. The training label is represented by

$$\mathbf{W} = \mathbf{H} + \mathbf{IQI} + \mathbf{ICI} + \mathbf{PAN}, \tag{7}$$

where $\mathbf{H}$, $\mathbf{IQI}$, $\mathbf{ICI}$, and $\mathbf{PAN}$ represent the channel frequency response matrix, the in-phase and quadrature imbalance, the inter-carrier interference due to the phase noise, and the power amplifier nonlinearity hardware impairments, respectively.

Aiming to address the hardware impairments problem shown in Fig. 1, we propose an estimator algorithm based on a convolutional neural network. The architecture of the proposed methodology is illustrated in Fig. 3 and considers the training data labeled according to (7). The method is based on the interpolation pilot estimation of the received OFDM frame. In Fig. 3, the first step includes the extraction of the pilots from the known pilots and the received signal. The size of both resource grids is $K \times N_S$. Then, $\mathbf{p}$ and $\mathbf{p}_U$ vectors represent the obtained known pilot samples and the received pilot samples, respectively. Here,

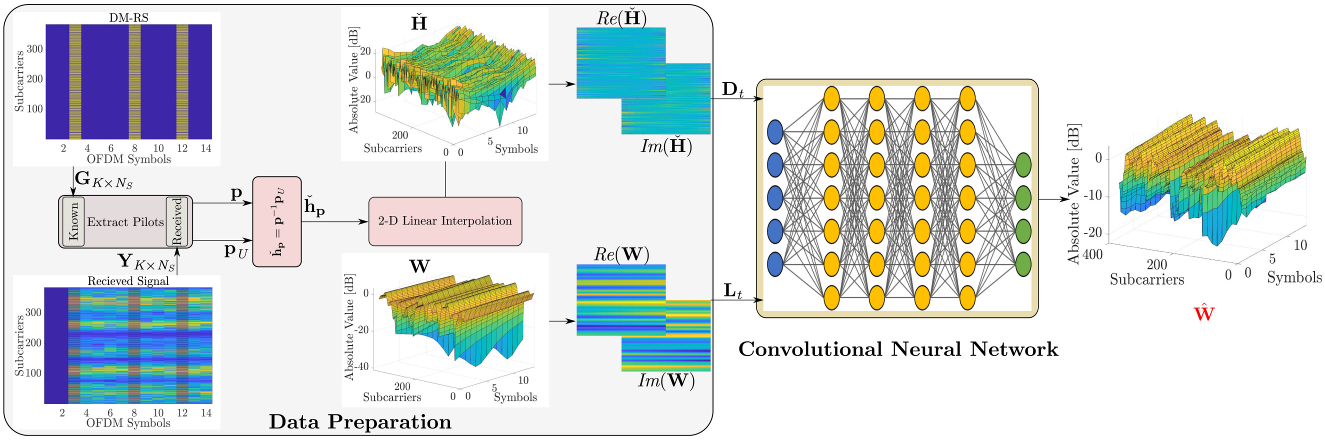

FIGURE 3. Block diagram of the proposed CNN-based algorithm for channel and hardware impairments estimation.

a least square channel estimation is applied to estimate the channel frequency response of the pilot positions. The result vector is given by $\check{\mathbf{h}}_\mathbf{p}$. After that, a 2-D linear interpolation is implemented to estimate the data positions in the received resource grid ($\mathbf{Y}$) and $\check{\mathbf{H}}$ is obtained. It should be highlighted that the estimated $\check{\mathbf{H}}$ matrix contains all hardware impairments introduced in both transmitter and receiver nodes. Knowing the input matrices of our convolutional neural network and considering the channel time-frequency response as an image proposed in [32], we can define the real and imaginary parts of $\check{\mathbf{H}}$ and $\mathbf{W}$ as the training data and the proposed training label, respectively. Consequently, the proposed neural network architecture for data-driven channel estimation is trained using $\check{\mathbf{H}}$ and $\mathbf{W}$ matrices to learn how to estimate the channel with hardware impairments during the offline training process. After training the network, the outputs of the CNN provides the estimate of the mmWave channel through the matrix represented by

$$\hat{\mathbf{W}} = \hat{\mathbf{H}} + \hat{\mathbf{IQI}} + \hat{\mathbf{ICI}} + \hat{\mathbf{PAN}}, \tag{8}$$

where the $\hat{\mathbf{W}}$ estimator matrix is employed to estimate, equalize, and decode the mmWave 5G received signal in real time.

B. TRAINING DATA

In this subsection, we describe how the proposed CNN-aided algorithm can collect the training data considering realistic assumptions like the 5G frame standardized by the 3GPP, channel conditions, and hardware impairments.

We consider a SISO downlink communication scenario with one gNodeB and one UE. The carrier frequency, bandwidth, and subcarrier spacing of the emulated 5G wireless communication are set as 27 GHz, 50 MHz, and 120 kHz, respectively. In addition, we have implemented a sampling rate of 61.44 Msps and employed a 512-point iFFT for OFDM modulation. According to the block diagram presented in Fig. 1, the resulting waveform after the iFFT block carries a single frame comprising 80 slots, each containing 14 symbols. Fig. 4 shows 10 slots of the 80 slots

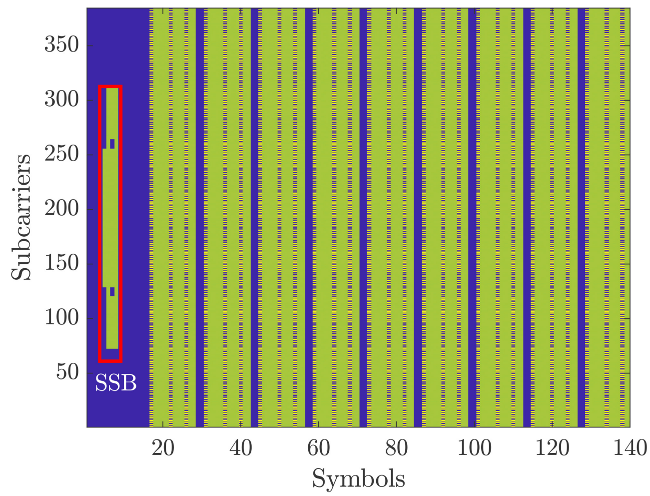

FIGURE 4. Transmitted RG: representation of 10 slots ($\mathbf{I}^{\mathrm{NR}}_{\mathrm{FR2}}$).

that carried the emulated FR2 frame, which considers all 5G channels included in the synchronization block (SSB). It should be noted that for the modulation scheme selection of the data channel, we have selected randomly among Q-PSK and 16-QAM constellations. The parameters describing FR2 signals are presented in Table 1. The D/A block subsequently upconverts 614400 samples before forwarding them to the hardware impairments block.

Regarding the hardware impairments, we assume PN, IQI, CFO, and PAN impairments, as illustrated in the system model in Fig. 1. We assume that both the transmitter and receiver nodes use identical fixed IQI coefficient imbalance of $m = m_B = m_U = 0.2$ dB and a phase imbalance of $\phi = \phi_B = \phi_U = 0.5$ degrees to the time-domain samples at the output of the mixer step (*Mix* block). To practically represent the PN of the oscillators at transmitter and receiver, a multi-pole/zero model is employed as proposed in [31], which is based on the proposed model by the 3GPP (a model A for high ICI cases has been used) and the expression for

TABLE 1. Main parameters to implement the FR2 signals.

Parameters	Signal 1	Signal 2
Bandwidth [MHz]	50	50
Carrier Frequency [GHz]	27	27
Samples Rate [MHz]	61.44	61.44
Cyclic Prefix	Normal	Normal
Sub. Spacing [kHz]	120	120
Modulation	Q-PSK	16-QAM
Target Code Rate	1/3	1/2
Number of FFT	512	512
Allocated RBs	32	32
K	384	384
N_s	80	80

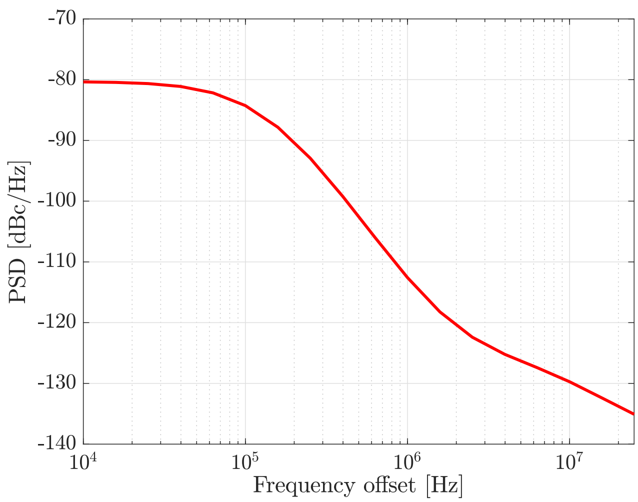

FIGURE 5. Phase noise spectrum for oscillator set at 27 GHz.

the PSD is given by

$$P(f) = P(0) \frac{\prod_{n_z=1}^{N_z} \left(1 + \left(\frac{f}{f_{n_z}}\right)^2\right)}{\prod_{n_p=1}^{N_p} \left(1 + \left(\frac{f}{f_{n_p}}\right)^2\right)}, \tag{9}$$

where $P(0)$ is the PSD at zero frequency, measured in dBc/Hz. The symbols f_{n_z} and f_{n_p} denote the frequency of the n-th zero and pole, respectively. N_z is the number of zeros and N_p is the number of poles. This mask is defined for a determined frequency offset, f_{offset}, from the central carrier f_c. Before and after this mask, the PSD is considered to be flat. To represent that the effects of PN are worse as the carrier frequency f_c increases, $P(f)$ is shifted by $20\log_{10}(f_c/f_{c,base})$ dBc/Hz. Fig. 5 shows the PN magnitude response implemented in this work at 27 GHz, which has been considered for LO impairment in both the transmitter and receiver sides.

In the following, we provide an overview of how the nonlinear impairment has been implemented to the FR2 5G signal generated after the upconversion step. We have used a memoryless power amplifier, which is a nonlinear device

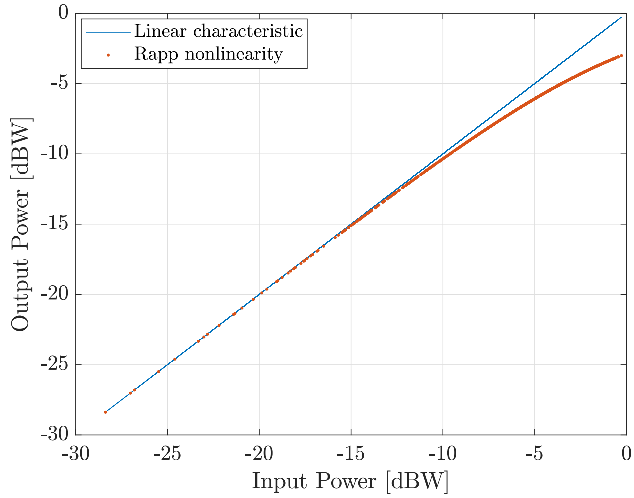

FIGURE 6. The relationship between input power and output power in dBW.

whose output does not depend on the previous input samples obtained at the Mix block output. We have adopted the Rapp model found in Annex 1 of 3GPP TR 38.803, which is given by the following expression

$$G = \frac{v}{\left[1 + \left(\frac{v}{v_0}\right)^{2p}\right]^{\frac{1}{2p}}}, \tag{10}$$

where v is the small signal gain, v_0 represents the saturation amplitude and p is the smoothness factor. For the implementation in this paper, we assume $v_0 = 1$ and $p = 1.05$. In addition, Fig. 6 illustrates the power ratio of the implemented memoryless power amplifier.

It is important to note that the generation of the dataset has been performed from a diverse set of propagation channel conditions proposed by the 3GPP standard and randomly selected.

In this sense, we have assumed that the mmWave communication link between the gNodeB and the UE is given by the line-of-sight (LOS) and non-line-of-sight (NLOS) propagation channel, as well as the urban macro (UMa) and indoor office (InH) scenarios for dataset generation based on the 3GPP standard has been implemented. In addition, we specifically explored the tapped delay line (TDL) models from A to E. Accordingly, the delay spread and Doppler shift have been defined from 10 to 300 ns and from 5 to 400 Hz, respectively. A summary of the system parameters described for dataset generation can be found at Table 2. It should be highlighted that for dataset generation, all these practical assumptions have been considered to extend the testing stages in a real-world environment providing a hard contribution to our work.

The training dataset acquisition is presented in Algorithm 1, which considers three input variables. The symbols in $\mathbf{I}_{\text{FR2}}^{\text{NR}}$ represent the features of the implemented FR2 signals.

TABLE 2. Default simulation parameters for dataset generation.

Parameters	Values
Scenarios [36]	Uma and InH
Channel conditions	LOS and NLOS
Channel delay profiles	TDL-A to TDL-E
SNR	0 to 20 dB
Delay spread	10 to 300 ns
Doppler shift	5 to 400 Hz
modulation order	Q-PSK and 16-QAM
IQ imbalance	$m = 0.2$ and $\phi = 0.5$
Nonlinear distortion	Rapp model
PLL model type	A at 27 GHz

Algorithm 1: Data Preparation Algorithm

Input : $\mathbf{Y}$, $\mathbf{W}$, $\mathbf{I}_{\text{FR2}}^{\text{NR}}$
$[\Gamma_{\text{c}}, \Gamma_{\text{d}}] = \Upsilon(\mathbf{I}_{\text{FR2}}^{\text{NR}})$
for $n_{fr} \leftarrow 0{:}N_{fr} - 1$ **do**
  $\mathbf{R}_{fr} = \mathbf{Y}(:, (N_{sf} \times N_d \times n_{fr} + 1) : (n_{fr} + 1)(N_{sf} \times N_d))$
  $\mathbf{T}_{fr} = \mathbf{W}(:, (N_{sf} \times N_d \times n_{fr} + 1) : (n_{fr} + 1)(N_{sf} \times N_d))$
  for $n_{sl} \leftarrow 0{:}N_s - 1$ **do**
    $\Gamma_{\text{c}}.N_s = n_{sl}$
    $\Gamma_{\text{c}}.N_{fr} = n_{fr}$
    $\mathbf{R}_{sl} = \mathbf{R}_{fr}(:, (N_d \times n_{sl} + 1) : (n_{sl} + 1)N_d)$
    $\mathbf{T}_{sl} = \mathbf{T}_{fr}(:, (N_d \times n_{sl} + 1) : (n_{sl} + 1)N_d)$
    if $n_{sl} \neq 0$ **then**
      $\mathbf{p} = \Lambda(\Gamma_{\text{c}}, \Gamma_{\text{d}})$
      $\mathbf{p}_U = \Lambda(\Gamma_{\text{c}}, \Gamma_{\text{d}}, \mathbf{R}_{sl})$
      $\check{\mathbf{h}}_{\text{p}} = \mathbf{p}^{-1}\mathbf{p}_U$
      2-D Linear Interpolation $\rightarrow \check{\mathbf{H}}$
      $\mathbf{D}_t = \Psi(\Re(\check{\mathbf{H}}), \Im(\check{\mathbf{H}}))$
      $\mathbf{L}_t = \Psi(\Re(\mathbf{T}_{sl}), \Im(\mathbf{T}_{sl}))$

Output: $\mathbf{D}_t$, $\mathbf{L}_t$

To generate the training data, we collected labels for several slot realizations of hardware impairments according to the system model described in Fig. 1. A total of 622160 slots were considered representing the transmission of 7777 frames with 80 slots. The symbol $\mathbf{Y} \in \mathbb{C}^{K \times N_s}$ represents the resource grids after the practical time and frequency corrections, SSB detection, and CP-OFDM demodulation for the end-to-end received signal in (6). The training labeled data is denoted by $\mathbf{W} \in \mathbb{C}^{K \times N_s}$. We assumed the system operates at a central frequency of 27 GHz within the SNR range from 0 to 20 dB, as detailed in Table 2.

The dataset collection was generated according to Algorithm 1, in which the first step corresponds to the extraction of the data (Γ_{d}) and control (Γ_{c}) features. In Algorithm 1, $\mathbf{R}_{fr} \in \mathbb{C}^{384 \times 1120}$ stores a frame associated with the n_{fr}^{th} frame received from the training data ($\mathbf{Y} \in \mathbb{C}^{384 \times 8710240}$, whose dimension 8710240 is given by the product of 80 slots per frame, 14 symbols per slot, and 7777 frames transmitted), and the $\mathbf{T}_{fr}$ matrix, $\mathbf{T}_{fr} \in \mathbb{C}^{384 \times 1120}$, represents a frame assigned from the training label ($\mathbf{W} \in \mathbb{C}^{384 \times 8710240}$). Let $\mathbf{R}_{sl} \in \mathbb{C}^{384 \times 14}$ and $\mathbf{T}_{sl} \in \mathbb{C}^{384 \times 14}$ matrices denote the slots of the training frame ($\mathbf{R}_{fr}$) and the label frame ($\mathbf{T}_{fr}$), respectively. The symbol n_{sl} represents the index of the slot in $\mathbf{R}_{sl}$ and $\mathbf{T}_{sl}$ based on the implemented FR2 signals in Table 1 and Fig. 4. For $n_{sl} = 0$ only a synchronization block is considered. Therefore, in this work, we exclude $n_{sl} = 0$ for training the data collection because the data channel is allocated from slots $n_{sl} = 1$ to $n_{sl} = 79$. We use the demodulation reference signals as standardized by the 3GPP for extracting pilot samples and to estimate $\check{\mathbf{h}}_{\text{p}}$; subsequently, a 2-D linear interpolation in the data position is performed to obtain the estimated complex channel matrix $\check{\mathbf{H}}$, which is adopted as the training data input.

It should be noted that CNN treats resource grids as 2-D images, requiring each grid element to be a real number. In this context, $\check{\mathbf{H}}$ and $\mathbf{T}_{sl} \in \mathbb{C}^{384 \times 14}$ resource grids contain complex data, where $\mathbf{T}_{sl}$ is a slot of the training labeled data. Therefore, the algorithm splits $\check{\mathbf{H}}$ and $\mathbf{T}_{sl}$ matrices into real and imaginary components, leading to two 3-D matrices. In the proposed algorithm, the training data and training labeled data are converted from a complex 384-by-14 matrix into a real-valued 384-by-14-by-2 matrix, with the third dimension representing the real and imaginary components. For predictions, because the real and imaginary grids must be input separately into the implemented CNN, the training data were converted into 4-D arrays of the form 384-by-14-by-1-by-2N_{sl}, where $N_{sl} = 614383$ is the number of training slots. The dataset can be denoted as

$$\Upsilon_{\text{d}} = \left[\mathbf{D}_t^0, \mathbf{L}_t^0\right], \left[\mathbf{D}_t^1, \mathbf{L}_t^1\right], \ldots, \left[\mathbf{D}_t^{\varphi_D - 1}, \mathbf{L}_t^{\varphi_D - 1}\right], \quad (11)$$

where $\mathbf{D}_t^{\varphi_d - 1}$ and $\mathbf{L}_t^{\varphi_d - 1}$ are the input features and the corresponding label for $\varphi_d = [0, 1, \ldots, \varphi_D - 1]$, and $\varphi_D = 1228766$ is the size of the training dataset. Therefore, a real-values training data matrix $\mathbf{D}_t \in \mathbb{R}^{384 \times 14 \times 1 \times 1228766}$ and a real-values training label matrix $\mathbf{L}_t \in \mathbb{R}^{384 \times 14 \times 1 \times 1228766}$ are obtained and considered to train the CNN architecture. For reproducibility purposes, we highlight that the training data and training labels obtained with the proposed algorithm are publicly available in [37].

C. CNN ARCHITECTURE

Encouraged by advances in deep learning across various fields and the success of convolutional neural networks in image processing, recent studies have demonstrated that CNNs are a promising architecture for channel estimation [2], [3], [22], [38], interpreting the time-frequency resource grid as a 2-D image. Therefore, to validate the effectiveness of our approach, we implemented the CNN architecture reported in [39], serving as a benchmark for evaluating the performance of our methodology. The

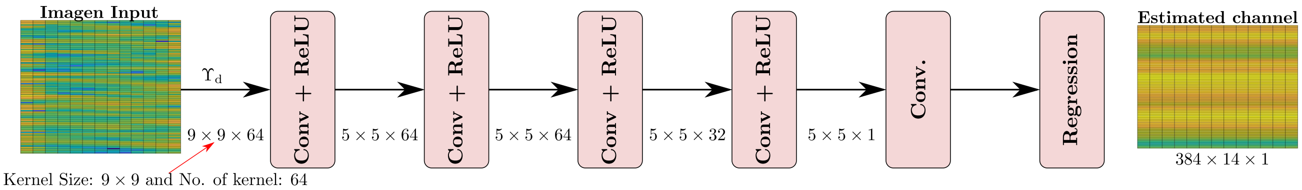

FIGURE 7. Proposed CNN architecture for channel and hardware impairments estimation.

architecture is illustrated in Fig. 7 and consists of a typical CNN model divided into four operation stages: convolution, nonlinearity with a rectifier linear unit (ReLU), pooling, and classification (fully connected layer) [32]. Therefore, the first layer accepts training data as the input, and the hidden layers of the proposed CNN architecture are implemented through five convolutional layers (L_n). The first convolutional layer adopts 64 filters of size 9×9, and the following two convolutional layers consist of 64 filters of size 5×5. Finally, the fourth and fifth hidden convolutional layers adopt 32 filters and one filter of size 5×5 for both layers, respectively. It should be noted that each hidden layer is associated with a ReLU activation layer, which is an element-based operation that replaces all negative element values in the characteristic map with zero. The output of the hidden layer is combined and passed through a fully connected layer that classifies the input image into various classes according to the training dataset.

The generated dataset included 1228766 real-valued samples and to ensure that the CNN did not overfit, the training data were divided into training and validation sets. Consequently, 983013 samples were used as validation data for monitoring the performance of the trained CNN, and 245753 realizations were employed for training. The training and validation steps were performed iteratively to address the potential issues of model overfitting. Considering the output of the proposed Algorithm 1, we define the MSE loss function as

$$\zeta_{\text{CNN}} = \frac{1}{\varphi_D} \sum_{\varphi_d=1}^{\varphi_D} \left\| f\left(\mathbf{D}_t^{\varphi_d}, \Phi\right) - \mathbf{L}_t^{\varphi_d} \right\|_2^2, \qquad (12)$$

which is minimized during the training stage using the adaptive moment estimation (ADAM) optimizer. Φ vector represents the weights and biases of the proposed CNN architecture. Hence, the input-output relation of the implemented CNN architecture can be expressed as [40]

$$\mathbf{L}_t^{\varphi_d} = f\left(\mathbf{D}_t^{\varphi_d}, \Phi\right) = f^{L_n-1}\left(\cdots f^{\,1}\left(\mathbf{D}_t^{\varphi_d}, \Phi_1\right), \Phi_{L-1}\right), \quad (13)$$

where $f^{L_{n_i}}$ denotes the activation functions of the layers and L_n is the total number of layers.

The proposed CNN-based algorithm for channel and hardware impairment estimation was implemented in MATLAB$^{\text{TM}}$ software using the Deep Learning Toolbox [41]. The model was trained for 5 epochs using a learning rate of 3×10^{-4} and a mini-batch size of 32. It should be noted that the learning rate during the training process determines the pace at which kernel weights are updated; thus, the total number of iterations during the training process is 38400, corresponding to 7680 for each of the five epochs used in the CNN architecture. In addition, the ADAM optimizer was adopted with the MSE loss function.

TABLE 3. Training parameters for CNN implementation.

Training parameters	Value
Number of epochs	5
Mini batch size	32
Iterations	38400
Iterations per epoch	7680
Optimizer	ADAM

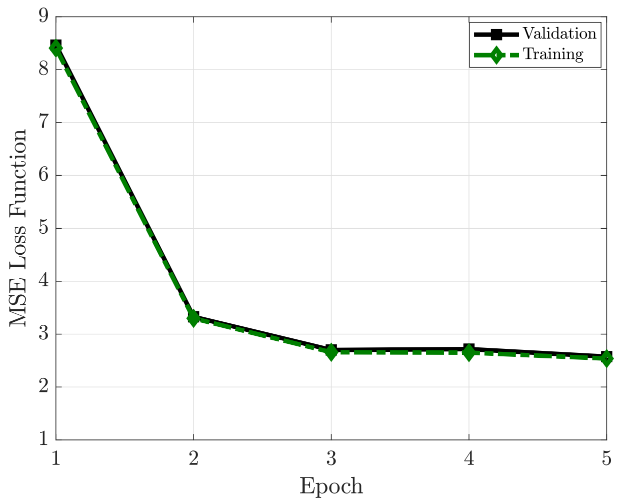

FIGURE 8. Loss value during CNN training and validation for different epochs.

A summary of the main parameters used for CNN implementation based on the hardware impairment model is presented in Table 3. Finally, to demonstrate learning convergence, Fig. 8 shows a curve describing the cost-function minimization of the CNN architecture during the training and validation stages. From the figure, we observe that the curves exhibit a similar trend. In addition, it can be concluded that during the training, the CNN loss function converged over the epochs, and the model loss after the training procedure got an averaging of 3.93 over the epochs. The MSE loss values obtained for training and validation purposes indicate a sustainable learning convergence over the epochs, as expected. For reproducibility purposes, the trained CNN is publicly available for download in [42].

IV. SIMULATION-BASED PERFORMANCE EVALUATION

This section describes the simulation scenario defined to test the CNN-based methodology for channel and hardware impairment estimation.

A. SIMULATION SETUP

We assess several key performance indicators (KPIs) of the considered end-to-end mmWave 5G communication system. We present Monte-Carlo computer simulations to evaluate the MSE, the bit error ratio (BER), and throughput versus SNR of the proposed CNN estimator. The results are analyzed and compared with a conventional link communication without hardware impairments correction. In addition, a comparison with a conventional link with hardware impairment correction is considered, assuming the conditions presented in [43], i.e., coarse frequency offset estimation and correction, I/Q imbalance estimation and correction, time and frequency synchronization using the demodulation reference signal (DM-RS) for control and data channels, and phase noise cancelation through the phase tracking reference signal (PT-RS).

We consider an indoor office environment for evaluating the effectiveness of the proposed channel estimation methodology. We assume that the communication link occurs within an NLOS scenario, with a distance of approximately 12.0 m between the transmitter and the receiver. For simulation purposes, we adopt the TDL-A propagation channel model. Delay spreads are usually shorter due to smaller room sizes and fewer significant reflectors. Typical values range from 5 to 50 ns in residual environments. In office environments, delay spreads can be longer due to larger spaces and more reflective surfaces (e.g., desks, partitions, etc.) and the typical values range from 10 to 100 ns. In large open indoor spaces, the delay spreads can be even longer due to larger distances and multiple reflections. Typical values range from 50 to 200 ns. In our work, we have considered the worst-case scenario, and $\sigma_\tau = 150$ and $\sigma_\tau = 200$ ns were considered according to the propagation scenarios described in the 3GPP report in [36]. The 3GPP indoor office path loss model is used to simulate the mmWave NLOS communication, which is expressed as

$$\mathrm{PL} = 38.3\log_{10}(\mathrm{d}_{3\mathrm{D}}) + 17.30 + 24.9\log_{10}(\mathrm{f}_\mathrm{c}) + \sigma_{\mathrm{SF}}, \quad (14)$$

where $\mathrm{d}_{3\mathrm{D}}$ represents the distance between gNodeB and UE, σ_{SF} denotes the shadowing term with zero mean and standard deviation of 8.03 dB, and f_c is the carrier frequency in GHz. In addition, we have considered all hardware impairments detailed in Table 2, and the main parameters of the transmitted FR2 signal are given by Signal 2 in Table 1. It should be highlighted that for testing purposes, the CNN estimator is evaluated by carrying out 5×10^3 independent Monte-Carlo trials adopting for each run an independent set of channel, noise, and hardware impairments realizations. For each transmission, 10 frames were sent. The UE node processes the received signal by applying the proposed CNN estimator before the data channel decoding stage.

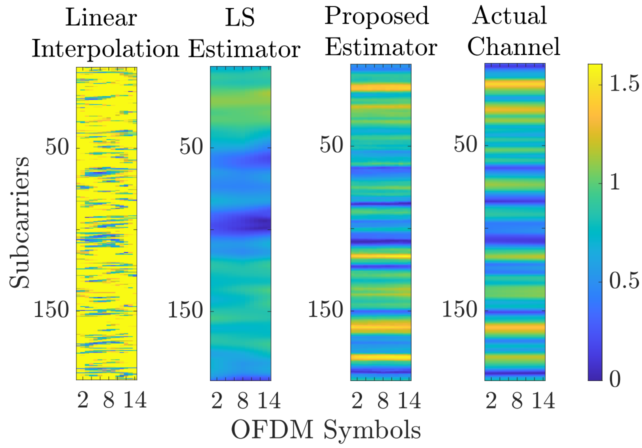

FIGURE 9. Spectrogram plot of the absolute value of the channel coefficient for channel models in an NLOS indoor-office channel model considering 16-QAM and σ_τ = 150 ns.

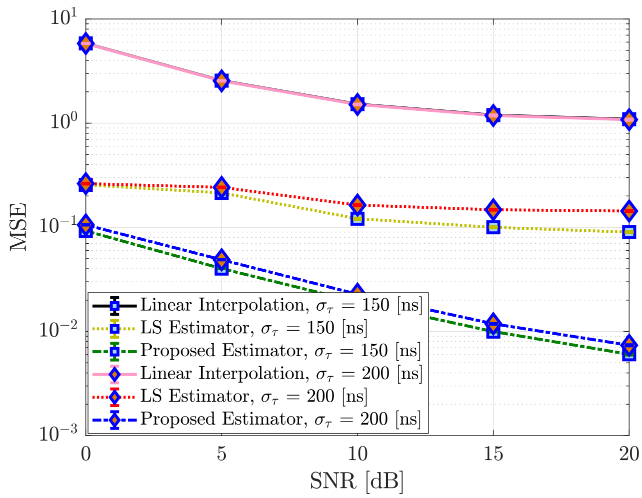

FIGURE 10. MSE comparison with confidence error bar using different channel estimation methods for 16-QAM modulation and σ_τ of 150 ns and 200 ns.

B. SIMULATION RESULTS

Fig. 9 shows the absolute value of the estimated channel coefficients as a spectrogram plot considering the described scenario for the 16-QAM modulation scheme and σ_τ =150 ns. It can be noticed that the figure illustrates the variation in channel coefficients across one data slot in the form of a 2-D image with a dimension of 384×14 (384 subcarriers and 14 symbols). Furthermore, the absolute values derived from both the traditional and proposed channel estimations are represented using a heat map.

A comparative analysis of performance is conducted among linear interpolation, LS practical estimation, and the proposed CNN estimator, considering the availability of actual channel coefficients to compute the MSE metric as shown in Fig. 10. The MSE performance of the proposed CNN estimator is compared with the channel estimation based on linear interpolation and LS practical estimation under different SNR values and for different delay spread values. From the results in Fig. 10, it can be observed that

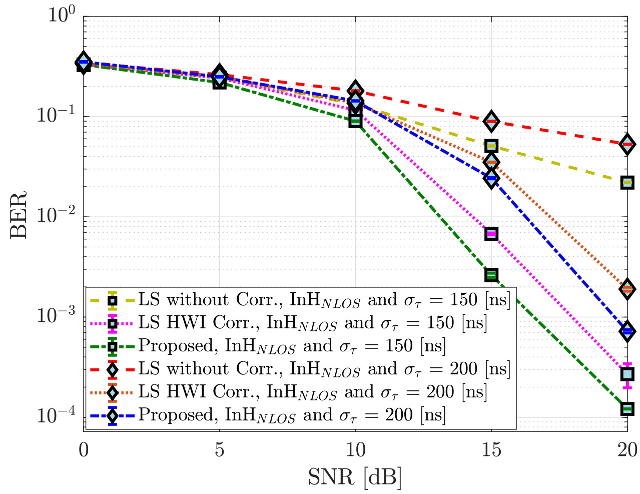

FIGURE 11. BER simulation results with confidence interval bar for the same scenario as considered in Fig. 10 (16-QAM modulation scheme and σ_τ of 150 ns and 200 ns).

the MSE of the proposed CNN estimator is significantly lower than the MSE values obtained with linear interpolation and LS estimator methods. According to Fig. 10, it can be verified that the LS estimator for $\sigma_\tau = 150$ ns reached the MSE level of 9×10^{-2} at approximately 20 dB. In addition, the MSE of the proposed estimator decreases when compared with the LS estimator, exhibiting an MSE value of 6.1×10^{-3} at approximately 20 dB. Finally, it is noteworthy that the linear interpolator achieves the lowest channel estimation performance.

Fig. 11 shows the corresponding BER performance comparison for the same scenario illustrated in Fig. 10. LS estimation is carried out considering the occurrence of hardware impairments (HWI), represented by the curves "LS without HWI Corr." or the cancelation of the impairments, represented by the curves "LS plus HWI Corr.". When considering the "LS without HWI Corr'.", the UE node performs functions opposite to those of the baseband steps of the transmitter. Based on this assumption, time synchronization is implemented based on continuous measurements of cross-correlation of the secondary synchronization signals (SSS, respectively). Subsequently, channel estimation based on the LS technique is applied to the slots carrying data symbols, where the LS estimator uses the DM-RS as pilot samples. Finally, the estimated channel matrix is used to implement the MMSE equalizer and for DL-SCH (downlink shared channel) decoding. It is important to note that under the "LS without HWI Corr." assumption, the hardware impairment corrections were not considered. Despite this, the performance remains relatively high, considering the significant impairments introduced in mmWave end-to-end communication systems. This performance can be attributed to the use of the low-density parity-check (LDPC) algorithm in the DL-SCH decoding block, as standardized by 3GPP.

When "LS plus HWI Corr." was employed, an algorithm, as described in [43] was actively running to correct the hardware impairments. The signal-processing chain begins with coarse frequency offset estimation and correction, followed by integer frequency offset estimation and correction to ensure that the signal is accurately aligned. Next, I/Q imbalance estimation and correction were performed using a blind adaptive algorithm to adjust the amplitude and phase discrepancies. Time synchronization and carrier frequency offset compensation are implemented based on continuous measurements of both autocorrelation and cross-correlation of the primary and secondary synchronization signals (PSS and SSS), respectively, allowing for the detection of the physical Cell ID, and full-bandwidth OFDM demodulation of the received waveform is performed. Subsequently, LS channel estimation was performed based on the DM-RS pilot sample-associated data channel, followed by MMSE equalization to mitigate any distortions. Finally, phased noise cancelation was applied to correct the phase noise discrepancies based on the PT-RS and following the 3GPP standard, ensuring that the signal was properly decoded [43]. The LDPC algorithm is also used in the DL-SCH decoding block.

From the obtained results in Fig. 11, it can be seen that the BER performance achieved when employing the conventional practical LS estimator is significantly high. We can observe the significant BER performance superiority of the proposed CNN estimator evidenced by the fact that for low SNR values (0 to 5 dB), the CNN estimator has comparable performance with the LS algorithm and better performance for SNR 5 dB. From Fig. 11, it can be verified that the proposed CNN estimator reached a BER level of 7.2×10^{-4} at approximately 20 dB and for $\sigma_\tau = 200$ ns. In this context, we highlight that the CNN estimator achieves the same BER level as the practical LS estimator at approximately 15.5 dB, and a performance gain of 4.5 dB is obtained. The proposed CNN algorithm yields better BER performance than the LS estimator plus hardware impairment correction (represented by the curve "LS plus HWI Corr."). Therefore, the mmWave end-to-end communication system using the proposed CNN estimator exhibits significantly higher performance across the SNR region. For example, considering $\sigma_\tau = 150$ ns and the proposed CNN estimator, the BER level of the end-to-end communication system improves and approaches the 1.2×10^{-4} BER level at around SNR = 20 dB, which suggests a gain of 1.12 dB of the LS with hardware impairment correction. As shown in Fig. 11, unlike the practical LS estimator used in data channel decoding, the CNN estimator outperforms the RF impairments of the hardware, reflecting its robustness to delay spread effects and hardware impairments.

The capacity performance of the end-to-end 5G communication scenario considered in Fig. 10 is illustrated in Fig. 12, which depicts the throughput achieved by the new radio downlink shared channel. The results show that the CNN estimator improves the throughput of the link over the LS estimator. This indicates that the proposed CNN estimator is effective in estimating the channel and

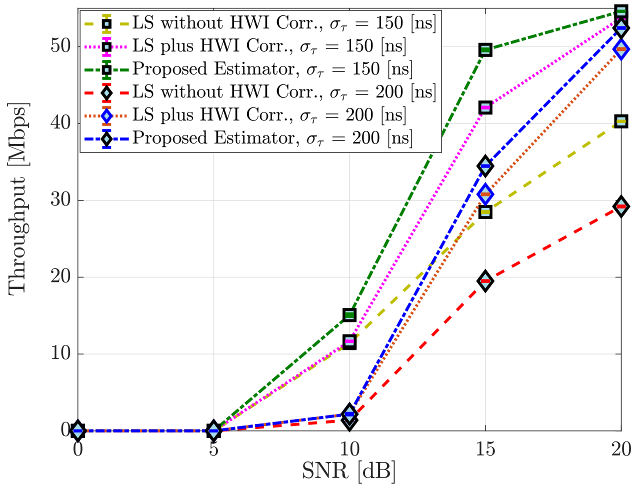

FIGURE 12. Throughput simulation results with confidence interval error bar for NLOS indoor-office channel model employing 16-QAM modulation scheme and σ_τ of 150 ns and 200 ns.

correcting hardware impairments. In addition, it can be observed that the conventional LS estimation suffers from severe degradation. The simulation results indicate that the conventional LS estimator achieves a throughput of approximately 29.21 Mbps for SNR = 20 dB; however, with the proposed CNN estimator, a throughput of 52.43 Mbps is achieved for the same SNR value, obtaining a performance gain of approximately 23.22 Mbps. Next, we assess the performance of the end-to-end communication system using the conventional LS estimation with hardware impairment correction. From Fig. 12, we observe that considering the practical LS estimator with HWI correction the end-to-end 5G communication achieves a high performance when compared to the LS estimator without HWI correction. In detail, the practical LS estimator with HWI correction achieves approximately 13.60 Mbps throughput improvement compared to the practical LS estimator without HWI correction at SNR = 15 dB and under σ_τ = 150 ns. In Fig. 12, the capacity performance obtained by the proposed CNN estimator under different σ_τ and SNR values is compared to the performance achieved by the practical LS estimator with HWI correction. The practical LS estimator with HWI correction suffers a degradation as noticed in Fig. 12. The proposed CNN channel and hardware impairment estimator can still outperform the HWI correction algorithm by up to 7.51 Mbps in terms of capacity for an SNR = 15 dB when σ_τ = 150 ns is considered. On the other hand, it should be noted that the maximum throughput performance of the simulated end-to-end mmWave 5G communication system is 54.60 Mbps, following the standard proposed by the 3GPP depending on the implemented FR2 frame structure, which is reached by the proposed CNN estimator for σ_τ = 150 ns. Finally, we highlight that the low-performance results observed for SNR < 5 dB are caused by a mixture of the combined effects (NLOS and high doppler spread), which cause high packet error rates. Consequently, in this region and for the considered NLOS scenario both LS solutions and the proposed one exhibit low performance because the packet decoding process is not able to succeed in such SNR conditions.

V. PRACTICAL PERFORMANCE EVALUATION

Data transmission in a real-world environment contains invaluable information regarding nonlinearities, phase noise, and the effects of hardware imperfections. To validate the robustness of the proposed CNN for mmWave channel estimation, in this section, we consider a hardware implementation testbench to capture the effects of these impairments on real-world channels. In this section, we briefly present the scenarios under evaluation and report the performance results achieved in the practical system.

A. EXPERIMENTAL TESTBED

Fig. 13 illustrates the hardware block diagram of the 5G communication system testbed. A list of the equipment adopted in the testbed is described in Table 4. On the gNodeB side, a personal computer (PC) generates a digital FR2 5G waveform compliant with 3GPP standards, implemented using MATLAB^{TM} software. The host computer is equipped with a USB modem 3.0, executing real-time physical layer and medium access control signal processing using the MATLAB^{TM} tool. Within the 5G signal chain, the FR2 5G baseband signal generated by the PC host is converted to analog waveforms in the FPGA board block. After processing in the FPGA board block, the signals emerged at an intermediate frequency (IF) of 3.5 GHz, necessitating subsequent upconversion to reach the 27 GHz carrier frequency. To facilitate this up-conversion process, our research group implemented a custom-developed RF front-end transmitter [44]. The 27 GHz transmitter/receiver modules were manufactured using off-the-shelf components, enabling straightforward characterization and integration into a high-level testbed. Table 4 presents the experimental components used in the manufactured transceiver. The horn antenna illuminating the UE was connected to the output of the mmWave front-end transmitter module. The FR2 5G signals passing through the mmWave wireless channel are then captured by the receiver antenna.

In the signal reception chain, signal processing at the implemented UE node reverses the steps undertaken in signal processing at the gNodeB. Custom-developed mmWave back-end receiver modules handle frequency conversion from mmWave to IF. Following this conversion, post-processing of the FR2 5G baseband occurred on the host computer for the signal received at the output of the FPGA receiver chain. In this stage, after time synchronization, SSB detection, and 5G waveform demodulation, the proposed CNN estimator is applied, and the estimated channel matrix is employed for the channel estimation, hardware impairment corrections, and decoding of the data channel.

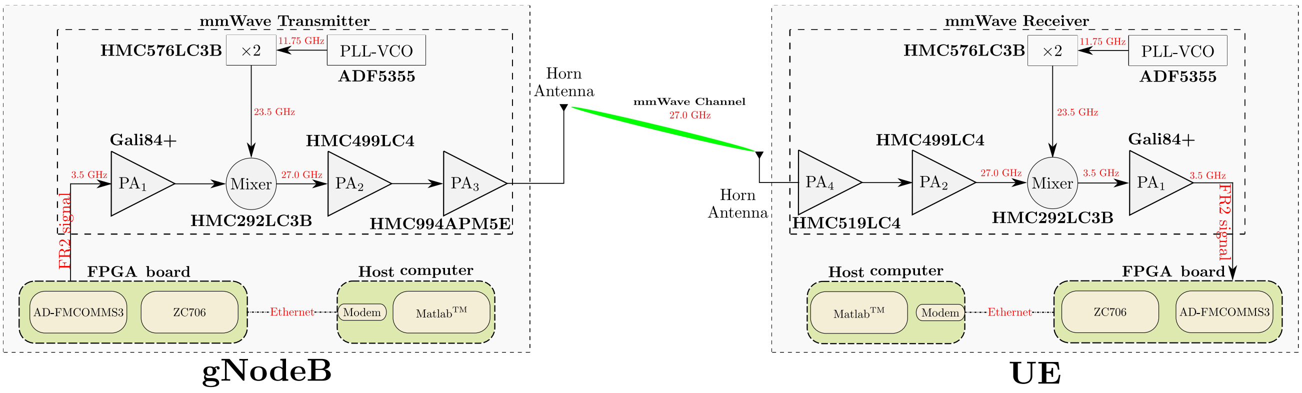

FIGURE 13. Hardware implementation testbench for mmWave CNN estimator validation.

TABLE 4. Summary of the experimental equipment name.

Trade Name	Component Type	Freq. (GHz)
ADF5355	PLL-VCO	6.8 to 13.6
HMC576LC3B	Multiplier $\times 2$	Up to 13.6
HMC292LC3B	Mixer	14 to 30
Gali84+	Power Amplifier (PA_1)	0 to 6
HMC499LC4	Power Amplifier (PA_2)	24 to 28
HMC994APM5E	Power Amplifier (PA_3)	20 to 28
HMC519LC4	Power Amplifier (PA_4)	18 to 28
Pasternack PE9851/2F	Horn Antenna	22 to 33
AD-FMCOMMS3-EBZ	FPGA board	Up to 6

B. MEASUREMENT SCENARIO

To evaluate the performance of the proposed CNN estimator in a real-world indoor scenario, an experiment was conducted in a hall scenario, as shown in Fig. 14. The shape of the hall is regular with a dimension of $79\times2.3\times3.0$ m^3, where the floor is marble and the walls are made of layered drywall.

In the hall, there are four big wooden cupboards symmetrically arranged on the left wall where each of which contains books, and the doors are made of glass. The base station is placed at a height of 1.55 m taking into account the horn transmitting antenna as a reference, and is denoted in the scenario with a blue rectangle. We have established an mmWave communication link between the base station and UE with an LOS propagation channel. In addition, two positions of the UE have been considered, respectively, 20 m and 25 m, as can be seen in Fig. 14. The red ellipses in the figure describe the locations of the mmWave UE node with a height of 1.15 m.

Measurements were taken with the developed testbench represented in Fig. 13. For assessing the KPIs in the deployed end-to-end mmWave 5G communication system using the proposed CNN estimator under the described scenario, we have varied the internal gain of the FPGA board by software. Therefore, the transmission power of the emulated gNodeB was between -20 dBm to 0 dBm with an interval of 5 dBm. Finally, we established 15 link communications with the UE node for every transmission power level, wherein each transmission carries out 100 frames, which allows testing the proposed CNN estimator over 7900 slots.

C. EXPERIMENTAL RESULTS

This subsection describes the evaluation of the proposed CNN estimator using realistic scenarios, as shown in Fig. 14. To evaluate the proposed estimator we define two metrics for the experimental study. We consider an FR2 5G structure as defined by the 3GPP for the mmWave bands, which has been implemented using MATLAB$^{\text{TM}}$ considering the parameters in Table 1, specifically for Signal 2 with the 16-QAM modulation scheme.

Fig. 15 shows the user's BER as a function of the gNodeB's transmission power P_{Tx}. As can be observed, for our deployed scenarios, we have performed a comparison between the proposed CNN algorithm with the LS estimator without correction and the LS estimator plus hardware impairment correction. In general, given the measurements that have been carried out in the two scenarios, the BER performance for all the considered approaches decreases with the transmission power. However, we can observe that the proposed algorithm ensures the lowest error rates of the end-to-end mmWave communication system for all of the considered transmission powers under the defined scenarios. As depicted in Fig. 15, the LS estimator without correction shows a much higher BER than the other approaches. More specifically, for $P_{Tx} = -5$ dBm and $P_{Tx} = 0$ dBm for the user's position at 25 m, the achieved BER performances were approximately 1.1×10^{-1} and 4.7×10^{-2}, respectively. These values are due to the error propagation caused by the hardware impairment of the developed hardware platform, and the channel matrix estimated by the algorithm is not capable of capturing the real effects of the mmWave propagation channel under the impairments.

In our experimental results, we introduced a practical LS channel estimation with hardware impairment correction as a reference benchmark following the technical details as standardized by the 3GPP, which considers pilot samples introduced in the 5G frame to correct the effect introduced

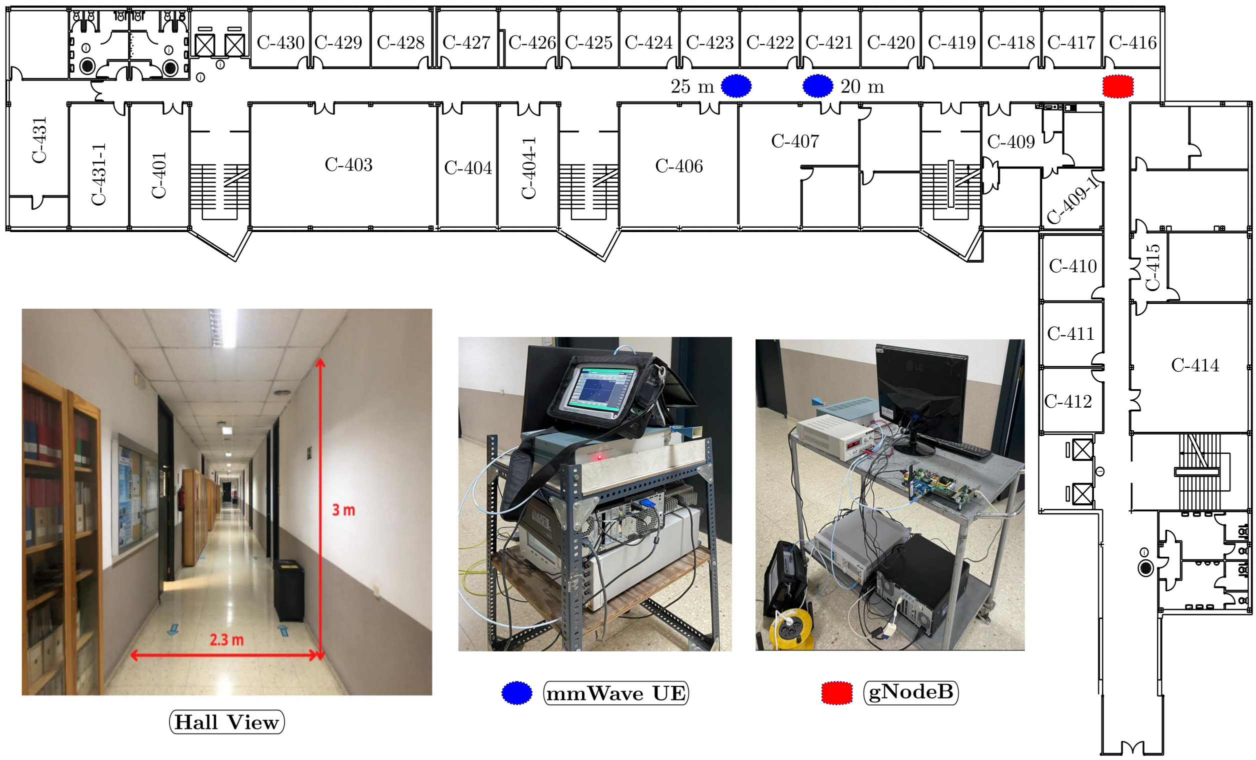

FIGURE 14. Measurement environment for testing the proposed CNN estimator in an indoor over-the-air environment.

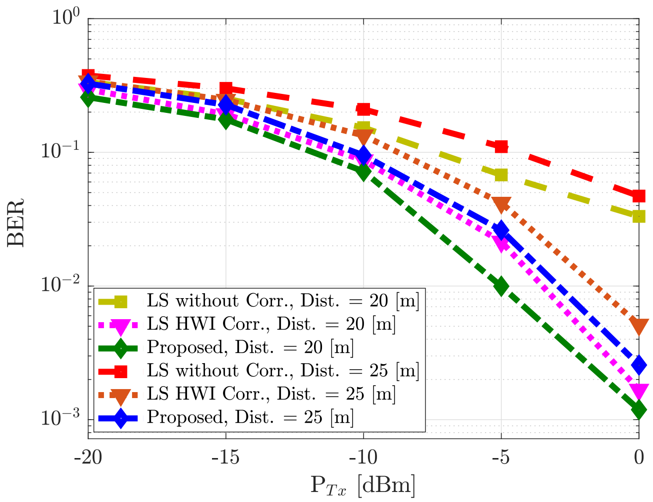

FIGURE 15. BER achieved and comparison for different scenarios.

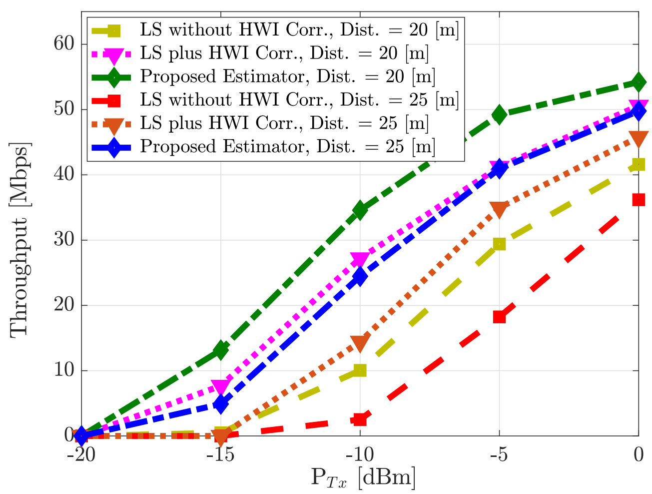

FIGURE 16. Throughput achieved and comparison for different scenarios.

by the channel and hardware, focusing on mmWave bands. Fig. 15 shows that the proposed CNN estimator outperforms the BER performance achieved by the LS estimator with HWI correction. Specifically, for $P_{Tx} = 0$ dBm and the scenario with 20 m, we obtained a BER level of 1.2×10^{-3}; however, the LS estimator with HWI correction achieved a BER performance of 1.7×10^{-3}.

Fig. 16 presents the throughput of the end-to-end mmWave communication system as a function of gNodeB transmission power. In this case, we tested the proposed CNN estimator with different distances between the gNodeB and UE (20 m and 25 m), and a comparison with reference algorithms is presented. In all cases, our algorithm outperforms the LS estimator without correction and the practical LS estimator with HWI correction for all considered transmission powers and scenarios. Specifically, for $P_{Tx} = -15$ dBm and 20 m, our estimator can provide an average throughput that is 27.39 and 1.71 times higher than the ones provided by the LS estimator without correction and the practical LS estimator plus HWI correction, respectively. Also, note that, unlike the other approaches, the proposed CNN estimator provides an average throughput maximum of 54.20 Mbps for $P_{Tx} = 0$ dBm, which is closer to the standardized by the 3GPP for the implemented architecture (54.60 Mbps). Finally, a mean capacity of 24.02 Mbps, 19.04 Mbps, and 11.39 Mbps was reached for the proposed CNN estimator, practical LS estimator with HWI correction, and convectional LS estimator without HWI correction, respectively, for the 25 m scenario.

D. FINAL REMARKS

In our approach, a supervised learning method was considered, as our primary goal was to embed the information of the physical impairments' models into a trained CNN for subsequent use in real-world experiments and thus evaluate its performance with an already validated DL technique. From the results reported in this section, the proposed approach ensures that the CNN model can accurately learn and apply the specific characteristics of physical impairments and the benefits of learning from existing theoretical physical impairment models. However, we acknowledge that unsupervised methods can also be employed, including self-supervised methods, a subset of unsupervised learning, where the models learn patterns and structures from unlabeled data underlying distribution or relationships within the data without predefined labels, that are usually implemented through variational autoencoders (VAE), generative adversarial networks (GAN), and others. These methods could be used to learn underlying data structures without labeled data, potentially reducing the reliance on extensive labeling processes, as recently reported in [18].

Finally, we highlight that the proposed supervised method relies on the synthetic labeled data used in the training process. Admittedly, the scalability of the proposed method can be limited by different factors including the accuracy of the theoretical physical impairments' models adopted to generate the labeled data; The selected region of the theoretical models' domain from where labeled data is sampled; The adequacy of the theoretical model parameters to represent the real-world scenarios where the CNN is adopted, even considering accurate models but parameterized unrealistically; The need for a high amount of labeled data for training purposes, particularly for the cases when long-tail distributions model the impairments. Consequently, the way how the synthetic labeled dataset is generated plays a crucial importance in the method's performance.

VI. CONCLUSION

In this paper, we addressed the channel estimation in an end-to-end mmWave 5G communication system. We have considered the 5G standard, and we have proposed a CNN estimator-based algorithm that adheres to the mmWave 5G standard structure. In addition, this paper introduced a groundbreaking methodology for wireless channel estimation in mmWave bands, specifically addressing multiple PHY-layer impairments. The pivotal contribution of this research lies in the innovative approach to training a CNN using a synthetic and labeled dataset that comprehensively captures various wireless channel conditions. The experimental results underscore the superiority of the proposed channel estimation methodology, demonstrating robust performance across different SNR regions and delay spread channel types. The success of the trained CNN in mitigating the impact of PHY-layer impairments in mmWave communication environments contributes to the advancement of reliable channel estimation techniques. This research holds promising implications for the evolution of next-generation mmWave wireless communication networks. We also test our algorithm based on the proposed CNN estimator through experiments with an FPGA-based millimeter-wave testbed in an indoor LOS environment and demonstrate that, in a real-world scenario, our proposed estimator improves end-to-end performance and outperforms the classical LS estimator with and without practical correction as defined by 3GPP.

REFERENCES

[1] L. V. Nguyen, A. L. Swindlehurst, and D. H. N. Nguyen, "SVM-based channel estimation and data detection for one-bit massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2086–2099, 2021.
[2] H. Ye, G. Y. Li, and B.-H. 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.
[3] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. NaderiAlizadeh, "Deep CNN-based channel estimation for mmWave massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 989–1000, Sep. 2019.
[4] A. K. Gizzini and M. Chafii, "RNN based channel estimation in doubly selective environments," *IEEE Trans. Mach. Learn. Commun. Netw.*, vol. 2, pp. 1–18, 2024.
[5] K. Zheng, H. Yang, Z. Ying, P. Wang, and L. Hanzo, "Vision-assisted millimeter-wave beam management for next-generation wireless systems: Concepts, solutions, and open challenges," *IEEE Veh. Technol. Mag.*, vol. 18, no. 3, pp. 58–68, Sep. 2023.
[6] X. Shang et al., "Some recent advances in measurements at millimeter-wave and terahertz frequencies: Advances in high frequency measurements," *IEEE Microw. Mag.*, vol. 25, no. 1, pp. 58–71, Jan. 2024.
[7] J. Kim, H. Chung, I. Kim, and G. Noh, "Integration of 5G mmWave-enabled V2I and V2V: Experimental evaluation," *IEEE Commun. Mag.*, vol. 62, no. 1, pp. 104–110, Jan. 2024.
[8] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.
[9] W. Shao, S. Zhang, X. Zhang, J. Ma, N. Zhao, and V. C. M. Leung, "Massive MIMO channel estimation over the mmWave systems through parameters learning," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 672–675, Apr. 2019.
[10] Y. Jin, J. Zhang, S. Jin, and B. Ai, "Channel estimation for cell-free mmWave massive MIMO through deep learning," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10325–10329, Oct. 2019.
[11] X. Zheng and V. K. N. Lau, "Online deep neural networks for mmWave massive MIMO channel estimation with arbitrary array geometry," *IEEE Trans. Signal Process.*, vol. 69, pp. 2010–2025, 2021.

[12] Y. Jin, J. Zhang, B. Ai, and X. Zhang, "Channel estimation for mmWave massive MIMO with convolutional blind denoising network," *IEEE Commun. Lett.*, vol. 24, no. 1, pp. 95–98, Jan. 2020.

[13] A. M. Elbir, A. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, "Deep channel learning for large intelligent surfaces aided mm-Wave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1447–1451, Sep. 2020.

[14] S. Lv, X. Li, J. Liu, and M. Shi, "Sub-6G aided millimeter wave hybrid beamforming: A two-stage deep learning framework with statistical channel information," *IEEE Trans. Green Commun. Netw.*, early access, Jan. 29, 2024, doi: 10.1109/TGCN.2024.3359208.

[15] S. Moon, H. Kim, and I. Hwang, "Deep learning-based channel estimation and tracking for millimeter-wave vehicular communications," *J. Commun. Netw.*, vol. 22, no. 3, pp. 177–184, Jun. 2020.

[16] C. Luo, J. Ji, Q. Wang, X. Chen, and P. Li, "Channel state information prediction for 5G wireless communications: A deep learning approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 227–236, Jan.–Mar. 2020.

[17] X. Fan, Y. Zou, and L. Zhai, "Spatial-attention-based channel estimation in IRS-assisted mmWave MU-MISO systems," *IEEE Internet Things J.*, vol. 11, no. 6, pp. 9801–9813, Mar. 2024.

[18] A. Doshi and J. G. Andrews, "One-bit mmWave MIMO channel estimation using deep generative networks," *IEEE Wireless Commun. Lett.*, vol. 12, no. 9, pp. 1593–1597, Sep. 2023.

[19] X. Ma, Z. Gao, F. Gao, and M. Di Renzo, "Model-driven deep learning based channel estimation and feedback for millimeter-wave massive hybrid MIMO systems," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2388–2406, Aug. 2021.

[20] W. Alves, I. Correa, N. González-Prelcic, and A. Klautau, "Deep transfer learning for site-specific channel estimation in low-resolution mmWave MIMO," *IEEE Wireless Commun. Lett.*, vol. 10, no. 7, pp. 1424–1428, Jul. 2021.

[21] P. K. Chundi, X. Wang, and M. Seok, "Channel estimation using deep learning on an FPGA for 5G millimeter-wave communication systems," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 2, pp. 908–918, Feb. 2022.

[22] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, Oct. 2018.

[23] P. Zheng, X. Lyu, and Y. Gong, "Trainable proximal gradient descent-based channel estimation for mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 12, no. 10, pp. 1781–1785, Oct. 2023.

[24] L. Zhao, H. Xu, Z. Wang, X. Chen, and A. Zhou, "Joint channel estimation and feedback for mm-Wave system using federated learning," *IEEE Commun. Lett.*, vol. 26, no. 8, pp. 1819–1823, Aug. 2022.

[25] H. Liu, Y. Zhang, X. Zhang, M. El-Hajjar, and L.-L. Yang, "Deep learning assisted adaptive index modulation for mmWave communications with channel estimation," *IEEE Trans. Veh. Technol.*, vol. 71, no. 9, pp. 9186–9201, Sep. 2022.

[26] A. Kiayani, L. Anttila, Y. Zou, and M. Valkama, "Channel estimation and equalization in multiuser uplink OFDMA and SC-FDMA systems under transmitter RF impairments," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 82–99, Jan. 2016.

[27] O. S. Faragallah, H. S. El-Sayed, and M. G. El-Mashed, "Estimation and tracking for millimeter wave MIMO systems under phase noise problem," *IEEE Access*, vol. 8, pp. 228009–228023, 2020.

[28] *5G; NR; Base Station (BS) Radio Transmission and Reception*, 3GPP Standard 5G TS 38.104, V.15.2.0, Release 15, 2018.

[29] F. Ghaseminajm, Z. Abu-Shaban, S. S. Ikki, H. Wymeersch, and C. R. Benson, "Localization error bounds for 5G mmWave systems under I/Q imbalance," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7971–7975, Jul. 2020.

[30] H. Chen et al., "Modeling and analysis of OFDM-based 5G/6G localization under hardware impairments," *IEEE Trans. Wireless Commun.*, early access, Dec. 12, 2023, doi: 10.1109/TWC.2023.3339523.

[31] *Study on New Radio Access Technology: Radio Frequency (RF) and Co-Existence Aspects*, 3GPP Standard TS 38.803, V.14.3.0, Release 14, 2022.

[32] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 652–655, Apr. 2019.

[33] S. Jo and J. So, "Adaptive lightweight CNN-based CSI feedback for massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 10, no. 12, pp. 2776–2780, Dec. 2021.

[34] Q. Bai, J. Wang, Y. Zhang, and J. Song, "Deep learning-based channel estimation algorithm over time selective fading channels," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 1, pp. 125–134, Mar. 2020.

[35] F. D. L. Coutinho, H. S. Silva, P. Georgieva, and A. S. R. Oliveira, "A novel CNN-based channel estimation algorithm in the presence of phase noise and CFO," *IEEE Wireless Commun. Lett.*, vol. 13, no. 1, pp. 193–197, Jan. 2024.

[36] "Study on channel model for frequencies from 0.5 to 100 GHz, Version 16.1.0, Release 16," 3GPP, Sophia Antipolis, France, Rep. TR 38.901, 2020.

[37] R. O. R. Verdecia-Peña and J. I. Alonso, 2024, "Dataset for CNN-aided estimator for channel and hardware impairment estimation," [Online]. Available: https://drive.upm.es/s/7oxKH9J9Gh9iASz

[38] A. S. M. M. Jameel, A. Malhotra, A. E. Gamal, and S. Hamidi-Rad, "Deep OFDM channel estimation: Capturing frequency recurrence," *IEEE Commun. Lett.*, vol. 28, no. 3, pp. 562–566, Mar. 2024.

[39] B. Banerjee, Z. Khan, J. J. Lehtomäki, and M. Juntti, "Deep learning based over-the-air channel estimation using a ZYNQ SDR platform," *IEEE Access*, vol. 10, pp. 60610–60621, 2022.

[40] A. Mohammadian, C. Tellambura, and G. Y. Li, "Deep learning-based phase noise compensation in multicarrier systems," *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2110–2114, Oct. 2021.

[41] "Deep learning toolbox." MathWorks. 2024. [Online]. Available: https://es.mathworks.com/help/deeplearning/

[42] R. O. R. Verdecia-Peña and J. I. Alonso, "CNN-aided estimator for channel and hardware impairment estimation." 2024. [Online]. Available: https://drive.google.com/file/d/1dqNL2pkVAGeyjwcx2kUnfrGFhbxk8Mg-/view?usp=sharing

[43] R. Verdecia-Peña, R. Oliveira, and J. I. Alonso, "A flexible mmWave layer 2 protocol implementation for integrated access and backhaul architecture," *IEEE Access*, vol. 11, pp. 52949–52968, 2023.

[44] R. Verdecia-Peña and J. I. Alonso, "Hardware platform for layer 2 IAB architecture based on a transceiver manufactured at 26 GHz," in *Proc. IEEE Int. Mediterr. Conf. Commun. Netw. (MeditCom)*, 2023, pp. 305–310.

RANDY VERDECIA-PEÑA (Member, IEEE) received the bachelor's degree in telecommunications and electronics engineering from the University of Oriente, Santiago de Cuba, Cuba, in 2014, the M.Sc. degree in electrical engineering from the Pontifical Catholic University of Rio de Janeiro, Brazil, in 2019, and the Ph.D. degree in telecommunication engineering from the Universidad Politécnica de Madrid (UPM), Spain, in 2023.

From 2014 to 2017, he worked as a Specialist in telematics with the Telecommunications Company of Cuba. He is currently a Postdoctoral Researcher with the Department of Information Processing and Telecommunications Center, ETSIT-UPM. His research interests include physical-layer solutions for 5G/6G mobile communications, 5G/6G hardware implementation, and the application of AI in wireless communications. He also participated in various research projects in collaboration with several top companies in the area of mobile communications. He was awarded a CAPES Scholarship from the Brazilian Ministry of Education in 2017 and an FPI Scholarship from the Spanish Ministry of Science, Innovation, and Universities in 2019.

RODOLFO OLIVEIRA (Senior Member, IEEE) received the Licenciatura degree in electrical engineering from the Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa (UNL), Lisbon, Portugal, in 2000, the M.Sc. degree in electrical and computer engineering from the Instituto Superior Técnico, Technical University of Lisbon, in 2003, and the Ph.D. degree in electrical engineering from UNL, in 2009. From 2007 to 2008, he was a Visiting Researcher with the University of Thessaly. From 2011 to 2012, he was a Visiting Scholar with Carnegie Mellon University. He is currently with the Department of Electrical and Computer Engineering, UNL, and is also affiliated as a Senior Researcher with the Instituto de Telecomunicações, where he researches in the areas of wireless communications, computer networks, and computer science. He serves as an Editorial Board for *Ad Hoc Networks*, *Elsevier*, *ITU Journal on Future and Evolving Technologies*, IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, and IEEE COMMUNICATIONS LETTERS.

JOSÉ I. ALONSO (Member, IEEE) received a degree in telecommunications engineering and the Ph.D. degree from the Technical University of Madrid (UPM).

He began his career at Telettra España, as a Microwave Design Engineer. He joined the Department of Signals, Systems, and Radiocommunications, UPM, where he is currently a Full Professor. His research has included the analysis and simulation of high-speed/high-frequency integrated circuits and their interconnections, as well as the computer-aided design and measurement of hybrid and GaAs monolithic microwave integrated circuits and their applications in the development and implementation of mobile, optical fiber, and communications systems. He has also worked in the development and radio planning of broadband point–multipoint radio systems in millimetre frequencies and wireless and mobile communications systems (WiFi, WiMAX, TETRA, GSM-R, and LTE). He is involved in the study of the viability of the use of 5G communications for critical communications and operational and passenger services in rail environments; in particular, its potential applicability to the FRMCS, in co-operative communications in 5G systems, and the development of localization techniques based on femtocell LTE networks. In addition, he holds three patents. He has authored more than 200 publications in scientific journals, symposium proceedings, seminars, and reports. He has participated in more than 90 research projects and contracts financed by national and international institutions and companies.