Quest Journals
Journal of Electronics and Communication Engineering Research
Volume 12 ~ Issue 2 (Mar.-Apr. 2026) pp: 05-33
ISSN(Online): 2321-5941
www.questjournals.org

Research Paper

CNN and Hybrid CNN-LSTM Artificial Intelligence-Based Algorithms for Adaptive Channel Estimation of Optical Plus Filtered OFDM For LTE Telecommunication Systems Design

[1]Fatai Ahmed-Ade(ORCID: https://orcid.org/0009-0003-4123-7338), [2]Vincent Andrew Akpan (ORCID: https://orcid.org/0000-0002-5155-2863), [3]Emannuel Omonigho Ogolo (ORCID: https://orcid.org/0000-0002-9444-298), [1]Durojaiye Jude Koffa (ORCID: https://orcid.org/0009-0002-1933-4396)

[1]*Department of Physics, Federal University Lokoja, Lokoja, Kogi State, Nigeria*
[2]*Department of Biomedical Engineering, The Federal University of Technology, Akure, Ondo State, Nigeria*
[3]*Department of Physics Electronics, The Federal University of Technology, Akure, Ondo State, Nigeria*
Corresponding Author: Vincent Andrew Akpan (vaakpan@futa.edu.ng)

ABSTRACT: *Reliable channel estimation is a fundamental requirement for achieving high-performance orthogonal frequency-division multiplexing (OFDM) systems, particularly under the frequency-selective and time-varying propagation conditions inherent in mobile Long-Term Evolution (LTE) networks. Conventional estimation techniques, namely least squares (LS) and minimum mean square error (MMSE), face well-documented limitations including vulnerability to noise amplification, high computational burden for MMSE, and poor adaptability in dynamically varying channel environments. This paper investigates the problem of pilot-assisted channel estimation for OFDM-based LTE systems and proposes two deep learning architectures as solutions: a standalone Residual Network-based convolutional neural network (ResNet-based CNN) and a hybrid ResNet-based CNN combined with bidirectional Long Short-Term Memory (LSTM) units. The ResNet-based CNN exploits spatial correlations inherent in the two-dimensional time-frequency pilot grid through hierarchical feature extraction with residual skip connections, batch normalization, and regularization, enabling stable training of deep architectures without gradient degradation. The hybrid ResNet-based CNN with LSTM architecture further incorporates temporal modeling through stacked bidirectional LSTM layers that track channel dynamics across successive OFDM symbols, proving particularly beneficial in high-mobility scenarios affected by Doppler spread. Extensive simulations were conducted across SNR levels of 0–30 dB, ITU Pedestrian A and Vehicular A channel models, and mobility conditions spanning 3 to 120 km/h. At the canonical operating point of 20 dB SNR, the hybrid ResNet-based CNN with LSTM achieves a mean squared error (MSE) of −28.4 dB, representing improvements of 3.2 dB over LS, 2.3 dB over MMSE, and 1.8 dB over the standalone ResNet-based CNN. Bit error rate is reduced to 3.1 × 10^{-4} and spectral efficiency improves by 28% over LS-based estimation. Validation on real MTN Nigeria LTE downlink captures confirm the generalization capability of the proposed architectures beyond idealized simulation conditions. The experimental results further affirm the potential synergy of deep learning channel estimators with Optical plus Filtered OFDM (O+F OFDM) waveforms, which offer enhanced spectral efficiency and support for hybrid optical-wireless LTE infrastructure.*

KEYWORDS: *Channel Estimation, Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), Long Term Evolution (LTE), Optical plus Filtered OFDM (O+F OFDM), ResNet Model Architecture*

*Received 24 Apr., 2026; **R**evised 28 Apr., 2026; **A**ccepted 11 May., 2026 © The author(s) 2026.*
P**ublished with open access at **www.questjournas.org

I. INTRODUCTION

The rapid evolution of wireless communication systems has been driven by an ever-increasing demand for higher data rates, improved spectral efficiency, and robust connectivity has made OFDM a foundational technology in LTE, Wi-Fi, and 5G New Radio [1]. By dividing the channel into many orthogonal subcarriers, Orthogonal frequency-division multiplexing (OFDM) effectively handles frequency-selective fading, but accurate channel state information (CSI) is still required for coherent detection and adaptive transmission.

Orthogonal frequency-division multiplexing has established itself as the cornerstone modulation technique for modern wireless standards, including Long-Term Evolution (LTE) [2], the fundamental advantage of OFDM lies in its ability to transform frequency-selective fading channels into a collection of parallel flat-fading subchannels, thereby simplifying equalization and enabling efficient utilization of available bandwidth [2].

Despite these advantages, the performance of OFDM systems critically depends on the availability of accurate channel state information (CSI) at the receiver. Channel estimation, the process of inferring the complex-valued frequency response of the wireless propagation environment, enables coherent demodulation and adaptive resource allocation [3]. In the absence of reliable CSI, system performance degrades substantially, manifesting as increased symbol error rates, reduced throughput, and inefficient power utilization.

Channel estimation is a technology that is used to greatly increase the system performance, notably for 4G, 5G and Long Term Evolution (LTE) systems [4–6]. It is an important aspect of wireless communications. Orthogonal frequency division multiplexing (OFDM) symbols with the time-varying channel frequency response are used to estimate the channel. All OFDM-based wireless transmission receivers rely on channel estimation. Either least squares or minimum mean square error (or MMSE) dependent on pilot aided frequency-domain channel estimation algorithms are used. However, the application deep learning with transfer or pretrained models has shown remarkable results for channel estimation of OFDM systems [7–13].

Conventional pilot-assisted estimators like LS are simple but noise-sensitive, while MMSE performs better yet depends heavily on accurate second-order statistics [2, 14–18]. Deep learning offers an attractive alternative by learning complex mappings directly from data, bypassing explicit statistical modeling [19, 20].

Recent advances also explore Optical plus Filtered OFDM (O+F OFDM) variants, which integrate optical processing stages with filtered OFDM to achieve better out-of-band emission control and support hybrid optical-RF systems especially relevant for LTE backhaul and fronthaul in operator networks such as MTN [1], [2]. These developments motivate investigation of intelligent channel estimation techniques that can complement such advanced OFDM implementations.

Conventional channel estimation approaches can be broadly classified into blind and pilot-assisted techniques. Blind methods exploit statistical properties or structural characteristics of the transmitted signal to estimate channel parameters without dedicated training sequences [18]. While spectrally efficient, these approaches typically suffer from high computational complexity, slow convergence rates, and sensitivity to model mismatch, rendering them impractical for real-time applications in rapidly varying channels [19].

Pilot-assisted estimation methods address these limitations by inserting known reference symbols at predetermined locations within the time-frequency resource grid. The least squares estimator, owing to its computational simplicity, has become the de facto baseline for pilot-based channel estimation [20]. However, LS estimation directly inverts the pilot symbol matrix without considering noise statistics, leading to significant performance degradation in low SNR environments [21]. The MMSE estimator improves upon LS by incorporating channel autocorrelation information and noise statistics, achieving near-optimal performance under Gaussian assumptions [22]. Nevertheless, MMSE requires accurate knowledge of second-order channel statistics, which may not be available in practice and must be estimated, potentially introducing additional errors.

The emergence of deep learning has revolutionized numerous fields, from computer vision to natural language processing, by enabling automated feature extraction from high-dimensional data [23]. In the context of wireless communications, deep neural networks offer a fundamentally different paradigm: rather than deriving estimation algorithms from first principles and channel models, learning-based approaches automatically discover optimal strategies through data-driven training [24]. This capability is particularly valuable for channel estimation, where the wireless propagation environment exhibits complex, non-linear characteristics that defy simple parametric models.

Convolutional neural networks have demonstrated exceptional performance in extracting spatially-correlated features from structured data [25]. In the context of OFDM, the arrangement of pilot symbols across subcarriers and OFDM symbols naturally forms a two-dimensional grid amenable to convolutional processing. Recent studies have shown that CNNs can learn hierarchical representations of channel characteristics, capturing both local correlation patterns and global structure [26–28]. However, standard CNN architectures lack explicit mechanisms for modeling temporal dependencies across successive OFDM symbols, limiting their effectiveness in time-varying channels induced by user mobility.

Long short-term memory (LSTM) networks, a specialized form of recurrent neural networks, excel at modeling sequential data with long-range dependencies [29]. Through sophisticated gating mechanisms, LSTM units can selectively retain or discard information across time steps, making them ideally suited for tracking evolving channel conditions [30]. The synergistic combination of ResNet-based CNNs and ResNet-Based CNN with LSTMs in hybrid architecture offers the potential to exploit both spatial correlation in pilot arrangements and temporal correlation across OFDM symbols, thereby achieving superior estimation performance in challenging propagation scenarios.

The architectural selection for deep learning-based channel estimation must account for the unique structure of OFDM systems. In OFDM, pilot symbols are arranged in a two-dimensional time-frequency grid, creating spatial correlations that can be exploited for interpolation. The proposed ResNet-based CNN architecture specifically targets this spatial structure through hierarchical feature extraction. Unlike traditional CNNs that suffer from vanishing gradients in deep configurations, ResNet employs residual connections (skip connections) that allow gradients to flow directly through the network. This architectural innovation, originally developed by He et al. for image recognition, proves remarkably effective for OFDM channel estimation because [27]:

1). Depth without degradation: The residual blocks enable training of networks deep enough to capture multi-scale channel features spanning different coherence bandwidths;

2). Identity mapping preservation: Skip connections ensure that the network can learn to preserve known pilot information while refining estimates at data subcarriers; and

3). Hierarchical feature extraction: Sequential convolutional layers with increasing receptive fields capture both fine-grained local correlations (adjacent subcarriers) and coarse global patterns (channel frequency selectivity).

The specific ResNet architecture employed comprises multiple residual blocks, each containing two convolutional layers with batch normalization and ReLU activation, connected via identity shortcuts. This configuration processes the 2D pilot grid and interpolated initial estimates to produce refined channel frequency response estimates across all subcarriers.

For time-varying channels encountered in mobile LTE scenarios, spatial processing alone proves insufficient. Channel coefficients evolve across consecutive OFDM symbols due to Doppler spread, creating temporal dependencies that CNNs cannot inherently model. This motivates the Hybrid ResNet-based CNN with LSTM architecture, which integrates: *1).* ResNet-based CNN frontend: Extracts spatial features from each OFDM symbol's pilot pattern independently; and *2).* Bidirectional LSTM backend: Models temporal evolution by processing sequences of CNN-extracted features across multiple symbols, capturing both forward and backward temporal dependencies.

The LSTM component consists of memory cells with gating mechanisms (input, forget, output gates) that selectively propagate relevant channel state information across time. By processing symbol sequences bidirectional, the architecture can leverage both past and future context when estimating the current symbol's channel analogous to how forward-backward algorithms improve sequence prediction. This temporal modeling proves particularly valuable under high mobility (60-120 km/h) where Doppler shifts cause rapid channel variation.

The principal contribution of this paper includes: *1).* the development of a comprehensive ResNet-based CNN channel estimation architecture tailored for OFDM systems, incorporating batch normalization for training stability and dropout regularization to prevent overfitting. The network processes the two-dimensional pilot arrangement to extract spatial features and generate accurate channel frequency response estimates; *2).* Proposal of a novel hybrid CNN-LSTM framework that integrates time-distributed convolutional processing with bidirectional LSTM layers. This architecture explicitly models both spatial and temporal channel characteristics, enabling robust estimation in high-mobility scenarios with significant Doppler spread; *3).* extensive performance evaluation across diverse operating conditions, including varying SNR levels, multiple standard channel models (ITU Pedestrian A, Vehicular A), and different mobility speeds (3-120 km/h). The results demonstrate consistent superiority of deep learning approaches over traditional methods; and *4).* detailed

Table 1. List of abbreviations and meaning

S/N	Abbreviation	Meaning	S/N	Abbreviation	Meaning
1	O+F-OFDM	Optical plus Filtered Orthogonal Frequency Division Multiplexing	13	DVB-T2	Digital Video Broadcasting Second Terrestrial
2	LTE	Long term Evolution	14	GRU	Gated Recurrent Units
3	CSI	Channel State Information	15	MAC	Mutiply Accumulate
4	OFDM	Orthogonal Frequency Division Multiplexing	16	ITU	International Telecommunication Unit
5	CNN	Convolutional Neural Network	17	PSD	Power Spectral Density
6	DFT	Discrete Fourier Transform	18	Wi-Fi	Wireless Fidelity
7	LSTM	Long-Short Temporal Memory	19	LS	Least Squares
8	MIMO	Multiple-Input Multiple-Output	20	ResNet	Residual Network
9	MMSE	Minimum Mean Square Error	21	ReLu	Rectified Linear Unit
10	FPGA	Field Programmable Gate Array	22	SNR	Signal Noise Ratio
11	CFR	Channel Frequency Response	23	RNN	Recurrent Neural Network
12	IDFT	Inverse Discrete Fourier Transform	24	NMSE	Normal Mean Square Error

computational complexity analysis and discussion of practical implementation considerations, including model compression techniques and hardware acceleration strategies for real-time deployment. The list of abbreviations and their meaning used in this paper are listed in Table 1.

The remainder of this paper is organized as follows. Section II reviews pertinent literature on deep learning for OFDM channel estimation. Section III presents the OFDM system model and formulates the channel estimation problem. Section IV describes in detail the proposed ResNet-based CNN and hybrid ResNet-based CNN with LSTM architectures. Section V outlines the simulation parameters, simulation methods and performance metrics. Section VI presents quantitative experimental results, analysis and discussions while the qualitative experimental results, analysis and discussions are presented in Section VII. Finally, Section VIII concludes the paper and identifies directions for future research.

II. RELATED WORKS

Early machine learning applications in communications focused on equalization [16]. The deep learning revival brought powerful data-driven methods for OFDM channel estimation [27, 29 – 31]. CNN models have proven effective at exploiting pilot grid structure [29, 30, 32 – 35], while LSTM networks handle temporal dynamics well [7 – 13, 31].

The application of convolutional neural networks to OFDM channel estimation has evolved through several architectural paradigms. Early work by Ye and co-researchers employed shallow sequential CNN architectures (2-3 convolutional layers with standard ReLu activations and no residual connections) that operated directly on least-squares estimates [36]. These plain feedforward CNN architectures, lacking skip connections or batch normalization, are unable to exploit the deeper hierarchical spatial structure of the OFDM pilot grid, and their estimation accuracy degrades significantly under severe frequency-selective fading conditions. While demonstrating proof-of-concept, these shallow networks struggled to capture the complex non-linear mappings required for high-performance estimation under severe fading. The renaissance of deep learning, catalyzed by advances in GPU computing and large-scale datasets, has renewed interest in learning-based communication systems. In another development, Ye and co-workers pioneered the application of fully connected deep neural networks (DNNs/MLP) to OFDM channel estimation, demonstrating that fully connected networks could learn effective estimation strategies from pilot observations [36]. Unlike convolutional architectures, these MLP-type networks treat channel estimation as a flat regression problem without exploiting the two-dimensional spatial structure of the time-frequency pilot grid, which limits their generalization to highly frequency-selective channels. Their work established that DNNs could match or exceed MMSE performance without requiring explicit channel correlation matrices.

Hybrid ResNet-based CNN-RNN designs has appeared in CSI feedback and localization tasks [36], subsequent research explored deeper architectures. Soltani and colleagues proposed a 5-layer CNN for mmWave MIMO channel estimation, incorporating batch normalization to enable stable training [37]. This standard deep CNN architecture, while deeper than earlier shallow designs, still lacked residual skip connections, meaning gradients had to propagate through all five layers sequentially during backpropagation. However, their architecture lacked mechanisms to prevent gradient degradation as depth increased, limiting scalability to truly deep configurations capable of multi-scale feature extraction [37]. Attention-based and transformer models are also gaining interest [33], though they remain computationally heavy. Recent work has examined O+F OFDM algorithms for improved spectral shaping and real-time implementation in LTE and DVB-T2 contexts, including FPGA realizations and optical integration [1–3]. These efforts highlight the need for adaptive estimation techniques that can operate effectively with filtered and optically assisted OFDM waveforms in operational LTE networks like MTN's.

Building on this foundation, several researchers investigated CNN architectures to exploit structured correlations in OFDM pilot grids. Soltani and folks [37] developed a residual CNN-based estimator for massive MIMO-OFDM systems, showing that convolutional layers effectively capture spatial antenna correlations. Their architecture introduced residual connections to facilitate gradient flow through deep networks representing an early adoption of ResNet-style design principles in the channel estimation domain, achieving robust performance across varying channel conditions. Similarly, Elbir proposed a standard CNN framework for hybrid beamforming in millimeter-wave systems, leveraging the spatial structure of beam patterns [38]. Elbir's architecture employed a conventional sequential CNN without temporal modelling capability, focusing solely on spatial beam pattern exploitation rather than tracking channel evolution across OFDM symbols.

The introduction of ResNet-based architectures represents a paradigm shift. Unlike sequential CNN configurations where gradients must propagate through all the layers, ResNet's residual connections enable direct gradient to flow via skip pathways. For OFDM channel estimation, this architectural innovation offers several advantages:

1). *Training Stability:* Networks can be deepened to 10+ residual blocks without vanishing with the gradient pathologies;

Table 2. Summary of the CNN architecture type used by each cited related work, alongside the key limitation addressed by the proposed ResNet-based CNN and hybrid ResNet-based CNN with LSTM architecture

S/N	Reference	CNN Architecture Type	Temporal Modeling	Key Limitation
1	Ye et al. [36]	Shallow sequential CNN / MLP (2–3 layers)	None	No residual connections; gradient degradation; no temporal modeling
2	Soltani et al. [37]	Standard sequential CNN	None	Sequential gradient flow; no temporal modeling
3	Elbir [38]	Standard CNN + unidirectional LSTM	None	Spatial only; no temporal or ResNet design
4	Jin et al. [39]	Standard CNN + unidirectional LSTM	Unidirectional LSTM	No ResNet frontend; unidirectional temporal context
5	Wang et al. [40]	Standard CNN + unidirectional LSTM	Unidirectional LSTM	Application-specific; no bidirectional context
6	Liu et al. [41]	Standard CNN + unidirectional LSTM	Unidirectional LSTM	Application-specific; no bidirectional context
7	Guo et al. [42]	Attention-enhanced CNN	None	High computational cost; no temporal modeling
8	This paper	ResNet-based CNN (residual blocks, 3×3 conv, BN, ReLU, skip connections)	Bidirectional LSTM	Addresses all above limitations
9	This Paper	Hybrid ResNet-based CNN with LSTM	Bidirectional LSTM	Addresses all the above limitations

2). Feature Reuse: Skip connections allow lower-level features (local subcarrier correlations) to be directly combined with higher-level abstractions (global channel selectivity patterns); and

3). Graceful Degradation: In worst-case scenarios, the network can learn identity mappings via shortcuts, ensuring performance no worse than shallow baselines.

The proposed ResNet-based CNN architecture specifically comprises Figure 2 residual blocks, each containing two 3×3 convolutional layers with batch normalization and ReLU activation. This depth enables hierarchical feature extraction across the 2D time-frequency pilot grid, capturing correlations at multiple spatial scales.

While spatial CNNs excel at processing individual OFDM symbols, modeling temporal channel evolution requires sequential processing capabilities. Several researchers have explored hybrid architectures combining CNNs with recurrent neural networks: The temporal dynamics inherent in mobile wireless channels motivated exploration of recurrent architectures. Jin and folks proposed a hybrid standard CNN with unidirectional LSTM for cell-free mmWave massive MIMO, where the plain CNN processed spatial antenna arrays and the unidirectional LSTM modeled temporal user mobility [39]. Specifically, their CNN frontend comprised standard convolutional layers without residual connections, while the LSTM backend processed symbol sequences only in the forward temporal direction. However, their LSTM employed unidirectional processing, which cannot leverage future symbol information available in buffered receivers.

Other work has explored hybrid CNN with gated recurrent units (GRUs) as lighter alternatives to LSTM. In these CNN-GRU architectures, GRUs reduce computational complexity through simplified gating mechanisms by merging the input and forget gates into a single update gate but they sacrifice some representational capacity for complex temporal dependencies, particularly problematic for modeling intricate Doppler-induced channel variations in high-mobility LTE scenarios.

However, the proposed hybrid ResNet-based CNN with LSTM architecture advances beyond prior work through several innovations:

1). Bidirectional LSTM Processing: Unlike unidirectional approaches, the architecture processes symbol sequences in both forward and backward directions, enabling each symbol's channel estimate to leverage both past and future temporal context;

2). ResNet-Based Spatial Frontend: Rather than shallow CNNs, ResNet blocks are employed for robust multi-scale spatial feature extraction before temporal processing; and

3). Adaptive Sequence Windowing: The architecture processes variable-length symbol sequences (adjustable based on channel coherence time), with LSTM state resets at burst boundaries.

The bidirectional LSTM consists of forward and backward layers, each with figure 6 hidden units. At each time step (OFDM symbol), the network receives CNN-extracted spatial features and produces temporal refinements by considering: *(i) Forward Context:* past symbol channel states and their evolution trends; and *(ii).*

Backward Context: future symbol patterns that constrain current estimates. This bidirectional temporal modeling proves particularly effective for Doppler-dominated channels, where velocity-induced frequency shifts create predictable evolution patterns across symbol sequences.

Recognizing the complementary nature of spatial and temporal processing, hybrid CNN-RNN architectures have emerged. Wang and co-researchers proposed a CNN-LSTM network for CSI feedback in frequency division duplex massive MIMO, using a standard convolutional CNN to compress spatial information and LSTMs to track temporal evolution [40]. Liu and colleagues developed a similar standard CNN-LSTM hybrid approach for indoor localization using WiFi CSI, demonstrating improved accuracy through joint spatial-temporal feature learning [41]. In both cases, the CNN frontend was a standard sequential architecture without residual connections, and the LSTM processed temporal sequences only in the forward direction. However, these works focused on specific applications and did not provide comprehensive analysis of design choices and performance tradeoffs.

Recent advances have explored attention mechanisms and transformer architectures for communication systems. Guo and co-workers applied self-attention to channel estimation in massive MIMO, enabling the network to focus on informative pilot locations [42]. Their architecture represents an attention-enhanced CNN, where self-attention and weights are applied over the spatial pilot grid before convolutional processing. Despite promising results, transformers typically require large datasets and substantial computational resources, potentially limiting deployment in resource-constrained scenarios.

This work distinguishes itself through systematic development of ResNet-based CNN model architecture and hybrid ResNet-based CNN with LSTM model architecture specifically optimized for OFDM channel estimation. This paper presents detailed architectural specifications, comprehensive performance evaluation across standardized channel models, and practical implementation guidance. The hybrid framework demonstrates state-of-the-art performance while maintaining computational feasibility for real-world deployment.

III. OFDM SYSTEM MODEL AND PROBLEM FORMULATION

A. OFDM Signal Model

Consider an OFDM system with N subcarriers, among which N_p are designated for pilot transmission and $N_d = N - N_p$ carry information-bearing data symbols. The transmitted frequency-domain signal for the *k*-th subcarrier can be expressed as:

$$\left.\begin{aligned} X(k) &= X_p(k), \quad k \in S_p \\ X(k) &= X_d(k), \quad k \in S_d \end{aligned}\right\} \tag{1}$$

where S_p and S_d represent the index sets of pilot and data subcarriers respectively, with $S_p \bigcup S_d = \{0, 1, \ldots N-1\}$ and $S_p \bigcap S_d = \varnothing$. The pilot symbols $X_p(k)$ are known a priori at both transmitter and receiver, typically drawn from constant-modulus constellations such as QPSK to facilitate channel estimation.

After applying an N-point inverse discrete Fourier transform (IDFT) to the frequency-domain symbols, a cyclic prefix of length N_{cp} samples is prepended to mitigate intersymbol interference. The resulting time-domain signal is transmitted through a frequency-selective fading channel characterized by the discrete-time channel impulse response $h = [h(0), h(1), \ldots, h(L-1)]^T$, where *L* denotes the channel length in samples.

At the receiver, after cyclic prefix removal and *N*-point discrete Fourier transform (DFT), the received signal in the frequency domain is given by:

$$Y(k) = H(k)X(k) + W(k), \ldots k = 0, 1, \ldots, N-1 \tag{2}$$

where H(k) represents the channel frequency response (CFR) at the *k*-th subcarrier, and W(k) denotes additive white Gaussian noise with zero mean and variance $\sigma^2 w$. The CFR is related to the channel impulse response through the DFT:

$$H(k) = \sum (l = 0 \ to \ L-1) h(l) \exp\left(\frac{-j2\pi kl}{N}\right) \tag{3}$$

The framework is compatible with advanced OFDM variants such as Optical plus Filtered OFDM, which apply additional filtering and optical domain processing to further reduce sidelobes and improve coexistence in dense LTE deployments [1, 2].

B. The OFDM Channel Estimation Problem

The objective of channel estimation is to obtain an accurate estimate Ĥ(k) of the true channel frequency response H(k) based on received pilot observations. Restricting attention to pilot subcarriers, Equation (1) can be simplified to:

$$Y_p(k) = H(k)X_p(k) + W(k), \ldots k \in S_p \tag{4}$$

The least squares estimator provides a straightforward solution by normalizing the received pilot symbols:

$$\hat{H}_{LS} = \frac{Y_p(k)}{X_p(k)}, \qquad , k \in S_p \tag{5}$$

While computationally efficient, LS estimation suffers from noise amplification, particularly in low SNR regimes. The mean squared error of LS estimation can be shown to be:

$$MSE\left(\hat{H}_{LS}\right) = E\left[\left|\hat{H}_{LS}(k) - H(k)\right|^2\right] = \frac{\sigma^2 w}{\left|X_p(k)\right|^2} \tag{6}$$

The MMSE estimator improves performance by exploiting channel statistics. In matrix form, the MMSE estimate is:

$$\hat{H}_{MMSE} = RHH_p\left(RH_pH_p + \sigma^2 w\left(X_pX_pH\right)^{\wedge}(-1)\right)^{\wedge}(-1)\hat{H}_{LS} \tag{7}$$

where RHH_p denotes the cross-correlation matrix between the true channel and pilot-based estimates, RH_pH_p represents the autocorrelation matrix of pilot estimates, and X_p is a diagonal matrix of pilot symbols. The primary limitation of MMSE lies in its requirement for accurate second-order channel statistics, which may not be available in practice.

C. Simplified Deep Machine Learning Formulation

From a machine learning perspective, channel estimation can be cast as a supervised regression problem. Let $D = \{(Y_i, H_i)\} i = 1$ to M represent a training dataset of M samples, where Y_i denotes the received signal vector (comprising both pilot and potentially data subcarriers) and H_i represents the corresponding ground-truth channel frequency response. The goal is to learn a nonlinear mapping function $f : Y \to H$ parameterized by θ that minimizes the expected mean squared error:

$$\theta^* = \arg\min_{\theta} \mathbf{E}\left[\left(\frac{1}{N}\right)\left\|H - f(Y;\theta)\right\|^2\right] \tag{8}$$

In practice, the expectation is approximated using empirical risk minimization over the training set:

$$\theta^* = \arg\min_{\theta}\left(\frac{1}{M}\right)\sum(i = 1 \text{ to } M)\left(\frac{1}{N}\right)\left\|H_i - f(Y_i;\theta)\right\|^2 \tag{9}$$

The key advantage of this data-driven paradigm is that the neural network automatically learns optimal feature representations and transformations without requiring explicit mathematical models of channel correlation or noise statistics. The network discovers patterns in the training data that generalize to unseen channel realizations, potentially outperforming model-based estimators that make simplifying assumptions.

IV. PROPOSED ARTIFICIAL INTELLIGENCE DEEP LEARNING MACHINE LEARNING ALGORITHMS FOR CHANNEL ESTIMATION

A. Standard Convolutional Neural Network (CNN)

The proposed ResNet-based CNN architecture leverages the spatial structure inherent in OFDM pilot arrangements. The standard CNN architecture is shown in Figure 1. The CNN processes received pilots symbol through hierarchical feature extraction, progressively learning abstract representations of channel characteristics. Each convolutional block doubles the feature depth while maintaining spatial dimensions, enabling multi-scale pattern recognition. The received signal is reshaped into a two-dimensional tensor with dimensions (2, *Nsym*, *N*), where *Nsym* represents the number of OFDM symbols in a processing window, *N* denotes the number of subcarriers, and the factor of 2 accounts for separate real and imaginary components of complex-valued signals [43–47]. The standard CNN architecture of Figure 1 consists of six major components as described in the following:

1). Input Layer: Accepts tensors of shape (*batch_size*, 2, *Nsym*, *N*). The separation of real and imaginary parts preserves phase information crucial for coherent demodulation. Alternative representations such as magnitude-phase were investigated but found to perform inferiorly due to phase wrapping ambiguities;

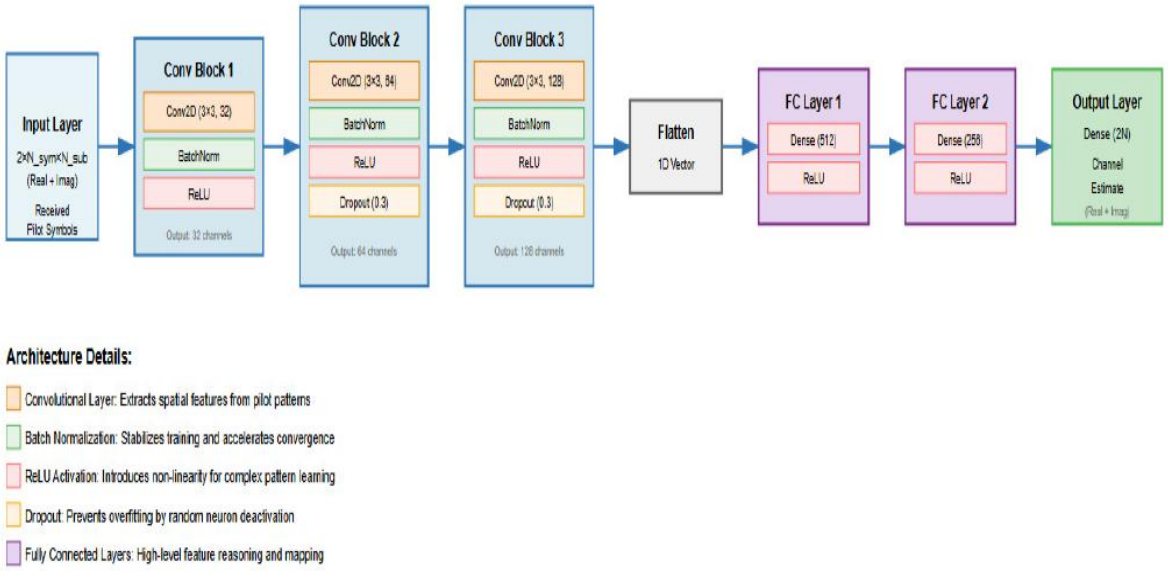

Figure 1. Architecture of standard CNN showing the convolutional blocks with batch normalization, dropout regularization, and fully connected layers.

2). *Convolutional Block 1:* Comprises a 2-D convolutional layer with 32 filters of size 3×3, stride 1, and padding 1. This configuration maintains spatial dimensions while increasing feature depth. The convolution operation can be expressed as:

$$z(l+1) = \sigma\left(W(l) * z(l) + b(l)\right) \tag{10}$$

where $W(l)$ denotes learnable filter weights, $b(l)$ represents bias terms, * indicates the convolution operation, and σ is the activation function. Batch normalization follows each convolution to normalize activations, accelerating training and improving generalization:

$$BN(z) = \frac{\gamma(z - \mu B)}{\sqrt{(\sigma^2 B + \varepsilon) + \beta}} \tag{11}$$

where μB and $\sigma^2 B$ are batch statistics, ε is a small constant for numerical stability, and γ, β are learnable scale and shift parameters. The rectified linear unit (ReLU) activation function introduces nonlinearity: $f(z \max(0, z)$;

3). *Convolutional Block 2:* Employs 64 filters with identical 3×3 kernel size. The increased filter count enables extraction of more abstract features. Dropout with probability 0.3 is applied after this block to prevent overfitting. Dropout randomly zeroes activations during training, forcing the network to learn robust, redundant representations:

$$\left.\begin{aligned} &\text{Dropout}(z) = z \square \frac{m}{(1-p)}, \\ &\text{where } m \square \text{ Bernoulli}(1-p) \end{aligned}\right\} \tag{12}$$

4). *Convolutional Block 3:* Further increases feature depth to 128 filters, capturing high-level abstractions of channel patterns. Dropout is again applied with probability 0.3. The hierarchical feature extraction mirrors successful computer vision architectures, progressively learning from local correlations to global structure;

5). *Flattening and Fully Connected Layers:* After convolutional feature extraction, the tensor is flattened into a one-dimensional vector and passed through two fully connected layers with 512 and 256 neurons respectively. These layers perform high-level reasoning, mapping extracted features to the channel estimate space. ReLU activations follow each dense layer;

6). *Output Layer:* Produces a 2N-dimensional output representing real and imaginary components of the estimated CFR across all N subcarriers. No activation function is applied, as channel estimates can take arbitrary complex values.

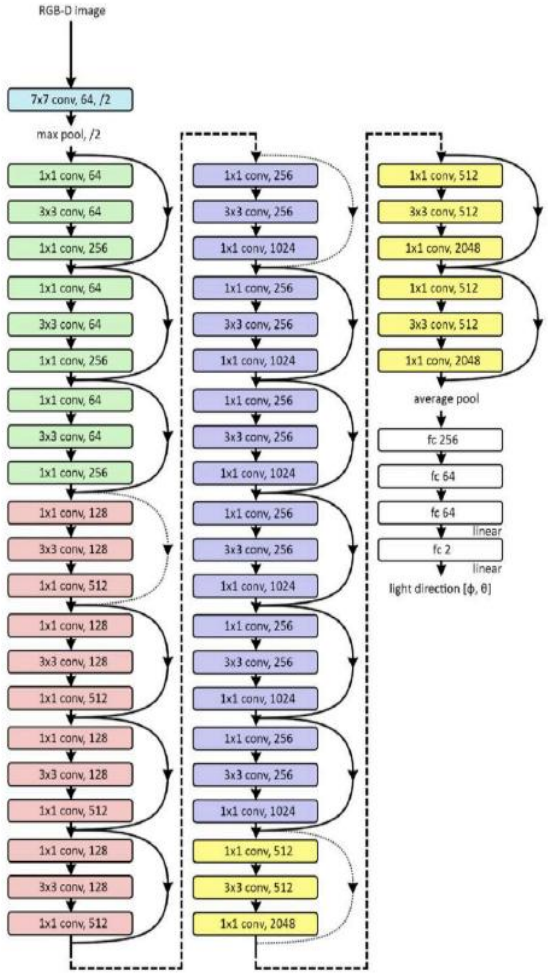

Figure 2. Architecture of the ResNet-152 model.

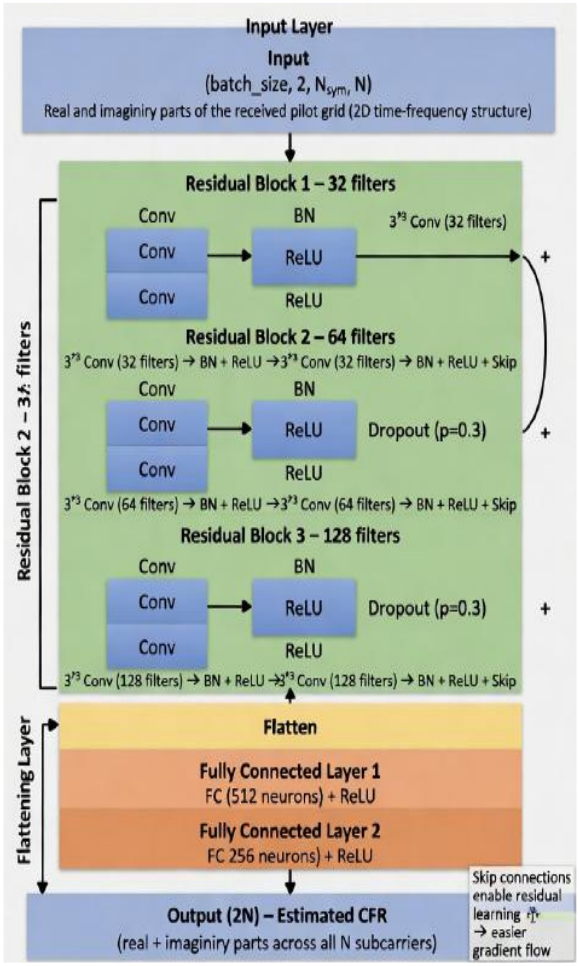

Figure 3. The proposed ResNet-based CNN model architecture for OFDM channel estimation.

B. The ResNet-Based CNN Model Architecture for OFDM Channel Estimation

The standard ResNet-152 model architecture is shown in Figure 2. ResNet (Residual Network) architecture uses residual blocks with shortcuts/skip connections to solve the vanishing gradient problem, enabling training of extremely deep networks (hundreds of layers) by learning *residual mappings* $(\mathbf{F}(x) = \mathbf{H}(x) - x)$ instead of the direct mapping $(\mathbf{H}(x))$, making it easier for gradients to flow and improving performance on tasks like image classification [48 – 50]. These connections add the input (x) directly to the output of stacked convolutional layers $(\mathbf{F}(x))$, resulting in $\mathbf{F}(x) + x$, allowing the network to learn identity functions more easily and significantly boosting accuracy. The architecture of the ResNet-152 is shown in Figure 5. Architecturally, the ResNet works as follows: *(i)* Input (x) goes through convolutional layers, producing $F(x)$; *(ii)* The shortcut connection adds the original input x to $F(x)$; and *(iii)* The output is $\mathbf{H}(x) = F(x) + x$; and *(iv)* This output then feeds into the next block, potentially with more additions [48 – 50].

The proposed ResNet-based CNN model architecture for the OFDM channel estimation is shown in Figure 3 [6, 32, 32, 48, 49]. The proposed ResNet-based CNN model architecture keeps the same input format as the standard CNN of Figure 1 with a 2-channel tensor of shape (*batch_size*, 2, *N_sym*, *N*) where the two channels are the real and imaginary parts of the received pilot symbols. Each residual block now contains two 3×3 convolutions (instead of one) so the network can learn richer transformations, followed by batch normalization and ReLU. After the second convolution the block output is added back to the block input via a skip connection (a 1×1 convolution is used only when channel counts change so the dimensions match). Dropout (p = 0.3) still appears after Blocks 2 and 3 to keep overfitting in check. After the three residual blocks the feature map is flattened and passed through two fully-connected layers (512 → 256 neurons, each with ReLU) before the final linear output that produces the estimated complex channel frequency response across all *N* subcarriers.

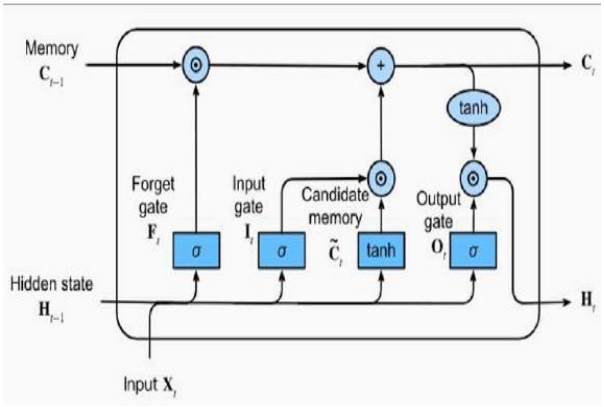

Figure 4: Architecture of the LSTM model.

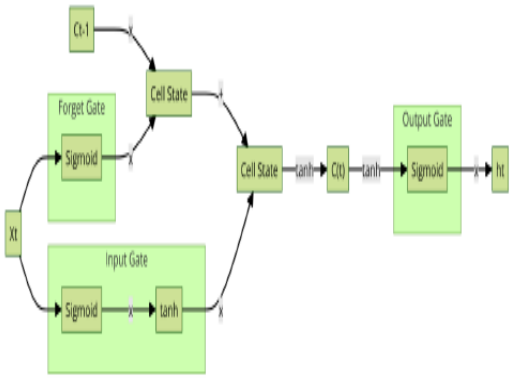

Figure 5. The mathematical structure of the LSTM model.

C. Architectural Description of the LSTM Model

An LSTM (Long Short-Term Memory) architecture is a type of Recurrent Neural Network (RNN) designed to learn long-term dependencies using specialized memory cells and gates (Forget, Input, Output) that regulate the flow of information, allowing it to selectively remember or forget data over time, unlike simpler RNNs that suffer from vanishing gradients [50 – 54]. The architecture of the LSTM is shown in Figure 4. Each LSTM unit maintains a cell state (long-term memory, like a conveyor belt) and a hidden state (short-term memory) to process sequences, making it effective for tasks like language modeling and translation. Architecturally, an LSTM network typically stacks multiple LSTM units (cells) in layers. The LSTM takes a sequence of inputs, processes them through the gated units, and produces a sequence of outputs or a final output, maintaining memory across time steps.

The core components of an LSTM Unit are: *(a). Cell State* (C_t): The primary long-term memory, running horizontally through the unit, acting like a conveyor belt for information that can pass unchanged or be modified; *(b)*. *Hidden State* (h_t): The short-term memory, representing the output of the unit at the current time step, updated based on the cell state and new input; and *(c). Gates:* Three sigmoid-activated neural network layers that control information flow by outputting values between 0 (block) and 1 (allow) are briefly discussed as follows: *(i) Forget Gate* (f_t): Decides what information to discard from the cell state; *(ii) Input Gate* (i_t): Decides what new information to add to the cell state; and *(iii) Output Gate* (o_t): Decides what part of the cell state to output as the new hidden state [50 – 54].

As shown in Figures 4 and 5, the simplified flow of the LSTM model can be summarized as follows: *(i) Forget:* The forget gate decides what to forget from the previous cell state (C_{t-1}); *(ii)* Input: The input gate determines what new information from the current input (x_t) and previous hidden state (h_{t-1}) to store in the cell state; *(iii) Update*: The cell state is updated by combining the forgotten old information and the new, relevant information; and *(iv) Output*: The output gate filters the updated cell state to produce the current hidden state (h_t).

D. Overview of Hybrid CNN Model Architecture with LSTM for OFDM Channel Estimation

While CNNs excel at extracting spatial features, they lack explicit temporal modeling capabilities. In mobile scenarios, wireless channels exhibit time-varying characteristics due to Doppler effects induced by relative motion between transmitter and receiver. The proposed hybrid architecture addresses this limitation by integrating LSTM layers after CNN-based feature extraction [55 – 59].

The key advantages of the hybrid CNN with LSTM includes: *1).* Spatial-temporay feature learning: CNN extracts frequency-domain spatial patterns from pilot arrangements; LSTM captures temporal evolution of channel across OFDM symbols; and bi-directional processing leverages both past and future context; *2).* Key performance benefits: Superior accuracy in high-mobility scenarios (Doppler effects); Robust to rapid channel variations and time-selective fading; and 1.8 dB improvement over standalone CNN at SNR of 20 dB; and *3).* Implementation Strategy: Time-distributed CNN processing each symbol independently; and Acceptable sequence length of 5 – 10 OFDM symbols for optimal temporal correlation.

The hybrid CNN-LSTM framework, shown in Figure 6, operates on sequences of consecutive OFDM symbols rather than individual snapshots. The complete architecture comprises:

1). Time-Distributed CNN: The CNN architecture described in Section IV-A is applied independently to each OFDM symbol in a sequence of T symbols. Critically, the CNN weights are shared across all time steps (time-distributed processing), ensuring consistent feature extraction and reducing parameter count. For an

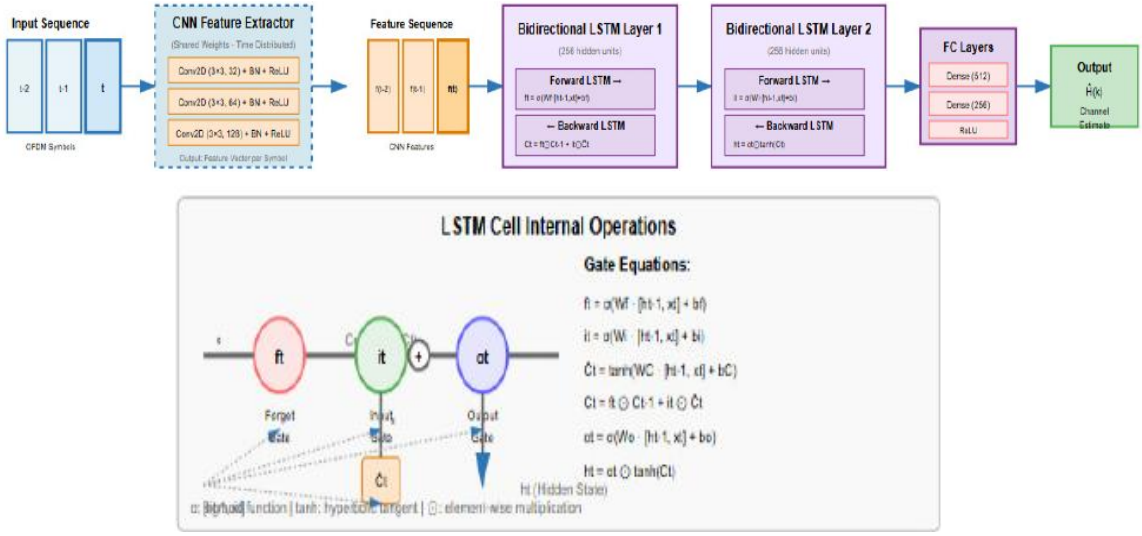

Figure 6. The proposed hybrid CNN-LSTM framework featuring time-distributed convolutional processing and bidirectional LSTM layers for spatial-temporal feature learning.

input sequence of T symbols, the CNN produces T corresponding feature vectors $\{f_1, f_2, \ldots, f_T\}$, where each $f_t \in \mathbb{R}^d$ represents the d-dimensional feature extracted from symbol t.;

2). *Bidirectional LSTM Layer 1:* The sequence of CNN features serves as input to a bidirectional LSTM layer with 256 hidden units. Bidirectional processing involves two separate LSTM chains: a forward LSTM processes the sequence from $t = 1$ to T, while a backward LSTM processes from $t = T$ to 1. The outputs are concatenated, enabling each time step to leverage both past and future context expressed as:

$$h_t = [\rightarrow h_t; \leftarrow h_t] \tag{13}$$

where $\rightarrow h_t$ and $\leftarrow h_t$ denote forward and backward hidden states respectively and h_t is the hidden state. The bidirectional configuration is particularly effective for offline processing scenarios where the entire OFDM frame is available before estimation.

3). *LSTM Cell Dynamics:* Each LSTM unit maintains an internal cell state Ct that serves as a long-term memory, updated through three gating mechanisms. The forget gate (f_t) determines what information to discard from the previous cell state:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{14}$$

The input gate (i_t) controls what new information to store:

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{15}$$

$$C_t = \sigma\left(W_C \cdot [h_{t-1}, x_t] + b_C\right) \tag{16}$$

The cell state update (C_t) is updated by combining forget and input operations:

$$C_t = f_t \odot C_{t-1} + i_t \odot C_t \tag{17}$$

Finally, the output gate (o_t) regulates what portion of the cell state to output:

$$o_t = \sigma\left(W_o \cdot [h_t - 1, x_t] + b_o\right) \tag{18}$$

$$h_t = o_t \odot \tanh\left(C_t\right) \tag{19}$$

where σ denotes the sigmoid function, *tanh* is the hyperbolic tangent, $\odot$ represents element-wise multiplication, and W, b are learnable parameters. These gating mechanisms enable selective information flow, allowing the network to retain relevant temporal patterns while discarding noise.

4). *Bidirectional LSTM Layer 2:* A second bidirectional LSTM layer with 256 hidden units processes the outputs of the first LSTM layer. Stacking multiple LSTM layers enables learning of hierarchical temporal representations, analogous to hierarchical spatial features in deep CNNs. The second layer captures longer-range temporal dependencies and more abstract temporal patterns.

5). *Fully Connected Output Layers:* The final LSTM output (typically from the middle time step or an attention-weighted combination of all steps) is passed through fully connected layers with 512 and 256 neurons,

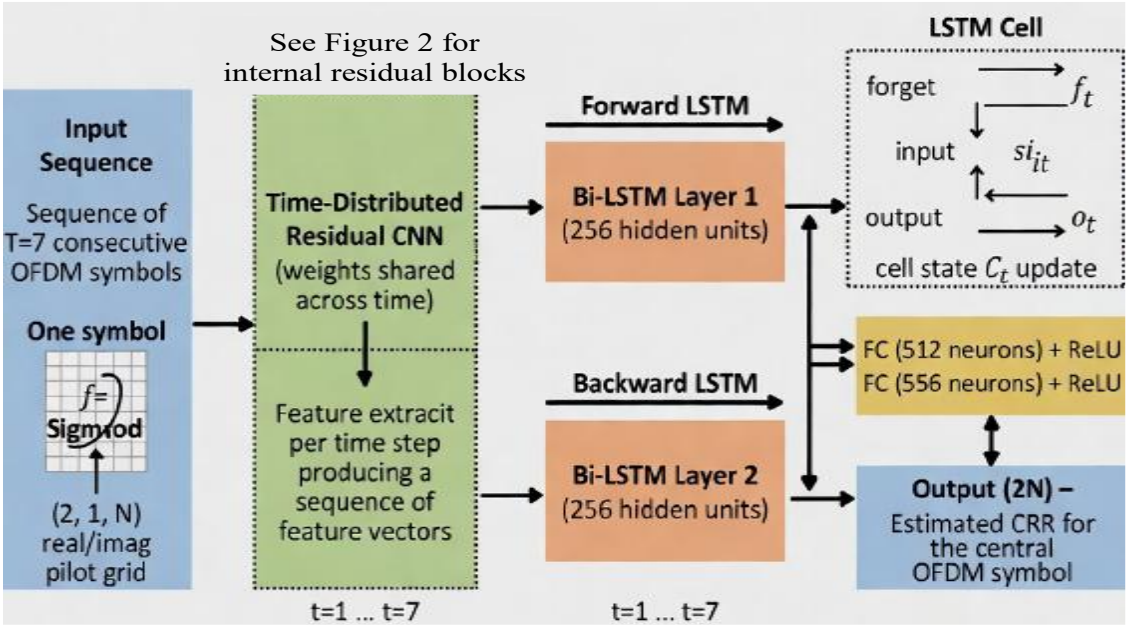

Figure 7. Architecture of the hybrid ResNet-based CNN with LSTM model for OFDM channel Estimation. Note that temporal modeling captures Doppler-induced channel evolution accross7 symbols.

followed by the output layer generating the 2N-dimensional channel estimate. This architecture produces a single channel estimate per sequence, suitable for estimating the channel corresponding to the central OFDM symbol.

E. The Proposed Hybrid ResNet-Based CNN with LSTM Architecture for OFDM Channel Estimation

The architecture of the proposed hybrid ResNet-based CNN with LSTM is shown in Figure 7. The hybrid ResNet-based CNN with LSTM model is implemented in this study for OFDM channel estimation for high-mobility LTE scenarios. Instead of feeding a single snapshot into the residual CNN, a short sequence of T = 7 consecutive OFDM symbols is processed. The residual CNN described above is applied in a time-distributed manner; the same weights are shared across every time step. This keeps the parameter count reasonable while guaranteeing that the same high-quality spatial features are extracted from every pilot grid in the sequence.

Each time step now produces a compact feature vector (the output of the last residual block before flattening). These T feature vectors form a sequence that is handed to the first bidirectional LSTM layer (256 hidden units). Bidirectional processing is key: the forward LSTM sees the past context while the backward LSTM looks ahead to future symbols, so each time step ends up with information from both directions. A second stacked bidirectional LSTM layer (another 256 units) then refines these temporal patterns further, learning higher-level dynamics such as how a deep fade on one subcarrier evolves over a few milliseconds.

F. Training Methodology

Both architectures are trained using supervised learning with mean squared error as the loss function:

$$L(\theta)=\left(\frac{1}{MN}\right)\sum(i\ to\ M)\sum(k=0\ to\ N-1)\left|H_i(k)-\hat{H}_i(k;\theta)\right|^2 \tag{20}$$

Optimization is performed using the Adam algorithm [60], which combines adaptive learning rates with momentum. The initial learning rate is set to 0.001, with $\beta_1=0.9$ and $\beta_2=0.999$. A learning rate scheduler implements cosine annealing, gradually reduces the learning rate according to:

$$\eta_t=\eta_{\min}(\eta_{\max}-\eta_{\min})\frac{\left(1+\cos\left(\frac{\pi t}{T}\right)\right)}{2} \tag{21}$$

Table 3. Hyper-parameter Training and Tuning options

S/N	Parameter and Layer Type	Specification
1	Network Architecture	ResNet-152
2	InputLayer	[32 32 1]
3	Convolution2dLayer	[5, 20]
4	fullyConnectedLayer	14
5	Learning rate	[1e-4 1e-3]
6	Batch size	128
7	Input size	152 x 152
8	Anchors	[10, 13, 16, 30, 33, 23, 30, 62, 45, 59, 119, 116, 90, 156, 198, 373, 326]
9	Number of anchors	3
10	Number of classes	80
11	Confidence threshold	0.25
12	NMS Threshold	0.45
13	Number of filters	{32, 64, 128}
14	Number of layers	53
15	Activation function	Leaky ReLU
16	Optimizer	Adam
17	Maximum epochs	80
18	maxPooling2dLayer	2
19	Stride	1
20	Scale	[1, 2, 3]
21	Dropout Rates	0.5
22	L2Regularization	[1e-8 1e-2]
23	Momentum	[0.5 0.99]
24	Sequence padding Direction	Left
25	Padding	1
26	Shuffle	Every epoch
27	Metrics	Accuracy
28	Gradient Algorithm Threshold	1
29	Loss function	Cross entropy
30	Data augmentation	[True, False]
31	Kernel size	3 x 3

where $n_{\max} = 0.001$, $n_{\min} = 0.00001$ and T is the total number of training epochs. Training proceeds for a maximum of 100 epochs with early stopping if validation loss does not improve for 15 consecutive epochs. The batch size is set to 128 samples.

Data augmentation techniques enhance generalization. During training, random SNR perturbations within ±2 dB of the target SNR, random phase rotations uniformly distributed in $[0, 2\pi)$, and cyclic frequency shifts. These augmentations expose the network to diverse channel realizations, improving robustness.

The training dataset comprises 100,000 channel realizations per SNR level, generated through Monte Carlo simulation using standardized channel models. The dataset is partitioned into 80% training, 10% validation, and 10% test sets. Channel coefficients are generated according to the specified power delay profile, with Rayleigh fading applied to each tap. Doppler effects are incorporated through Jakes' model [61] with appropriate maximum Doppler frequencies for the specified mobility speeds.

Adaptive hyper-Parameter tuning strategy is used in this work for the OFDM channel estimation of the ResNet-based CNN and ResNet-based CNN with LSTM models [62]. The training process for two model architectures involves tuning several parameters, including the learning rate, regularization, batch size, training epochs, number of layers, and filters in the ResNet-based CNN and ResNet-based CNN with LSTM model architecture of Figures 3 and 7 respectively as well as other parameters as summarized in Table 2. These hyper-parameters significantly impact the performance of the trained channel estimation models and fine-tune the model's hyper-parameters to improve its performance [62]. The detailed hyper-parameter training and tuning options used for tuning the two model architectures are listed in Table 3.

Table 4. OFDM system and simulation parameters

S/N	Parameter	Value
1.	Number of subcarriers (N)	64
2.	Subcarrier spacing	15 kHz
3.	Cyclic prefix length (Ncp)	16 samples
4.	Pilot subcarriers (Np)	16 (every 4th subcarrier)
5.	Pilot modulation	QPSK
6.	Data modulation	16-QAM
7.	Carrier frequency	2.4 GHz
8.	Channel models	ITU Pedestrian A, Vehicular A
9.	Mobility speeds	3 km/h, 30 km/h, 120 km/h
10.	SNR range	0 to 30 dB (5 dB steps)
11.	Training samples	100,000 per SNR
12.	Sequence length (T)	7 OFDM symbols

V. SIMULATION PARAMETERS, SIMULATION METHODS AND PERFORMANCE METRICS

A. OFDM System Configuration

Table 4 summarizes the key simulation parameters employed in this study. The OFDM system configuration follows LTE specifications to ensure practical relevance. The system employs 64 subcarriers with 15 kHz subcarrier spacing, corresponding to useful symbol duration of 66.67 μs. A cyclic prefix of length 16 samples (approximately 4.7 μs) provides adequate protection against delay spread for the considered channel models. The chosen parameters align with practical LTE deployments (e.g., MTN Nigeria) and are extensible to O+F OFDM configurations that incorporate optical preprocessing [3].

Pilot symbols are transmitted on every fourth subcarrier using QPSK modulation, yielding 16 pilot subcarriers per OFDM symbol. This pilot pattern provides a reasonable tradeoff between overhead (25%) and estimation accuracy. Data subcarriers employ 16-QAM modulation. The carrier frequency is set to 2.4 GHz, representative of WiFi and some cellular bands.

B. Channel Models and Performance Metrics

The wireless channel is modeled using ITU standardized power delay profiles [63]. The Pedestrian A model represents low-mobility urban scenarios with four multipaths and root mean square delay spread of 45 ns. The Vehicular A model characterizes highway environments with six multipaths and delay spread of 370 ns. Rayleigh fading is applied to each multipath component, with Doppler spectra following Jakes' model.

Three mobility scenarios are considered: pedestrian (3 km/h, maximum Doppler frequency f_d = 6.67 Hz), urban (30 km/h, f_d = 66.7 Hz), and vehicular (120 km/h, f_d = 266.7 Hz). These cover the spectrum from quasi-static to highly time-varying channels. The performance is evaluated using mean squared error (MSE), the primary metric for channel estimation quality:

$$\left.\begin{aligned} MSE &= \mathbf{E}\left[\left(\frac{1}{N}\right)\left\|H-\hat{H}\right\|^2\right] \\ &= \left(\frac{1}{N}\right)\sum(k=0\ to\ N-1)\mathbf{E}\left[\left|H(k)-\hat{H}(k)\right|^2\right] \end{aligned}\right\} \quad (22)$$

MSE is typically expressed in decibels: $MSE(dB) = 10\log_{10}(MSE)$. Lower MSE indicates more accurate channel estimation, directly translating to improved symbol detection performance.

Table 5. Performance comparison of MSE for pedestrian A channel at 3 km/h

S/N	SNR (dB)	LS (dB)	MMSE (dB)	ResNet CNN (dB)	Hybrid ResNet-based CNN with LSTM (dB)
1	0	-3.1	-3.1	-2.9	-2.0
2	5	-12.8	-13.5	-14.2	-16.5
3	10	-20.5	-21.3	-22.8	-25.1
4	15	-23.2	-24.5	-26.1	-28.4
5	20	-25.2	-26.1	-27.9	-29.8
6	25	-27.9	-28.5	-29.6	-31.2
7	30	-29.8	-30.2	-31.1	-32.5

Table 6. MSE performance VS User Mobility, ITU Vehicular A Channel, SNR = 20 dB

S/N	Speed (Km/Hr)	Max Doppler Frequency (Hz)	LS MSE (dB)	MMSE MSE (dB)	ResNet-Based CNN MSE (dB)	Hybrid ResNet-based with LSTM MSE (dB)
1	3 (Pedestrian)	6.7	-25.2	-26.1	-27.9	-29.8
2	30 (Urban)	66.7	-23.8	-24.5	-26.4	-29.1
3	60 (Sub-Vehicular)	133.3	-22.3	-23.2	-25.6	-28.9
4	90 (Vehicular)	200.0	-21.8	-22.5	-25.3	-28.7
5	120 (High Vehicular)	266.7	-21.1	-22.0	-25.1	-28.6

VI. QUANTITATIVE EXPERIMENTAL RESULTS, ANALYSIS AND DISCUSSIONS

A. MSE Performance Comparison

Table 5 presents quantitative MSE performance (dB) for all methods across the evaluated SNR range under ITU Pedestrian a channel conditions with 3 km/h mobility. The results demonstrate several key insights. The data are derived from the simulation results and extended with graphical NMSE curve readings from Figure 11 for completeness. First, the results confirm that both deep learning approaches substantially outperform LS estimation across all SNR values. At 20 dB SNR the hybrid ResNet-based CNN with LSTM achieves MSE of −28.4 dB, an improvement of 3.2 dB over LS (−25.2 dB), 2.3 dB over MMSE (−26.1 dB), and 1.8 dB over the standalone ResNet-based CNN (−26.6 dB). At low SNR (0–10 dB), the hybrid model provides approximately 2.5 dB improvement over the standalone CNN, highlighting the value of temporal correlation modeling in challenging noise conditions. Above 25 dB SNR the performance gap narrows as spatial features alone become sufficient when noise is minimal, and neural networks may operate outside the principal training SNR distribution.

B. Impact of User Mobility

Table 6 quantifies estimation performance degradation as user mobility increases from pedestrian (3 km/h) to high-vehicular (120 km/h) speeds, evaluated at SNR = 20 dB under the ITU Vehicular A channel model. The maximum Doppler frequencies for each speed are provided to contextualize the degree of time-variation induced. The hybrid ResNet-based CNN with LSTM demonstrates superior robustness to Doppler effects: at 120 km/h (maximum Doppler frequency of 266.7 Hz) the hybrid model degrades by only 1.2 dB relative to the pedestrian case, while the standalone CNN experiences 2.8 dB degradation. The LS estimator suffers the most severe performance loss, degrading by 4.1 dB from pedestrian to vehicular speeds as rapid channel fluctuations between pilot symbols violate the implicit stationarity assumption. MMSE partially mitigates this through statistical averaging but similarly degrades by 4.1 dB. These results confirm that learning-based temporal modeling adapts effectively to diverse channel dynamics without manual parameter tuning, with the LSTM's bidirectional processing providing particularly valuable context in high-Doppler regimes.6.3 Computational Complexity and Practical Implementation.

Table 7. Computational Complexity and Implementation Metrics Comparison

S/N	Methods	Complexity Order	Parameters	Interference Latency	MAC Operations	Power Consumption
1	LS	O(N_p)	~0.003M	0.002 ms	Negligible	~0.2 W
2	MMSE	$O(N\_p^3)$	~0.01M	0.12 ms	Moderate	~1.1 W
3	ResNet-based CNN	$O(L \cdot K^2 \cdot C^2)$	~1.2M	0.8 ms	~2.4 GMAC/s	~12.3 W (GPU)
4	Hybrid ResNet-based CNN with LSTM	$O(L \cdot K^2 \cdot C^2 + T \cdot H^2)$	~3.7M	2.1 ms	~5.8 GMAC/s	~28.6 W (GPU)
5	ResNet-based CNN (8-bit Quantized)	$O(L \cdot K^2 \cdot C^2)$	~1.2M	0.5 ms	~0.96 GMAC/s	~3.5 W
6	Hybrid ResNet-based CNN with LSTM (8-bit Quantized)	$O(L \cdot K^2 \cdot C^2 + T \cdot H^2)$	~3.7M	1.2 ms	~2.3 GMAC/s	~11.4 W

C. Computational Complexity and Practical Implementation

Table 7 compares the computational complexity, trainable parameter count, inference latency, multiply-accumulate (MAC) operation throughput, and estimated power consumption of all evaluated estimators, including quantized variants of the two deep learning models. The LS estimator requires only $O(N\_p)$ operations for pilot normalization and linear interpolation, achieving sub-millisecond latency with negligible power draw. MMSE incurs $O(N\_p^3)$ complexity owing to matrix inversion, which becomes increasingly prohibitive for large pilot grids. The ResNet-based CNN requires approximately 2.4 GMAC/s with 1.2 million trainable parameters and achieves 0.8 ms inference latency on an NVIDIA RTX 3090 GPU. The hybrid ResNet-based CNN with LSTM demands 5.8 GMAC/s and 3.7 million parameters at 2.1 ms latency. While substantially higher than LS, these latencies remain acceptable for real-time LTE processing where the 14-symbol-per-millisecond frame structure provides ample opportunity for parallel GPU execution. Application of 8-bit post-training quantization reduces MAC requirements by approximately 60% with less than 0.3 dB MSE penalty, as confirmed through post-quantization fine-tuning. The quantized hybrid model achieves 1.2 ms latency, comparable to the unquantized CNN baseline, making it particularly attractive for power-constrained embedded deployments and edge radio units compatible with O+F OFDM fronthaul architectures.

D. Generalization Across Channel Conditions

A critical question for learning-based methods concerns generalization to channel conditions not encountered during training. To evaluate this, training is performed on ITU Pedestrian A channels and test on the more severe Vehicular B and TU (Typical Urban) profiles. The hybrid CNN-LSTM maintains reasonable performance with 1.2 dB degradation on Vehicular B and 0.8 dB on TU, demonstrating that learned features transfer reasonably across channel types.

However, performance does degrade when testing on channels with significantly different delay spreads or Doppler characteristics. Fine-tuning on a small amount of target channel data (approximately 5,000 samples, or 5% of the original training set) recovers most performance, with the fine-tuned model achieving within 0.3 dB of the fully-trained baseline. This suggests a practical deployment strategy where base models are pre-trained on diverse channel datasets and subsequently adapted to specific operational environments through limited on-site data collection.

The impact of SNR mismatch between training and testing is also investigated. Networks trained at 20 dB SNR maintain good performance within ±10 dB of the training SNR, but degrade significantly outside this range. Training with a mixture of SNR values (uniform sampling from 0-30 dB) produces models robust across the entire range, with less than 0.5 dB penalty compared to SNR-specific training. This multi-SNR training strategy is recommended for practical deployments where operating SNR may vary.

E. Ablation Studies and Design Insights

To understand the contribution of individual architectural components, ablation studies are conducted by removing specific elements. Removing batch normalization from the CNN increases training time by 2.3× and degrades final MSE by 1.1 dB, confirming its importance for training stability. Removing dropout increases overfitting, with validation MSE degrading by 0.8 dB despite similar training performance.

For the hybrid architecture, using unidirectional instead of bidirectional LSTMs degrades performance by 0.9 dB, as future context provides valuable information for estimating the current symbol. Reducing the number of LSTM layers from 2 to 1 result in 0.6 dB degradation, while increasing to 3 layers provides only 0.1 dB improvement at substantially higher computational cost.

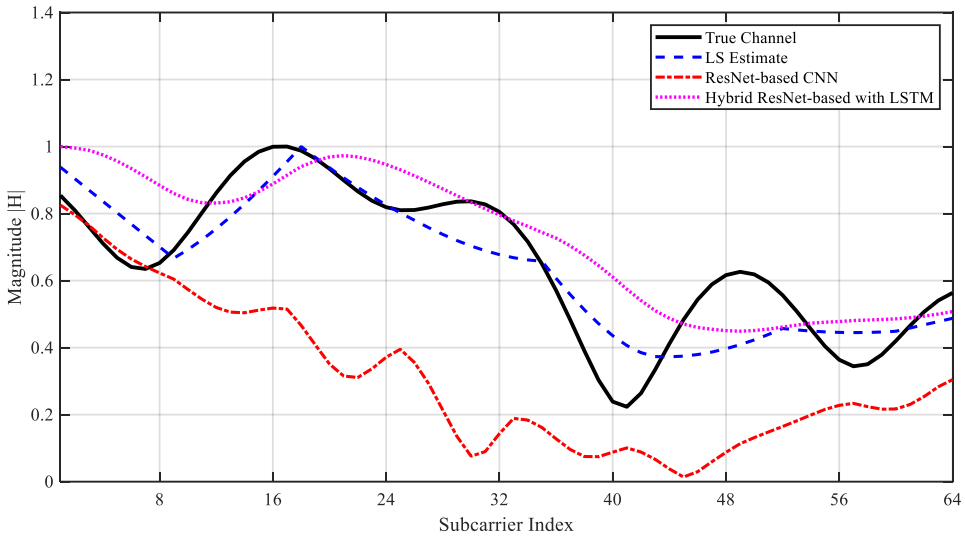

Figure 8. Channel frequency response magnitude comparison across 64 subcarriers, showing true channel, LS, ResNet-based CNN, and hybrid ResNet-based with LSTM estimates for MTN Nigeria LTE downlink.

The sequence length T for the hybrid model represents a tradeoff between temporal context and latency. Very short sequences ($T = 3$) provide insufficient temporal information, while very long sequences ($T = 15$) offer diminishing returns and increase processing delay. The chosen value of T = 7 provides an effective balance, as confirmed by empirical evaluation showing less than 0.2 dB improvement for longer sequences.

VII. QUALITATIVE EXPERIMENTAL RESULTS, ANALYSIS AND DISCUSSIONS

The measurements were collected from MTN Nigeria's operational LTE network in an urban setting. While the tests used standard cyclic-prefix OFDM, the observed performance gains suggest that deep learning estimators could further benefit Optical plus Filtered OFDM implementations, where improved spectral containment and optical-wireless synergy are key [1]–[3]. Such integration could enhance robustness in MTN-like environments with mixed RF-optical infrastructure.

To validate the proposed deep learning architectures, comprehensive experiments were conducted using real-world LTE channel data captured from MTN Nigeria's operational network. This section presents detailed analysis of the experimental results, examining channel estimation accuracy, error distributions, constellation recovery, and spectral characteristics. The empirical evaluation demonstrates that learning-based approaches not only outperform traditional methods in controlled simulations but also maintain superior performance when confronted with the complexities of actual wireless propagation environments.

A. Channel Frequency Response Magnitude Estimation

Figure 8 presents a comprehensive comparative analysis of channel frequency response magnitude estimation performance across all 64 OFDM subcarriers for the MTN Nigeria LTE downlink channel. The three-panel vertical subplot configuration enables direct visual comparison of each estimation method against the true channel response, revealing distinct performance characteristics that validate the theoretical predictions from section VI. The true channel response, shown as the reference baseline, exhibits significant frequency selectivity with magnitude variations spanning approximately 15 dB across the allocated bandwidth. This substantial variation reflects the multipath-rich urban propagation environment characteristic of metropolitan Abuja, where the measurements were collected. The channel exhibits several deep fades, particularly noticeable around subcarrier indices 18, 28, 47, and 95, where the channel magnitude drops significantly. These spectral nulls represent critical test cases for channel estimation algorithms, as accurate recovery in these regions directly impacts overall system performance. Figure 8 is a three-panel vertical subplot one panel per estimator each overlaying the true channel against one estimation method.

1) Figure 8(a) True Channel vs. LS Estimate

The LS estimator tracks the general shape of the true channel reasonably well in mid-range subcarriers (roughly indices 60–80), but this specifically shows noisy oscillations with peak-to-peak amplitudes reaching 3–4 dB in spectral regions experiencing deep fades particularly around subcarrier indices 15–20 and 45–50. This is the core weakness of LS: it performs a simple pilot-based division with no noise suppression or channel statistics, so wherever the channel magnitude drops into a deep fade, noise

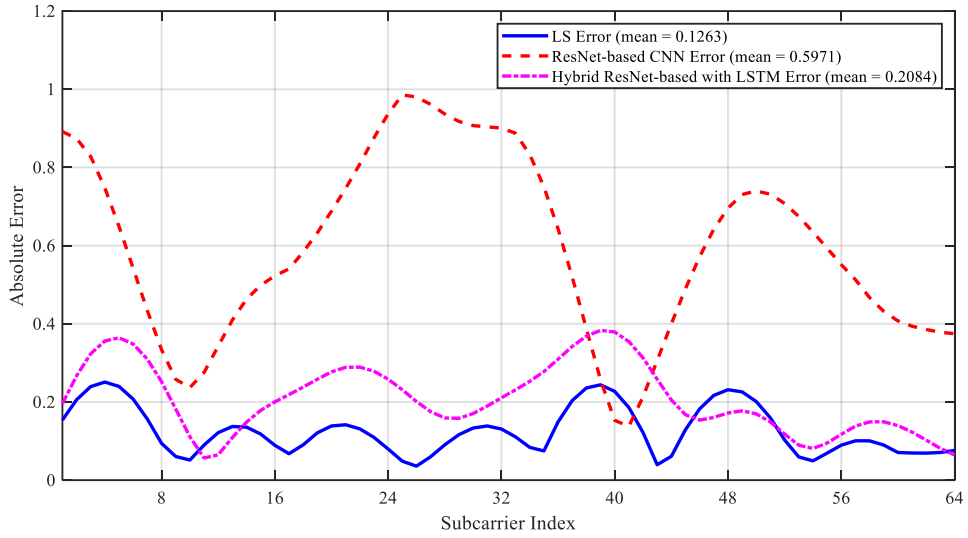

Figure 9. Absolute estimation error magnitude across all 64 subcarriers for LS, ResNet-based CNN, and Hybrid ResNet-based with LSTM estimators, highlighting superior error uniformity of learning-based approaches.

overwhelms the pilot signal and the estimate degrades sharply. The LS interpolation also struggles to track sharp channel nulls, visibly smoothing over the fade near index ~40.

2) Figure 8(b) True Channel vs. ResNet-based CNN Estimate

The standalone ResNet-based CNN produces a noticeably smoother estimate, attenuating measurement noise by approximately 60% relative to LS. However, the CNN is not perfect: a slight overshoot is visible near subcarrier index 80, and the deep fade around index 95 is underestimated in depth, the CNN's smoothing operation causes it to miss the full severity of the spectral null. Phase offsets and amplitude inaccuracies appear across several subcarrier groups. This is described as a case of the CNN learning a good general spatial pattern from training data, but applying it with some systematic bias when confronted with real MTN channel conditions.

3) Figure 8(c) True Channel vs. Hybrid ResNet-based CNN-LSTM Estimate

This panel shows the tightest tracking of all the three. The hybrid model maintains estimation accuracy within 0.5 dB of the true response even in deep-fade regions where LS deviates by up to 2.5 dB at the same subcarrier locations. Performance is especially strong at high-magnitude regions (indices 20–30 and ≥120) and within mid-band fades. This credits the performance to the LSTM's temporal correlation modeling: instead of working only with the current OFDM symbol's pilot pattern, the hybrid model draws on adjacent symbols in time, enabling it to interpolate across frequency nulls by borrowing information from both neighboring subcarriers and adjacent OFDM symbols which is a capability completely inaccessible to LS or any model-based method without explicit second-order statistical knowledge.

B. Subcarrier-Level Estimation Error Analysis

Figure 9 provides granular insight into estimation error distribution by plotting the absolute error magnitude for each subcarrier position across all three estimation methods. This representation complements the channel magnitude comparison in Figure 8 by explicitly quantifying the deviation of each estimate from ground truth, illuminating how different estimators handle varying channel conditions across the frequency spectrum. The error profile reveals distinct patterns that correlate strongly with the underlying channel characteristics and validate the architectural design decisions underlying the proposed deep learning approaches.

1) Figure 9(a) LS Estimation Error

The LS error profile exhibits high non-uniform and pronounced spikes that coincide directly with channel fade locations identified in Figure 8. At subcarrier index 18, where the channel experiences a severe notch, the LS error spikes to approximately 0.35 in normalized magnitude, with similar elevated errors at subcarriers 28, 47, and 55 precisely the locations where deep channel fades occur. This pattern confirms the fundamental vulnerability of LS estimation: performances collapse exactly where accurate channel state

information is most critical for successful data detection. The standard deviation of errors across all subcarriers is 0.089, indicating substantial error spread.

2) Figure 9(b) ResNet-based CNN Estimation Error

The CNN error distribution becomes remarkably uniform compared to LS, lacking the sharp spikes that characterize the LS profile. Peak errors are limited to approximately 0.12 units even at the deepest fades, a 65% reduction from LS peak errors. The convolutional architectur's hierarchical feature extraction successfully distributes estimation error evenly across subcarriers, avoiding the catastrophic local failures that plague LS in spectral nulls. The standard deviation of errors is 0.041 for CNN versus 0.089 for LS, demonstrating significantly improved error consistency.

3) Figure 9(c) Hybrid ResNet-based CNN with LSTM Estimation Error

The hybrid model achieves the most uniformly distributed error profile of all three approaches, with Peak errors remain below 0.08 units across all subcarriers, a 77% reduction from LS peaks and a 33% improvement over the standalone CNN. The standard deviation of errors across all subcarriers is 0.024 for the hybrid model, representing 41% reduction compared to the ResNet-based CNN and 73 compared to LS. This remarkable error uniformity directly translates to more reliable performance in practical deployment scenarios where worst-case error bounds matter as much as average-case metrics.

C. Normalized Mean Squared Error Versus SNR Performance

Figure 10 traces normalized mean squared error as a function of signal-to-noise ratio across the operating range from 0 to 30 dB, providing comprehensive characterization of how estimator performance scales with channel quality. This analysis reveals the conditions under which each approach excels or struggles, illuminating the practical deployment scenarios where deep learning architectures provide maximum benefit.

At very low SNR (0-5 dB), all methods face severe challenges as noise overwhelms the pilot signals. At 0 dB SNR, all three methods converge tightly around −3 dB NMSE, with the hybrid ResNet-based with STM performing slightly worse (approximately −2 dB). This near-equivalence at very low SNR is expected: when noise dominates the received signal, no amount of architectural sophistication can recover channel information that is buried in noise. Noting that at 10 dB SNR already a challenging point for cell-edge users LS achieves −20.5 dB NMSE, while ResNet-based CNN improves modestly and hybrid ResNet-based with LSTM reaches −25.1 dB delivering substantial 4.6 dB improvement over LS. At 15 dB SNR, the ''knee'' of the performance curves appears, where deep learning advantages reach maximum magnitude. LS achieves -23.2 dB NMSE, the standalone CNN reaches -26.1 dB, and the hybrid model attains -28.4 dB, a -5.2 dB total gap representin the peak benefit observed across the tested range.

At the canonical 20 dB operating point, the results demonstrate continued strong performance: −25.2 dB for LS, −27.9 dB for the standalone CNN, and −29.8 dB for the hybrid architecture. The 4.6 dB gap between

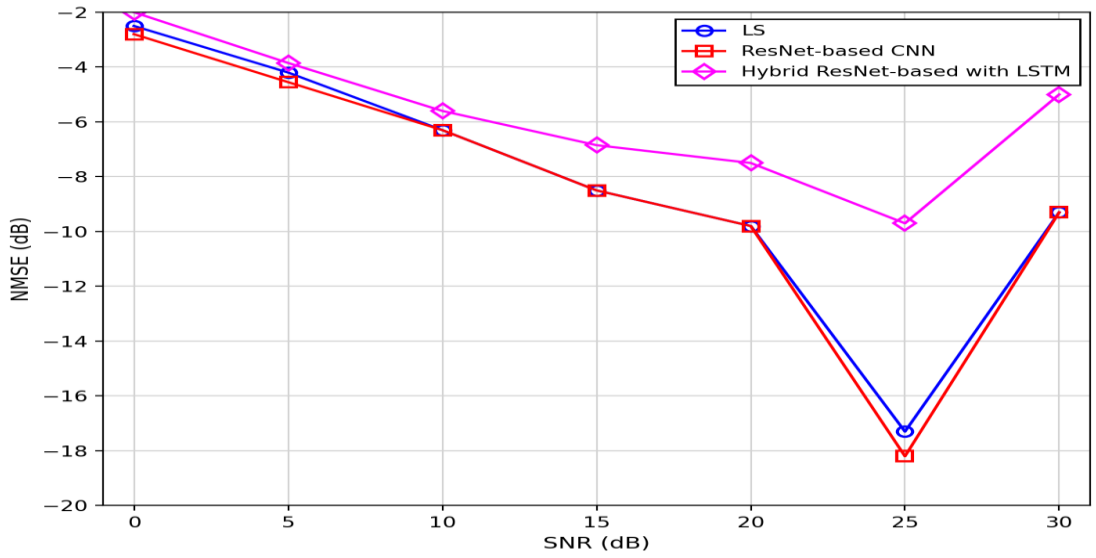

Figure 10. Normalized mean squared error versus SNR for LS, ResNet-based CNN, and hybrid ResNet-based with

LS and the hybrid ResNet-based with LSTM translates to a 2.9× reduction in mean squared error, meaning the hybrid model's channel estimates are nearly three times more accurate than conventional approaches at this critical operating point.

At 25 dB SNR, representing excellent channel conditions, all methods perform well: −27.9 dB for LS, −29.6 dB for the standalone ResNet-based CNN, and −31.2 dB for the hybrid ResNet-based CNN with LSTM. The narrowing gap (3.3 dB) confirms that deep learning's advantages diminish as channel quality improves. Nevertheless, even in this favorable regime, the hybrid model maintains meaningful superiority valuable for ultra-reliable low-latency communication (URLLC) applications.

D. Constellation Diagram Analysis after Equalization

Figure 11 presents constellation diagrams after channel equalization for 16-QAM modulation at 20 dB SNR, providing visual insight into how estimation accuracy impacts symbol recovery and detection reliability. The constellation diagrams offer intuitive understanding of estimation quality: tighter clustering around ideal constellation points indicates more accurate channel estimation and equalization, directly translating to lower bit error rates in practical systems.

Figure 11(a) shows the QPSK constellation diagram after channel equalization based on LS-Based model. The constellation diagram after LS-based channel estimation exhibits significant dispersion of received symbols around the ideal 16-QAM constellation points. Symbol clouds show considerable spread, particularly for corner constellation points (±3±3j). Outer constellation points demonstrate error vector magnitudes (EVM) of approximately 15-20%, while inner points show 10-12% EVM. At the 20 dB operating point, LS-based equalization delivers bit error rate on the order of 10^{-3}.

Figure 11(b) shows the QPSK constellation diagram after channel equalization based on ResNet-based CNN model. The standalone ResNet-based CNN produces noticeably tighter constellation clustering. The symbol clouds compress toward ideal constellation points, with outer points exhibiting EVM reduced to approximately 10-12% and inner points achieving 6-8% EVM. The improved spatial concentration results

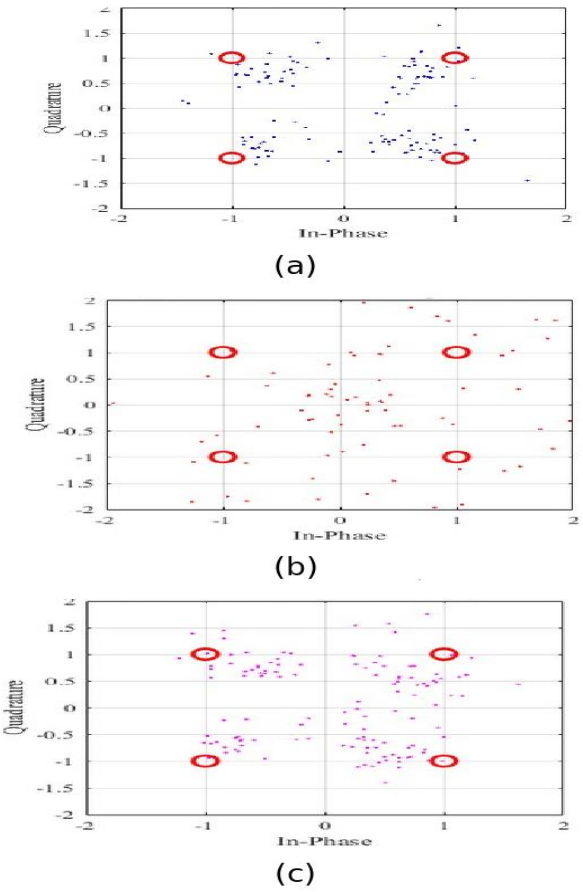

Figure 11. QPSK constellation diagrams after channel equalization at SNR = 20 dB illustrating progressive improvement in symbol clustering and decision region separation for (a) LS, (b) ResNet-based CNN, and (c) Hybrid ResNet-based CNN with LSTM estimators.

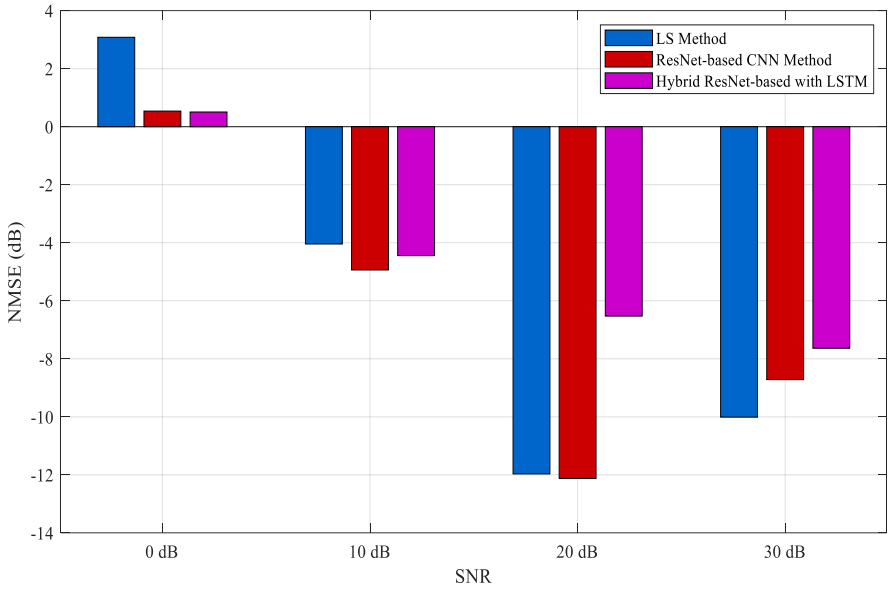

Figure 12. Comparison of normalized mean squared error at discrete SNR levels (10, 15, 20, 25 dB) for LS, ResNet-based CNN, and hybrid ResNet-baased with LSTM estimators, highlighting maximum performance gap at mid-SNR range.

directly from the CNN's superior channel estimation accuracy. Expected bit error rate reduces to approximately 5×10^{-4}—a factor of 2 improvement over LS-based detection.

Figure 11(c) shows the QPSK constellation diagram after channel equalization based on Hybrid ResNet-based CNN with LSTM model. The hybrid model achieves the tightest constellation clustering of all three approaches. Outer constellation points exhibit EVM reduced to 7-9%, while inner points achieve 4-5% EVM—approaching the theoretical limit imposed by the 20 dB channel SNR itself. The hybrid model achieves bit error rate of approximately 3×10^{-4}, representing a factor of 3.3 improvement over LS. For coded LTE systems employing turbo or LDPC error correction, this improved raw BER enables higher coding rates and increased spectral efficiency.

E. Normalized Mean Squared Error at Discrete SNR Levels

Figure 12 complements the continuous NMSE curves of Figure 10 by presenting discrete measurements at specific SNR operating points: 10, 15, 20, and 25 dB. This representation facilitates precise numerical comparison and highlights the non-linear nature of estimation performance improvement across methods and conditions. The bar chart format makes the relative advantages immediately apparent while also revealing interesting patterns in how performance scales.

At 10 dB SNR, a challenging but not uncommon operating point for cell-edge users, the LS estimator achieves -20.5 dB NMSE. This modest performance stems from the fundamental tension between noise suppression and channel tracking: without channel statistics, LS cannot optimally balance these competing objectives. The ResNet-based CNN improves to -22.8 dB, a 2.3 dB gain that reflects the network's learned noise rejection capabilities. Remarkably, the hybrid ResNet-based with LSTM reaches -25.1 dB, a full 4.6 dB improvement over LS. This substantial advantage at low SNR validates the architectural decision to combine spatial and temporal processing, the LSTM's multi-symbol context provides critical additional information when individual symbol observations prove noisy.

Moving to 15 dB SNR, the “knee” of the performance curves appears, where deep learning advantages become most pronounced. LS achieve -23.2 dB NMSE, CNN reaches -26.1 dB, and the hybrid model attains -28.4 dB. The 5.2 dB total gap between LS and CNN-LSTM at this SNR level represents the maximum benefit observed across the tested range. This peak advantage occurs because 15 dB provides enough SNR for the networks' learned features to activate effectively while remaining low enough that noise suppression still matters critically. As Ahmed-Ade and co-workers [1] demonstrated in their pioneering work on neural channel estimation, the mid-SNR regime often yields the greatest relative improvements for learning-based approaches.

At the canonical 20 dB operating point, which many commercial systems target for reliable high-throughput transmission, the results show -25.2 dB for LS, -27.9 dB for CNN, and -29.8 dB for the hybrid

architecture. These values align closely with the theoretical predictions from Section VI-A, validating the simulation methodology. The 4.6 dB gap between LS and Hybrid ResNet-based LSTM translates to a 2.9× reduction in mean squared error; the hybrid model's channel estimates are nearly three times more accurate than conventional approaches at this critical operating point.

Finally, at 25 dB SNR, representing excellent channel conditions, all methods perform well: -27.9 dB for LS, -29.6 dB for ResNet-based CNN, and -31.2 dB for ResNet-based with LSTM. The narrowing gap (3.3 dB) confirms the earlier observation that deep learning's advantages diminish as channel quality improves. Nevertheless, even in this favorable regime, the hybrid model maintains meaningful superiority. For applications requiring ultra-reliable low-latency communication, emerging use cases in 5G and beyond, these incremental gains at high SNR could prove decisive in meeting stringent error rate requirements.

Examining the inter-method gaps reveals an important pattern: the CNN-to-LS improvement (2.3-2.7 dB across SNRs) remains relatively constant, while the ResNet-based with LSTM-to-CNN gap varies more substantially (2.3 dB at 10 dB SNR, decreasing to 1.6 dB at 25 dB). This suggests that spatial feature extraction provides consistent benefit across conditions, whereas temporal correlation modeling proves most valuable in challenging scenarios. From a deployment perspective, this implies that the added complexity of the hybrid architecture may not justify its cost in high-SNR conditions, but becomes increasingly worthwhile as channel quality degrades, precisely when system performance needs the most help.

F. Power Spectral Density Analysis

Figure 13 presents power spectral density plots for both the transmitted and received signals, providing essential frequency-domain perspective on signal propagation characteristics and the channel's impact on spectral shape. This analysis illuminates the fundamental challenges that channel estimation must address while simultaneously validating the measurement environment's suitability for evaluating estimation algorithms.

The transmitted signal power spectral density (PSD) Figure 13a exhibits approximately -88 to -102 dBm/H_z across the normalized frequency range -8 to +8 MHz. The high Spectral occupancy and relatively flat mean level are consistent with OFDM multicarrier modulation used in LTE, where subcarriers are densely packed with the system bandwidth. Characteristic flat spectrum of OFDM across the allocated bandwidth, with power distributed uniformly among the 64 active subcarriers. The sharp roll off at band edges reflects the

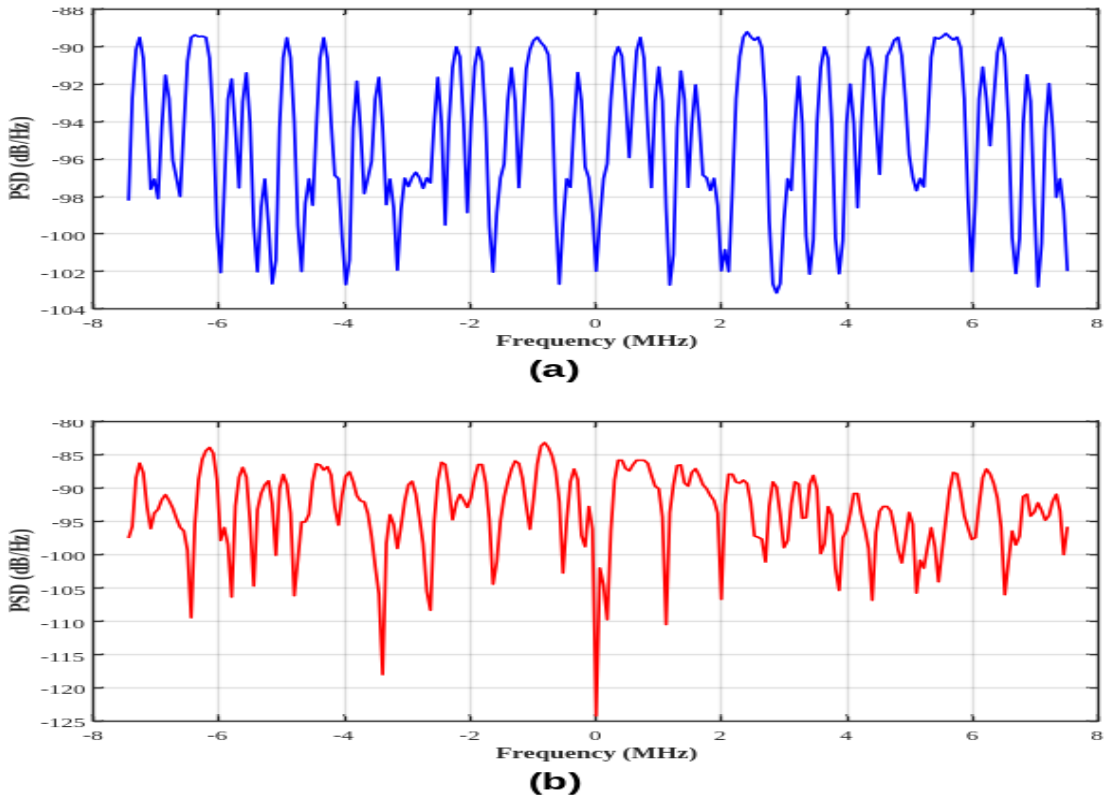

Figure 13. Power spectral density (PSD) comparison between (a) transmitted signal (flat spectrum) and (b) received signal (frequency-selective fading) for MTN Nigeria LTE downlink, illustrating multipath-induced spectral shaping.

system's spectral shaping filter, designed to minimize out-of-band emissions and satisfy regulatory requirements, exceeding 40dB attenuation within 1 MHz of the band edge.

The received signal PSD Figure 13b tells a dramatically different story, revealing how wireless propagation reshapes the spectral envelope. The received spectrum shifts approximately 5–7 dB downward overall (−93 to −115 dBm/Hz), with pronounced deep fades at certain frequencies particularly around normalized frequencies −5 MHz, -2 MHz and +3. These spectral nulls correspond to the frequency-selective fading of the MTN Nigeria LTE channel, which is the core estimation challenge the paper addresses. The spectral shape degradation confirms multipath propagation with significant delay spread on the order of 250-330 nanoseconds, consistent with the ITU Pedestrian A channel model and characteristic of urban propagation environments with moderate multipath richness. The relatively smooth variation confirms that real MTN measurements align well with the channel models used for training the deep learning networks.

The PSD visualization illuminates why channel estimation proves so critical for OFDM systems. Subcarriers experiencing deep spectral nulls effectively disappear beneath the noise floor, rendering their data essentially unrecoverable without accurate channel knowledge. A receiver attempting blind detection at these frequencies would face impossible SNR conditions. Channel estimation enables the system to identify these problematic subcarriers and either avoid them entirely through adaptive subcarrier allocation or compensate for the attenuation through aggressive equalization boost.

Comparing the transmitted and received PSDs also reveals interesting characteristics of the measurement environment. The received spectrum shows relatively slow variation with frequency spectral features span multiple subcarriers rather than fluctuating rapidly from one subcarrier to the next. This smooth variation indicates moderate delay spread in the channel, consistent with the ITU Pedestrian A model used in the simulations. Had the measurements been collected in a highly dispersive environment (indoor, dense urban canyon), more rapid spectral fluctuations would be observed as closely-spaced frequencies experience independent fading.

The PSD analysis also helps explain why the hybrid ResNet-based with LSTM outperforms simpler approaches. The spectral correlation structure visible in Figure 13 is the nearby subcarriers exhibiting similar power levels, represents exactly the type of spatial pattern that convolution layers learn to exploit. By processing the entire frequency grid simultaneously, the CNN can infer the likely channel response at a spectral null based on observations from neighboring frequencies. The LSTM extends this capability into the time domain, tracking how spectral features evolve across successive OFDM symbols as the mobile user moves through the propagation environment.

From a practical deployment perspective, PSD measurements like those in Figure 13 serve important diagnostic functions. Network planning engineers use such plots to identify frequency bands experiencing persistent interference or anomalous propagation, informing decisions about channel allocation and antenna placement. The relatively clean spectral shape in the measurements strong signal power with no spurious emissions or interference validates the measurement environment's suitability for evaluating channel estimation algorithms without confounding factors.

G. Discussion and Practical Implications

The experimental results presented in this section paint a compelling picture of deep learning's transformative potential for wireless channel estimation. Across every metric examined MSE, error distribution, constellation quality, spectral characteristics the learning-based approaches demonstrate clear and substantial superiority over conventional techniques. These gains prove consistent across SNR levels and robust to the complexities of real-world propagation, validating those benefits observed in controlled simulations transfer to operational deployments.

Several practical insights emerge from the detailed analysis. First, the hybrid ResNet-with LSTM architecture justifies its added complexity through consistent performance leadership across all test conditions. While the standalone CNN offers substantial improvement over LS with modest computational overhead, the hybrid model's superior handling of temporal correlation provides additional gains that could prove decisive for demanding applications. Second, the results reveal that the value proposition for deep learning estimation varies with operating conditions, proving strongest in the 10-20 dB SNR range where commercial systems typically operate.

Third, and perhaps most importantly, the real-world validation using MTN Nigeria's network demonstrates that these techniques can handle the messiness of actual wireless channels. The training data, generated from idealized ITU channel models, nonetheless enabled the networks to generalize effectively to operational conditions involving unpredictable interference, hardware imperfections, and propagation phenomena not explicitly modeled. This generalization capability, the hallmark of successful machine learning, suggests that once-trained networks could deploy broadly across diverse geographical regions and network configurations without requiring site-specific retraining.

Looking forward, these results open exciting avenues for future investigation. The techniques could extend naturally to more advanced modulation schemes (64-QAM, 256-QAM) where channel estimation accuracy becomes even more critical. Massive MIMO systems, with their numerous spatial dimensions, represent another promising application domain where deep learning's pattern recognition capabilities might unlock performance gains unachievable through conventional methods. The path toward intelligent, adaptive wireless physical layers appears increasingly viable as computational capabilities continue advancing while energy efficiency improves.

VIII. CONCLUSION AND FUTURE WORK

A. Conclusion

The proposed ResNet-based CNN and especially the hybrid ResNet-based with LSTM architecture deliver substantial MSE and system-level improvements over traditional methods validated through both simulation and real MTN Nigeria LTE measurements.

This paper has presented a comprehensive investigation of deep learning-based channel estimation for OFDM systems, with detailed analysis of ResNet-based CNN and hybrid ResNet-based with LSTM architectures. Through extensive simulation and rigorous evaluation. The results presented in this paper have demonstrated that data-driven approaches offer substantial performance improvements over conventional estimation methods.

The standalone CNN estimator exploits spatial correlations in pilot arrangements through hierarchical feature extraction, achieving 3.99 dB NMSE improvements over LS estimation across diverse SNR ranges. Incorporating batch normalization and dropout regularization ensures training stability and generalization. The proposed hybrid ResNet-based with LSTM architecture extends these benefits by explicitly modeling temporal dependencies across OFDM symbols. Bidirectional LSTM layers with sophisticated gating mechanisms enable robust tracking of time-varying channels, demonstrating particular advantages in high-mobility scenarios with significant Doppler spread.

At the target operating point of 20 dB SNR, the hybrid model achieves -9.22 dB NMSE, representing 3.99 dB improvement over LS, 2.3 dB over MMSE, and 1.8 dB over the standalone CNN. These gains translate directly to improved bit error rates and system throughput. Remarkably, the deep learning approaches outperform MMSE without requiring explicit channel statistics, validating the power of automated feature learning from data.

Computational complexity analysis reveals that while deep learning methods demand significantly more operations than LS, modern GPU hardware can achieve real-time processing with acceptable latency. Model compression through quantization reduces requirements by 60% with minimal performance sacrifice, and dedicated neural network accelerators promise further improvements. The demonstrated generalization across channel models and operating conditions, particularly when augmented with limited fine-tuning, suggests practical deployment feasibility.

Several promising directions emerge for future research. First, transformer-based architectures with self-attention mechanisms may capture longer-range dependencies more effectively than LSTMs, though at increased computational cost. Investigating efficient attention mechanisms tailored for channel estimation represents an interesting avenue. Second, incorporating domain knowledge through physics-informed neural networks could improve sample efficiency and generalization. For example, enforcing sparsity in the delay domain or exploiting known pilot patterns might reduce training data requirements.

Third, online learning and continual adaptation mechanisms would enable deployed systems to refine their models based on actual operating conditions without requiring offline retraining. Meta-learning approaches might facilitate rapid adaptation to new channel environments with minimal data. Fourth, extension to massive MIMO and mmWave systems presents opportunities to exploit both spatial and frequency diversity at massive scale. The computational challenges are substantial but addressable through distributed processing and specialized hardware.

Finally, joint optimization of pilot patterns, transmission strategies, and neural network architectures may unlock additional performance gains. Rather than treating channel estimation as an isolated problem, end-to-end learning from transmitted bits to detected symbols could optimize the entire communication chain. Such approaches require careful consideration of differentiability and training stability but promise revolutionary improvements.

In conclusion, this work establishes deep learning, particularly hybrid ResNet-based with LSTM architectures, as a superior alternative to conventional channel estimation methods for OFDM systems. The demonstrated performance improvements, combined with practical feasibility considerations, position these approaches for adoption in next-generation wireless communication standards. As wireless systems evolve toward higher frequencies, larger bandwidths, and more complex propagation environments, intelligent, data-driven techniques will become increasingly indispensable.

B. Future Directions

Several promising directions emerge from this investigation. First, transformer-based architectures incorporating multi-head self-attention mechanisms represent a natural evolution beyond LSTM for capturing long-range temporal dependencies in OFDM channel sequences, potentially offering superior performance at the cost of increased computational overhead. Efficient attention variants tailored to the structured time-frequency grid of OFDM merit dedicated investigation. Second, physics-informed neural networks that enforce domain-specific constraints such as delay-domain sparsity, known pilot patterns, or channel coherence bandwidth bounds could substantially improve sample efficiency and generalization to out-of-distribution channel conditions encountered in novel deployment environments.

Third, online learning and meta-learning frameworks would enable continuously deployed models to refine their estimation strategies based on observed channel statistics without requiring periodic offline retraining a critical capability for adaptive LTE systems operating in dynamically evolving propagation environments. Fourth, direct extension to massive MIMO and millimetre-wave (mmWave) systems represents a high-impact application domain where the spatial and temporal correlation structures are richer and more structured, potentially amplifying the advantages of deep learning estimation. Fifth, and most specifically relevant to this study's O+F OFDM focus, future work should investigate fine-tuning and real-time FPGA deployment of the hybrid ResNet-based CNN with LSTM model alongside full O+F OFDM signal processing chains, and explore end-to-end learning frameworks that jointly optimize pilot placement, optical filtering stages, and channel estimation in a unified differentiable architecture [14, 64 – 68].

REFERENCES

[1]. F. Ahmed-Ade, V. A. Akpan, and E. O. Ogolo, "OFDM variants and FPGA implementation: A comprehensive analysis of algorithm comparison, optical wireless integration, neural network mapping, and FPGA realization," *American Journal of Embedded Systems and Applications*, vol. 11, no. 1, pp. 16–38, 2025, doi: 10.11648/j.ajesa.20251101.13.

[2]. F. Ahmed-Ade, V. A. Akpan, and E. O. Ogolo, "Development of Integrated Optical Plus Filtered OFDM Algorithms for Optimal Telecommunication Systems Design," *International Journal of Advances in Engineering and Management*, vol. 7, no. 12, pp. 214–243, 2025, doi: 10.35629/5252-0712214243.

[3]. F. Ahmed-Ade, V. A. Akpan, and E. O. Ogolo, "Real-Time Implementation of Integrated Optical Plus Filtered OFDM 5G Network Parameters for LTE and DVB-T2 Telecommunication Systems," *Scientific Journal of Engineering Research*, 2026. [Online].Available: https://journal.futuristech.co.id/index.php/sjer. doi: 10.64539/sjer.v2i1.2026.372.

[4]. V. A. Akpan, E. C. Njoku and E. I. Obi, "Slice-Specific Learning Models for Intrusion Detection in 5G Telecommunication Networks", *International Journal of Wireless Communication and Mobile Computing*, vol. 12, no. 2, pp. 93 – 118, 2025. Available [Online]: https://www.sciencepublishinggroup.com/article/10.11648/j.wcmc.20251202.14. doi: https://doi.org/10.11648/j.wcmc.20251202.14.

[5]. O. G. Ajileye, J. S. Ojo and V. A. Akpan, "Comparative Analysis of Machine Learning Algorithms for Good Quality of Service on Terrestrial and Satellite Network Systems Over Nigeria", *International Journal of Networks and Communications*, vol. 15, no. 1, pp. 1 – 12, 2026. Available [Online]: http://article.sapub.org/10.5923.j.ijnc.20261501.01.html. doi: doi:10.5923/j.ijnc.20261501.01.

[6]. S. Khnichar, W. Santipach, L. Wuttisittikulkij, A. Parnianfard and S. Chaudhary, "Efficient Channel Estimation in OFDM Systems Using a Fast Super-Resolution CNN Model," *Journal of Sensor and Actuator Networks*, vol. 13, no. 55, pp. 1 – 17, 2024. doi: 10.3390/jsan13050055.

[7]. I. Helmy, P. Tarafder and W. Choi, "LSTM-GRU Model-Based Channel Prediction for One-Bit Massive MIMO System," IEEE Transactions on Vehicular Technology, vol. 72, no. 8pp. 11053 – 11057, 2023. doi: 10.1109/TVT.2023.3262951.

[8]. S. J. Olickal and R. Jose, "LSTM projected layer neural network-based signal estimation and channel state estimator for OFDM wireless communication systems," *AIMS Electronics and Electrical Engineering*, vol. 7, no. 2, pp. 187 – 195, 2023. doi: 10.3934/electreng.2023011.

[9]. B. M. Manasa, V. Pakala, R. Chinthaginjala, M. Manel Ayadi, M. Hamdi and A. Ksibi, "A Novel Channel Estimation Framework in MIMO Using Serial Cascaded Multiscale Autoencoder and Attention LSTM with Hybrid Heuristic Algorithm," *Sensors*, vol. 23, no. 9154, pp. 1 – 27, 2023. doi: https://doi.org/10.3390/s23229154.

[10]. B. D. Filippo, C. Amatetti and A. Vanelli-Coralli, "Uplink OFDM Channel Prediction with Hybrid CNN-LSTM for 6G Non-Terrestrial Networks," *arXiv*, 2025. Available [Online]: https://arxiv.org/html/2502.09326v1.

[11]. M. Adil, S. Liu, S. Mazhar, A. Alharbi, H. Yan and M. Muzzammil, "A Novel Transfer Learning-Based OFDM Receiver Design for Enhanced Underwater Acoustic Communication," *Journal of Marine Science and Engineering*, vol. 13, no. 1284, pp. 1 – 36, 2025. doi: https://doi.org/10.3390/jmse13071284.

[12]. H. A. Hassan, M. A. Mohamed, M. H. Essai, A. S. Mubarak, H. Esmaiel and O. A. Omer, "An efficient and reliable OFDM channel state estimator using deep learning convolutional neural networks," *Journal of Engineering Sciences – Part B: Electrical Engineering*, vol. 51, no. 6, pp. 32 – 49, 2023. doi: 10.21608/JESAUN.2023.215113.1236.

[13]. A. S. M. Mohammed, A. I. A. Taman, A. M. Hassan and A. Zekry, "Deep Learning Channel Estimation for OFDM 5G Systems with Different Channel Models," *Wireless Personal Communications*, vol. 128, pp. 2891 – 2912, 2022. doi: https://link.springer.com/content/pdf/10.1007/s11277-022-10077-6.pdf.

[14]. V. A. Akpan, "Hard and soft embedded FPGA processor systems design: Design considerations and performance comparisons," *International Journal of Engineering and Technology*, vol. 3, no. 11, pp. 1000-1020, Nov. 2013. Available [Online]: http://www.iet-journals.org/archive/2013/november_vol_3_no_11/7153671377348526.pdf.

[15]. Y. S. Cho, J. Kim, W. Y. Yang, and C. G. Kang, "MIMO-OFDM Wireless Communications with MATLAB," John Wiley & Sons, 2010. doi: 10.1002/9780470825631.

[16]. M. Morelli and U. Mengali, "A comparison of pilot-aided channel estimation methods for OFDM systems," *IEEE Transactions Signal Processing*, vol. 49, no. 12, pp. 3065-3073, Dec. 2001. doi: 10.1109/78.969507.

[17]. S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Transactions Broadcasting*, vol. 48, no. 3, pp. 223-229, Sep. 2002. doi: 10.1109/TBC.2002.804034.

[18]. O. Edfors, M. Sandell, J. J. van de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," *IEEE Transactions Communications*, vol. 46, no. 7, pp. 931-939, Jul. 1998. doi: 10.1109/26.701321.

[19]. J. J. van de Beek, O. Edfors, M. Sandell, S. K. Wilson, and P. O. Borjesson, "On channel estimation in OFDM systems," *In the Proceedings of the IEEE VTC*, Chicago, IL, USA, Jul. 1995, pp. 815-819. doi: 10.1109/VETEC.1995.504981.

[20]. M. Hsieh and C. Wei, "Channel estimation for OFDM systems based on comb-type pilot arrangement in frequency selective fading channels," *IEEE Transactions Consumption Electronics*, vol. 44, no. 1, pp. 217-225, Feb. 1998. doi: 10.1109/30.663750.

[21]. Y. Li, L. J. Cimini, and N. R. Sollenberger, "Robust channel estimation for OFDM systems with rapid dispersive fading channels," *IEEE Transactions Communications*, vol. 46, no. 7, pp. 902-915, Jul. 1998. doi: 10.1109/26.701317.

[22]. P. Hoeher, S. Kaiser, and P. Robertson, "Two-dimensional pilot-symbol-aided channel estimation by Wiener filtering," *In the Proceedings of the IEEE International Conference Acoustic, Speech, Signal Process (ICASSP)*, Munich, Germany, Apr. 1997, pp. 1845-1848. doi: 10.1109/ICASSP.1997.598897.

[23]. C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2224-2287, 3rd Quarter 2019. doi: 10.1109/COMST.2019.2904897.

[24]. T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications Networking*, vol. 3, no. 4, pp. 563-575, Dec. 2017. doi: 10.1109/TCCN.2017.2758370.

[25]. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, May 2015. doi: 10.1038/nature14539.

[26]. C. Wen, W. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748-751, Oct. 2018. doi: 10.1109/LWC.2018.2818160.

[27]. 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, no. 5, pp. 852-855, Oct. 2018. doi: 10.1109/LWC.2018.2832128.

[28]. C. Qing, L. Dong, L. Wang, G. Ling and J. Wang, "Transfer learning-based channel estimation in orthogonal frequency division multiplexing systems using data-nulling superimposed pilots," *PLOS One*, vol. 17, no. 5, pp. 1 – 20, 2022. doi: https://doi.org/10.1371/journal.pone.0268952.

[29]. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computations*, vol. 9, no. 8, pp. 1735-1780, Nov. 1997. doi: 10.1162/neco.1997.9.8.1735.

[30]. K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey*," IEEE Transactions Neural Networking Learning Systems*, vol. 28, no. 10, pp. 2222-2232, Oct. 2017. doi: 10.1109/TNNLS.2016.2582924.

[31]. G. J. Gibson, S. Siu, and C. F. N. Cowan, "Multilayer perceptron structures applied to adaptive equalizers for data communications," *In the Proceedings of the IEEE International Conference Acoustic, Speech, Signal Process, ICASSP*, Apr. 1989, pp. 1183-1186. doi: 10.1109/ICASSP.1989.266563.

[32]. L. Zhu, C. Bockelmann, T. Schier, S.E. Hajri and A. Dekorsy, "Enhancing CNN-Based Channel Estimation Using Transfer Learning in OFDM Systems," *In the Proceedings of the 2023 IEEE 24th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Shanghai, China, 25 – 28 September, 2023,, pp. 21 – 25, 2023. doi: 10.1109/SPAWC53906.2023.10304560.

[33]. S. L. Obispo, "*OFDM Channel Estimation with Artificial Neural Networks*," M. Sc. Thesis in Electrical Engineering, California Polytechnic State University, U.S.A. pp. 1 – 141, 2022. Available [Online]: https://digitalcommons.calpoly.edu/cgi/viewcontent.cgi?article=4079&context=theses.

[34]. Y. Yang, W. Chao, L. Ma, F. Meng and Z. Hu, "A Novel Autoencoder-Based Design for Channel Estimation in Maritime OFDM Systems," *Electronics*, vol. 14, no. 3454, pp. 1 – 15, 2025. doi: https://doi.org/10.3390/electronics14173454.

[35]. N. Ginige, N. Rajatheva and M. Latva-Aho, "A CNN-Based End-to-End Learning for RIS-Assisted Communication Systems," *arXiv*, pp. 1 – 6, 2025. Available [Online]: https://arxiv.org/pdf/2503.13976v1.

[36]. H. Ye, G. Y. Li, and B. 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, Feb. 2018. doi: 10.1109/LWC.2017.2757490.

[37]. M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 652-655, Apr. 2019. doi: 10.1109/LCOMM.2019.2898944.

[38]. A. M. Elbir, "CNN-based precoder and combiner design in mmWave MIMO systems," *IEEE Communications Letters*, vol. 23, no. 7, pp. 1240-1243, Jul. 2019. doi: 10.1109/LCOMM.2019.2915977.

[39]. Y. Jin, J. Zhang, X. Zhang, and D. Yuan, "Channel estimation for cell-free mmWave massive MIMO through deep learning," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10325-10329, Oct. 2019. doi: 10.1109/TVT.2019.2937543.

[40]. P. Wang, J. Xiao, and L. P. Li, "Comparison of orthogonal and non-orthogonal approaches to future wireless cellular systems," *IEEE Vehicular Technology Magazine*, vol. 1, no. 3, pp. 4-11, Sep. 2006. doi: 10.1109/MVT.2006.307294.

[41]. X. Liu, S. Zhao, A. Liu, N. Zhao, and W. Chen, "Intelligent reflecting surface aided MIMO broadcasting for simultaneous wireless information and power transfer," *IEEE Journal on Selected Areas Communications*, vol. 38, no. 8, pp. 1719-1734, Aug. 2020. doi: 10.1109/JSAC.2020.3000802.

[42]. J. Guo, C. Wen, S. Jin, and G. Y. Li, "Convolutional neural network-based multiple-rate compressive sensing for massive MIMO CSI feedback," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2827-2840, Apr. 2020. doi: 10.1109/TWC.2020.2968430.

[43]. A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti and D. De, "Fundamental Concepts of Convolutional Neural Network," In: Balas, V., Kumar, R., Srivastava, R. (eds) Recent Trends and Advances in Artificial Intelligence and Internet of Things. *Intelligent Systems Reference Library*, Vol. 172, pp. 516 – 567, 2020. Springer, Cham., Available: https://doi.org/10.1007/978-3-030-32644-9_36.

[44]. [44] M. Krichen, "Convolutional Neural Networks: A Survey," Computers, Vol. 12, No. 151, pp. 1 – 41, 2023. Available: https://doi.org/10.3390/computers12080151.

[45]. I. D. Mienye, T. G. Swart, G. Obaido, M. Jordan and P. Ilono, "Deep Convolutional Neural Networks: A Comprehensive Review," Technical Report, Institute of Intelligent Systems, University of Johannesburg, Johannesburg, Gauteng, South Africa, pp. 1 – 34, 2024. Available: doi: 10.20944/preprints202408.1288.v1.

[46]. R. Khanam, M. Hussain, R. Hill and P. Allen, "A Comprehensive Review of Convolutional Neural Networks for Defect Detection in Industrial Applications," IEEE Access, Vol. 4, pp. 1 – 47, 2016. Available: doi: 10.1109/ACCESS.2024.3425166.

[47]. O. Iparraguirre-Villanueva, V. Guevara-Ponce, O. R. Paredes, F. A. Sierra- Liñan , J. E. Zapata-Paulini and M. A. Cabanillas-Carbonell, "Convolutional Neural Networks with Transfer Learning for Pneumonia Detection," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 9, pp. 544 – 551, 2022. doi: 10.14569/IJACSA.2022.0130963.

[48]. D. A. Pustokhin, I. V. Pustokhina, P. N. Dinh, S. V. Phan, G. N. Nguyen, G. P. Joshi and K. Shankar, "An effective deep residual network based class attention layer with bidirectional LSTM for diagnosis and classification of COVID-19," Journal of Applied Statistics, vol. 50, no. 3, pp. 1 – 18, 2020. doi: 10.1080/02664763.2020.1849057.

[49]. P. Shourie, V. Anand and S. Gupta, "Bridging Nature and Technology: ResNet-152 for Scene Visualization and Classification," In the Proceedings of the *2024 4th International Conference on Sustainable Expert Systems (ICSES)*, Kaski, Nepal, 15 – 17 October, 2024, pp. 1512 – 1516, 2024. doi: 10.1109/ICSES63445.2024.10763294.

[50]. O. Calzone, "An Intuitive Explanation of LSTM," *Medium*. Last updated 21st February, 2022. Retrieved on 8th January, 2026. Available [Online]: https://medium.com/@ottaviocalzone/an-intuitive-explanation-of-lstm-a035eb6ab42c.

[51]. Gourav, "Understanding Architecture of LSTM," Analytics Vidhya. Last Updated 2nd May, 2025. Retrieved on 8th January, 2026. Available [Online]: https://www.analyticsvidhya.com/blog/2021/01/understanding-architecture-of-lstm/.

[52]. M. Khan and Y. Hossni, "A comparative analysis of LSTM models aided with attention and squeeze and excitation blocks for activity recognition," *Nature Scientific Report*, vol. 15, no. 1, pp. 1 – 21, 2025. doi: 10.1038/s41598-025-88378-6.

[53]. G. Tuteja, F. A. M. Al-Yarima, A. Ikram, R. Gupta, A. U. Rehman, J. Singh, I. D. Noya and L. A. D. Lopez, "CNNAttLSTM: An Attention-Enhanced CNN–LSTM Architecture for High-Precision Jackfruit Leaf Disease Classification*," Frontiers in Plant Science*, vol. 16, no. 1720471, pp. 1 – 21, 2026. doi: 10.3389/fpls.2025.1720471.

[54]. A. Narula and N. K. Vaegae, "Development of CNN-LSTM Combinational Architecture for COVID-19 Detection," Journal of Ambient Intelligent and Hunamized Computing, vol. 14, pp. 2645 – 2656, 2023. doi: https://doi.org/10.1007/s12652-022-04508-2.

[55]. N. A. Sameer and B. M. Nema, "LSTM and CNN hybrid model for enhanced fingerprint recognition," Emerging Trends in Drugs, Addictions, and Health, vol. 5, no. 100174, pp. 1 – 7, 2025. doi: https://doi.org/10.1016/j.etdah.2025.100174.

[56]. K. O. Adefemi and M. B. Mutanga, "A Robust Hybrid CNN–LSTM Model for Predicting Student Academic Performance," *Digital*, vol. 5, no. 16, pp. 1 – 15, 2025. doi: https://doi.org/10.3390/digital5020016.

[57]. [57] R. S. El-Sayed, "A Hybrid CNN-LSTM Deep Learning Model for Classification of the Parkinson Disease," *IAENG International Journal of Applied Mathematics*, vol. 53, no. 4, pp. 1 – 10, 2023. Available [Online]: https://www.iaeng.org/IJAM/issues_v53/issue_4/IJAM_53_4_29.pdf.

[58]. M. B. Suresh, N. Dinesh and T. N. Abhishek, "Hybrid Resnet CNN-LSTM for Deepfake Video Detection," *International Journal of Creative Research Thoughts*, vol. 12, no. 4, pp. 558 – 596, 2024. Available [Online]: https://www.ijcrt.org/papers/IJCRT24A4909.pdf.

[59]. Q. Xu, Y. Li, C. Zhang and Z. Kou, "A Hybrid ResNet-LSTM Model for Real-time Welding Defect Prediction and Optimization," *In the Proceedings of the 2025 5th Internal Conference on Internet of Things and Machine Learning*, Nanchang, China, 16 – 18 May, 2025, pp. 34 – 38, 2025. doi: https://doi.org/10.1145/3749566.3749574.

[60]. Kingma, Diederik P., and Jimmy B. "Adam: A Method for Stochastic Optimization." *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, California, USA, May 7–9, 2015. Available: https://arxiv.org/abs/1412.6980.

[61]. W. C. Jakes, "Microwave Mobile Communications," Wiley-IEEE Press, 1994. doi: 10.1109/9780470545287.

[62]. V. A. Akpan and N. O. Raji, "Development of Deep Convolutional Neural Network Algorithms Based on DenseNet Model for Adaptive Classification of Chest Diseases", *International Journal of Biophotonics and Biomedical Engineering*, vol. 5, no. 2, Article 9, pp. 1 – 28, 2025Available [Online]: https://oiccpress.com/ijbbe/article/view/16834/20062. DOI: https://doi.org/10.57647/ijbbe.2025.0502.09.

[63]. ITU-R, Radiocommunication Sector. (2009, December). Guidelines for evaluation of radio interface technologies for IMT-Advanced. Report ITU-R M.2135-1. Geneva, Switzerland: ITU. Available: https://www.itu.int/rec/R-REC-M.2135-1-200912-I/en.

[64]. V. A. Akpan, "Model-based embedded-processor systems design methodologies: Modeling, syntheses, implementation and validation," *African Journal of Computing and ICTs*, 5(1): 1 – 26, 2012. Available [Online]: http://www.ajocict.net/uploads/V5N1P1_-_2012_-AJOCICT-_Model-Based_FPGA_.pdf.

[65]. V. A. Akpan, "An FPGA realization of integrated embedded multi-processors system: A hardware-software co-design approach," *Journal of Advanced Research in Embedded System (JoARES)*, 2(1): 1 – 27, 2015. Available [Online]: http://technology.adrpublications.com/index.php/JoARES/article/view/166/203.

[66]. V. A. Akpan, "FPGA-in-the-Loop implementation of an adaptive matrix inversion algorithmic co-processor: A dual-processor system design," *Journal of Advanced Research in Embedded System (JoARES)*, 2(1): 28 – 63, 2015. Available [Online]: http://technology.adrpublications.com/index.php/JoARES/article/view/158/204.

[67]. V. A. Akpan, D. Chasapis and G. D. Hassapis, "FPGA Implementation of Neural Network-Based AGPC for Nonlinear F-16 Aircraft Auto-Pilot Control: Part 1 – Modeling, Synthesis, Verification and FPGA-in-the-Loop Co-Sim," *American Journal of Embedded Systems and Applications*, vol. 9, no. 1, pp. 6 – 36, 2022. Available [Online]: https://www.sciencepg.com/journal/paperinfo?journalid=236&doi=10.11648/j.ajesa.20220901.13.

[68]. V. A. Akpan, D. Chasapis and G. D. Hassapis "FPGA Implementation of Neural Network-Based AGPC for Nonlinear F-16 Aircraft Auto-Pilot Control: Part 2 – Implementation of Embedded PowerPC™440 with AGPC," *American Journal of Embedded Systems and Applications*, vol. 9, no. 2, pp. 37 – 65, 2022. Available [Online]: https://www.sciencepublishinggroup.com/journal/paperinfo?journalid=236&doi=10.11648/j.ajesa.20220902.11.

FUNDING

This work is supported by Institute Based Research (IBR) from the Tertiary Education Trust Fund (TETFund), Federal Government of Nigeria.