electronics

MDPI

Article

AI-Enabled Autoencoder-Based Physical Layer Design for 6G Communication Systems

Andreani Christopoulou [1,2,*], Dimitrios Kosmanos [1], Apostolos Xenakis [1] and Costas Chaikalis [1,*]

[1] Department of Digital Systems, University of Thessaly, 41500 Larissa, Greece; dikosman@uth.gr (D.K.); axenakis@uth.gr (A.X.)
[2] Department of Informatics and Telecommunications, University of Thessaly, 35100 Lamia, Greece
* Correspondence: anchristopoulou@uth.gr (A.C.); kchaikalis@uth.gr (C.C.)

Abstract

Next-generation wireless communication 6G systems are expected to operate under diverse channel conditions and structures, requiring flexible and data-driven communication schemes. As traditional techniques face limitations in complex and dynamic environments, trained communication architectures have emerged as promising alternatives. In this paper, we present a thorough study on deep learning trained physical layer components, focusing on autoencoder-based transceivers and neural network modules that enhance the receiver's intelligence. We further investigate two essential deep learning capabilities for modern receivers—modulation classification using neural architectures and generative data synthesis for channel estimation training. Moreover, the proposed models and simulation framework provide insight into how deep learning can be systematically integrated into the physical layer to improve adaptability, robustness, and efficiency.

Keywords: AI; autoencoders; 6G; physical layer; modulation; classification; neural network

1. Introduction

The transition toward the sixth generation (6G) of wireless communications is targeting ultra-low latency, terabit-per-second throughput, and global coverage. Unlike its predecessors, 6G is expected to operate across diverse spectrum bands, extending from sub-6 GHz to the millimeter-wave (mmWave) and Terahertz (THz) frequencies. While these high-frequency bands offer vast bandwidth, they introduce significant challenges: severe path loss, high susceptibility to atmospheric conditions, and hardware non-linearities that render traditional analytic signal processing models increasingly inaccurate. Recent 6G studies have identified the upper mid-band (7–24 GHz), also referred to as FR3, as a "golden band" that offers a favorable trade-off between coverage and capacity, with propagation characteristics superior to mmWave while enabling significantly wider bandwidth than sub-6 GHz, making it a strong candidate for early 6G deployments [1]. However, operation in the FR3 band introduces frequency-dependent propagation effects, hardware impairments, and non-ideal channel behaviors that are difficult to capture with conventional analytic models, thereby motivating learning-based physical layer designs that can adapt directly from data [1].

The continuous evolution of modern communication systems has accelerated and sparked the interest in a new form of physical layer design, using machine learning, motivated by the limitations of traditional signal processing in complex and dynamic environments. Among the most active areas of interest is the use of deep neural networks

Check for updates

Academic Editor: Yung-Fa Huang

Received: 30 December 2025
Revised: 22 January 2026
Accepted: 23 January 2026
Published: 26 January 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

in order to reshape the physical layer's core components, like channel coding and estimation, modulation, etc. As these learned and trained systems evolve, understanding how design parameters influence their behavior and performance becomes crucial. For example, choosing the normalization strategy of an autoencoder directly influences its constellation geometry and the reliability of the transmitted signals.

The ability to generate reliable training data and recognize diverse modulation formats is critical in such systems. Deep learning has demonstrated significant gains regarding modulation classification and channel estimation in various channel conditions, enabling adaptivity and providing an alternative to conventional channel modeling. This shift paves the way for AI-supported designs in next-generation telecommunications.

Motivated by this evolution in communication systems, autoencoders have emerged as promising end-to-end transceiver solutions, optimizing the encoding, modulation, and decoding without relying on the strict architecture of the existing analytic models. Prior works [2,3] have shown that learned constellations deviate from traditional QAM architecture depending on the power constraints that are imposed during the training process. Such differences affect the Bit Error Rate (BER) and provide insights into how the new neural models map the symbols in each occasion. Deep learning has also been proven effective in modulation classification and channel estimation, enabling receivers to infer signal characteristics and states, using features that are directly learned from data, rather than existing limited models.

In this work, we present a thorough study examining normalization effects, communication schemes, and deep-learning modules that are relevant to the physical layer's architecture and structure. This study addresses the limitations of traditional signal processing in complex 6G environments by providing a holistic evaluation of deep learning (DL)-based physical layer components. Moving beyond the isolated module analysis typical of prior works, we provide a unified assessment of the following key contributions:

- Analysis of Normalization Trade-Offs: we provide a direct comparative analysis between energy and average power normalization, revealing that while power normalization enables adaptive constellation shaping for high-SNR gains, it introduces structural vulnerabilities at low SNRs due to irregular constellation points and implicitly increases the Peak-to-Average Power Ratio (PAPR) (Sections 4.1 and 4.2).
- Validation of Near-Optimal Learned Coding: we demonstrate that (7,4) autoencoder-based coding schemes discover mapping and decision regions that statistically align with the theoretical performance limits of Maximum Likelihood Decoding (MLD), effectively bridging the gap between hard-decision decoding and optimal theory without explicit algebraic structures (Section 4.3).
- Design of Robust Auxiliary Receiver Modules: we implement and validate a 28-layer CNN for high-accuracy modulation classification (exceeding 95%) and a generative framework for channel estimation that consistently outperforms standard linear interpolation and approaches industry-standard practical estimators (Sections 4.4 and 4.5).
- Integrated System-Level Evaluation: we establish a unified framework that jointly analyzes coding, classification, and estimation under consistent channel assumptions, capturing how design choices—such as normalization—propagate through the entire 6G processing chain to influence end-to-end performance (throughout Section 4 and highlighted in Table 1).
- Computational Feasibility Assessment: unlike high-complexity DL models that require specialized hardware, we demonstrate that these AI-enabled components are computationally efficient enough to be successfully trained and deployed on standard CPU-based systems, proving their accessibility for practical 6G edge devices (throughout Section 4).

Table 1. Comparison of contributions between existing works and this study.

Aspect	Existing Works	Our Work/Contribution
Autoencoder Normalization Schemes	Explored energy or average power normalization separately	Direct comparison of energy vs. average power normalization and their impact on constellation geometry and block error rate (BER) variability
Autoencoder vs. Classical Code Performance	Autoencoders evaluated individually	Compare learned autoencoder coding performance directly to classical coding methods under consistent channel conditions
Modulation Classification at Different SNRs	SNR-dependent classification shown in isolation	Detailed modulation classification across multiple SNRs with confusion analysis, highlighting interactions with autoencoder output
Neural Network-Based Channel Estimation	CNN or other neural network estimators tested for robustness	Evaluate NN-based channel estimation against interpolation and practical estimators under autoencoder-generated signals
Integrated System-Level Evaluation	Most studies evaluate modules independently	Joint evaluation of autoencoder, modulation classifier, and channel estimator to capture end-to-end system-level performance

2. Related Work

Deep learning has emerged as a promising approach for physical layer communication systems, particularly through autoencoder architectures that jointly optimize encoding and decoding. Several studies have shown that such autoencoders can learn efficient communication schemes directly from data, demonstrating feasibility even in continuous over-the-air transmission scenarios, and often approaching the performance of classical coding and decoding methods without relying on explicit algebraic code constructions [3–6].

Normalization strategies, such as energy constraints and average power constraints, play a crucial role in shaping the learned constellations and influencing decoding performance. Previous research has examined power efficient communication system designs using neural networks, highlighting the benefits and trade-offs of different normalization approaches, including adaptive power allocation strategies that can improve performance at the cost of stability and generalization [7,8].

Convolutional neural networks (CNNs) have been widely employed to differentiate between various modulation schemes. These works consistently demonstrate that classification accuracy strongly depends on the signal-to-noise ratio (SNR), with performance degrading at low SNRs due to constellation overlap and noise. Confusions among modulations with similar spectral or structural properties have also been reported [9–11].

Neural network-based channel estimation has been proposed as an alternative to classical estimators in OFDM and MIMO systems. Such approaches provide robust estimates under diverse channel conditions and varying SNRs, typically outperforming simple interpolation techniques but sometimes falling short of practical estimators optimized for specific scenarios [12–14].

Recent studies have also highlighted the use of hybrid and residual deep learning models for robust modulation classification. For example, Convolutional Long Short-Term Deep Neural Networks (CLDNNs) have been shown to outperform standard CNNs and ResNets by effectively capturing complementary spatial and temporal features inherent

in radio signals. Furthermore, Deep Residual Networks have also seen significant improvement in sensitivity over the traditional featured-based baselines. These findings emphasize that accounting for specific channel effects like carrier frequency offset is often critical in OTA environments [15,16].

As communication networks evolve toward 6G, deep autoencoders are being deployed not only for signal reconstruction but also as unsupervised anomaly detectors capable of identifying novel intrusion patterns by learning latent traffic representations. Comprehensive reviews of autoencoder applications further underscore the necessity of addressing challenges such as non-differentiable channel components and computational complexity to realize practical, deployment-ready systems across wireless and optical domains. These advancements suggest that the next generation of physical layer deep learning must simultaneously optimize for spectral efficiency, channel robustness, and security [17,18].

This work extends existing research by providing a direct comparison of energy normalization and average power normalization in autoencoder-based communication systems, examining their impact on constellation geometry and block error rate variability. Additionally, it presents detailed modulation classification results across multiple SNR levels with confusion analysis and evaluates CNN-based channel estimation performance against both practical and interpolation methods over random channel realizations. The integrated evaluation across encoding, modulation classification, and channel estimation components offers a comprehensive view of the strengths and trade-offs of deep learning in end-to-end communication system design.

A key contribution of this work is the integrated system-level evaluation of deep learning-based physical layer components, in contrast to prior studies that typically examine each module such as autoencoders, modulation classifiers, or channel estimators separately. Our evaluation framework jointly analyzes coding performance, constellation shaping, modulation classification, and channel estimation under consistent channel conditions and across multiple SNR levels. By doing so, we capture how improvements or limitations in one component propagate through the entire communication chain. For example, the choice of normalization strategy in the autoencoder influences constellation geometry, which then affects robustness at low SNRs. Likewise, the channel estimation experiments reveal how learned estimators interact with the received signal characteristics produced by the autoencoder-based system. This assessment provides a more realistic understanding of the strengths and trade-offs of deep learning when deployed as part of a full physical layer pipeline, rather than as isolated algorithms. Overall, the integrated evaluation demonstrates that deep learning components can collectively approach classical performance bounds while offering adaptability that is difficult to achieve with traditional model-based designs.

Table 1 summarizes the key contributions of this study in comparison with existing works, emphasizing the integrated evaluation of deep learning components in physical layer communication systems.

3. System Model

We consider the communication system model depicted in Figure 1, which is composed of a stream-based processing loop. At the transmitter side, an input binary sequence is mapped to a message symbol $s \in \{1, 2, \ldots, M\}$, where $M = 2^k$. The message symbols are encoded by a neural network–based encoder, which replaces the conventional channel encoder and directly maps each message to an n-dimensional real-valued vector $\mathbf{x} \in \mathbb{R}^n$. A normalization stage is applied at the encoder output in order to satisfy the transmit power constraints. In this work, both energy normalization, enforcing $\|\mathbf{x}\|_2^2 \leq n$, and average power normalization, enforcing $\mathbb{E}[|x_i|^2] \leq 1$, are considered.

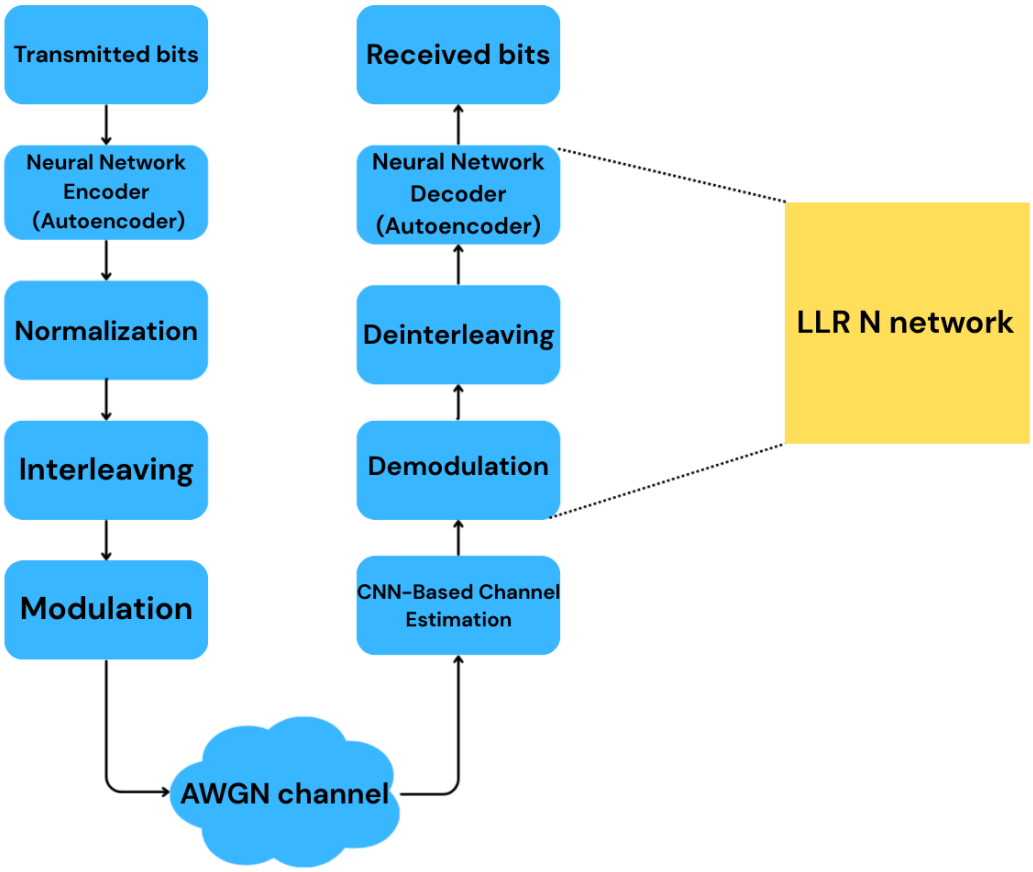

Figure 1. System model.

The normalized symbols are transmitted over a noisy wireless channel. In this work, the channel is modeled either as an additive white Gaussian noise (AWGN) channel or as a fading channel generated using clustered delay line (CDL) models.

The received signal vector is given by

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

where $\odot$ denotes element-wise Hadamard multiplication, $\mathbf{h}$ represents the channel coefficient vector, and $\mathbf{w}$ is the additive noise vector. This formulation is consistent with discrete-time baseband channel models widely adopted in the wireless communication literature [19]. For the AWGN channel, $\mathbf{h} = \mathbf{1}_n$, whereas for fading channels $\mathbf{h}$ is generated according to the clustered delay line (CDL) model [20]. The noise vector $\mathbf{w}$ is modeled as zero-mean additive white Gaussian noise [3]. Although Gaussian noise is assumed in this work for benchmarking purposes, the proposed neural network-based transceiver framework is not inherently restricted to traditional analytic channel models and can be extended to accommodate diverse and dynamic channel conditions through appropriate training.

In addition to the main encoding and decoding chain, auxiliary processing blocks are implemented at the receiver side. A convolutional neural network (CNN) is employed for modulation classification based on the received IQ samples, while a CNN-based channel estimator is used to estimate the channel response from pilot symbols or received OFDM grids. These auxiliary modules provide side information to the receiver processing chain and improve robustness under varying channel conditions. Optionally, a soft demodulation block can be used to generate log-likelihood ratio (LLR) estimates from the received symbols, as studied in previous works [21].

4. Simulation Results

All neural network training and simulation tasks were conducted using MATLAB version 24.2 (R2024b) and the 5G Toolbox on a system without a dedicated GPU. The computational environment consisted of a Windows 10 Pro 64-bit operating system running on an Intel Core i5 processor clocked at 2.90 GHz. The machine featured 15.0 GB of dual-channel DDR4 RAM operating at 1063 MHz and a Gigabyte Technology Co., Ltd., H410M S2H V3 motherboard. All model training was carried out exclusively on the integrated CPU, ensuring that the experiments can be replicated on systems lacking GPU acceleration.

4.1. Constellation Diagrams Comparison Between Energy and Average Power Normalization

In the autoencoder-based communication system, the encoder maps each of the $M = 2^k$ possible messages to a vector $x \in \mathbb{R}^n$, with constraints applied to ensure physically realizable transmissions. Two constraints are commonly imposed on the encoder output: the energy constraint,

$$\|x\|_2^2 \leq n$$

and the average power constraint,

$$\mathbb{E}[|x_i|^2] \leq 1, \quad \forall i$$

which regulate the total transmit energy and per-symbol power, respectively [22]. These constraints shape the learned constellation and influence the autoencoder's behavior under noise.

Under average power normalization, the autoencoder utilizes geometric shaping. Unlike the energy constraint, which forces all symbols onto a strict hypersphere, the average power constraint allows the network to push certain symbols further from the origin (higher energy) while pulling others closer (lower energy). While this optimizes decision regions, it implicitly increases the Peak-to-Average Power Ratio (PAPR). This creates a critical trade-off: the model achieves better error performance at high SNRs but requires high-linearity power amplifiers, representing a practical realization constraint that must be balanced against the observed geometric gains.

This shaping mimics the gains seen in higher QAM, where the network effectively maximizes the minimum Euclidean distance (dmin) between constellation points. This explains the performance gain at moderate to high SNRs, as the network optimizes the decision regions for the most likely noise vectors. However, this flexibility comes at a cost—the inner constellation points have significantly lower energy and are more prone to noise. This structural vulnerability explains the performance variability observed at low SNRs, where the noise magnitude frequently exceeds the smaller decision regions of the low energy symbols.

With energy normalization, the autoencoder typically produces constellation points that lie on a uniform energy shell. This regularity helps maintain consistent symbol robustness and supports well-structured decision regions in the latent space.

Figure 2 shows the constellation learned under average power normalization for the (7,4) autoencoder. In contrast to the energy-constrained case, the points are less uniform and exhibit noticeable asymmetry. This indicates that the network allocates power unevenly across symbols, effectively prioritizing some message vectors to improve decoding reliability. While such adaptive shaping can be advantageous under varying SNR conditions, allowing the model to emphasize more error sensitive symbols, it also produces more irregular decision boundaries, increasing susceptibility to misclassification at low SNR values.

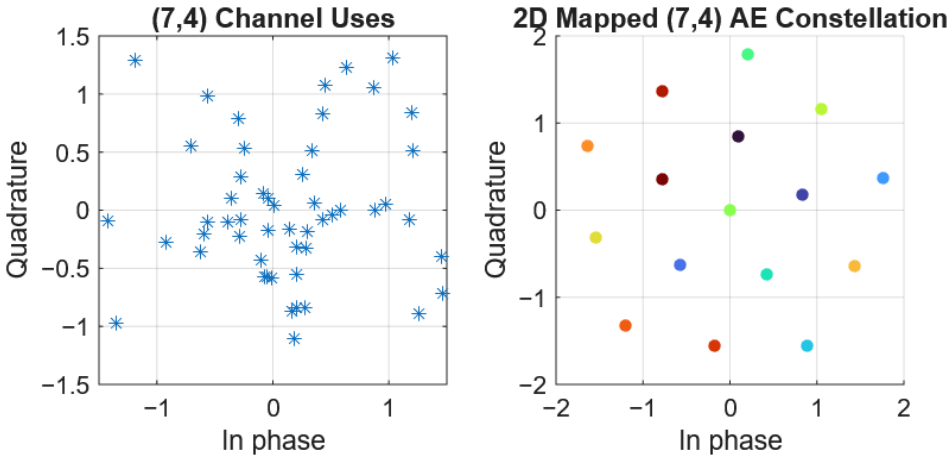

Figure 2. (7,4) autoencoder average power normalization constellation diagram.

4.2. Impact of Normalization Strategies on (7,4) Autoencoder Performance

Figure 3 compares the block error rate (BLER) performance of the proposed (7,4) autoencoder-based coding scheme with that of the classical (7,4) Hamming code under both hard decision decoding (HDD) and maximum likelihood decoding (MLD), using energy normalization for a fair comparison. The results show that the autoencoder closely tracks the performance of Hamming MLD across the entire SNR range, outperforming the conventional Hamming HDD upper bound by a significant margin. In the moderate-to-high-SNR regime, the learned code even exhibits a slight performance gain over Hamming MLD, demonstrating the ability of neural network-based encoders and decoders to discover signal constellations and decision regions that improve error correction capability under practical channel conditions.

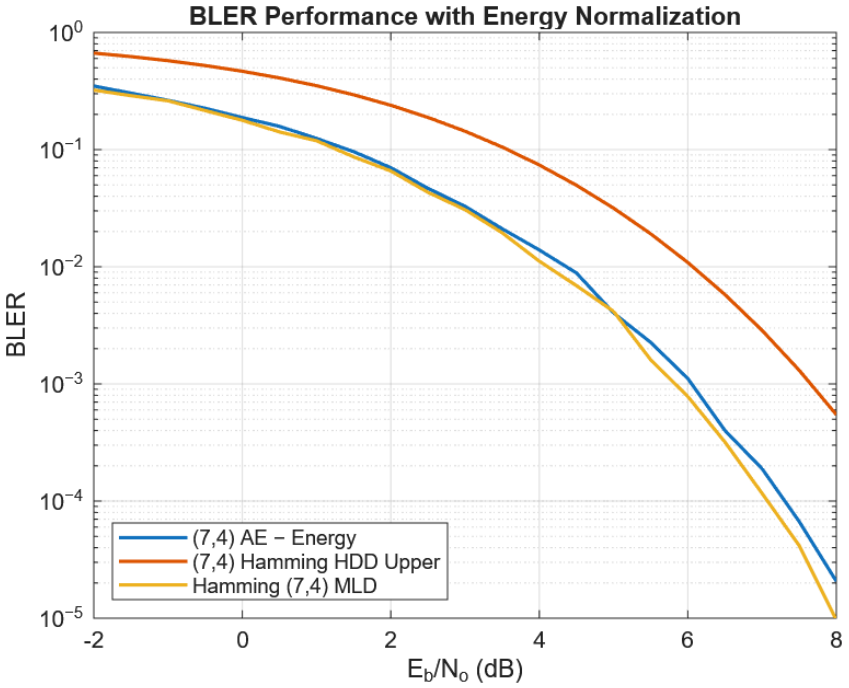

Figure 3. Energy normalization.

Figure 4 presents the results when the autoencoder is trained with average power normalization, which uses an average power constraint across all data, rather than equalizing their individual energies. In the moderate-to-high-SNR regime, average power normalization achieves a BLER performance that is statistically equivalent to the energy

normalized case, with marginal gains observed only at SNR levels exceeding 6 dB relative to the energy normalized case, while maintaining a clear advantage over the Hamming HDD. However, this adaptivity introduces greater variability in performance across different SNRs, as the model can allocate energy unevenly, occasionally leading to suboptimal behavior in low-SNR conditions. These observations highlight a trade-off between consistency and adaptability.

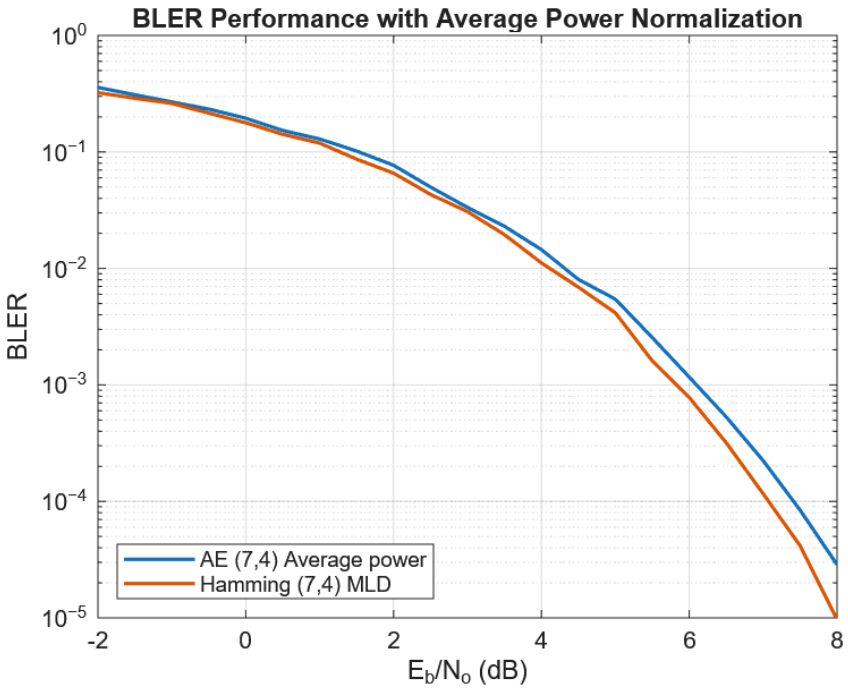

Figure 4. Average power normalization.

The two normalization schemes show that the choice of normalization has a heavy influence on both the learning dynamics and error rate characteristics of the autoencoder-based communication system. Energy normalization provides more stable performance and reliable generalization across SNRs, while average power normalization offers marginal performance gains through flexible energy allocation. The overall results confirm that neural autoencoders can approximate the performance of classical maximum likelihood decoders.

4.3. Performance Comparison of (7,4) Hamming Code and Autoencoder-Based Schemes

Figure 5 presents the block error rate (BLER) performance of a conventional (7,4) Hamming code and a variety of AEs. As expected, all schemes exhibit monotonically decreasing BLER with increasing SNR, confirming improved reliability under less noisy channel conditions. The autoencoder-based models consistently outperform the HDD upper bound, demonstrating that the learned encoder–decoder pair captures code structure and redundancy more effectively than conventional hard decision decoding. However, their performance remains slightly inferior to that of the ideal MLD curve, indicating that while the autoencoder approximates optimal decoding, it does not yet achieve the full maximum-likelihood capability. Notably, the models trained at intermediate SNRs (2 dB and 3 dB) achieve lower BLERs across a wider range of SNRs compared with the model trained at a higher SNR (7 dB). This finding suggests that training under moderately noisy conditions leads to better generalization and robustness, whereas training in overly clean environments limits the model's adaptability.

In Figure 6, we can see that, as expected, the uncoded QPSK (theoretical and simulated) is outperformed by the (7,4) AE and Hamming ML, maintaining a higher BLER while the

SNR increases. The hard decision (7,4) Hamming code with QPSK modulation typically offers about a 0.6 dB gain compared to uncoded QPSK, while maximum-likelihood (ML) decoding provides an additional 1.5 dB advantage in BLER performance. Notably, the (7,4) autoencoder, when trained at 3 dB, achieves BLER results that closely approach those of the Hamming ML decoder, effectively demonstrating a total coding gain of approximately 2 dB for a coding rate of R = 4/7 [22]. The (7,4) autoencoder achieves a lower BLER over the full SNR range, highlighting its ability to efficiently learn the coding and decoding mappings across varying SNR values. The most significant observation made is the fact that the autoencoder is very close and in some cases aligns with the theoretical performance limit that Hamming ML provides. This suggests that the autoencoder is very close to reaching the full potential of maximum-likelihood performance. Our results further confirm the fact that neural network-based coding schemes clearly outperform traditional decoding methods under constraints.

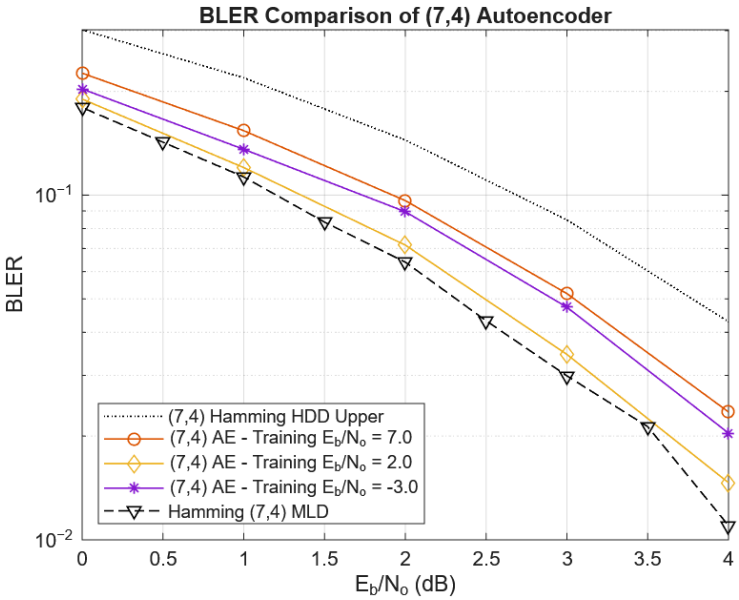

Figure 5. (7,4) Hamming code compared to AEs.

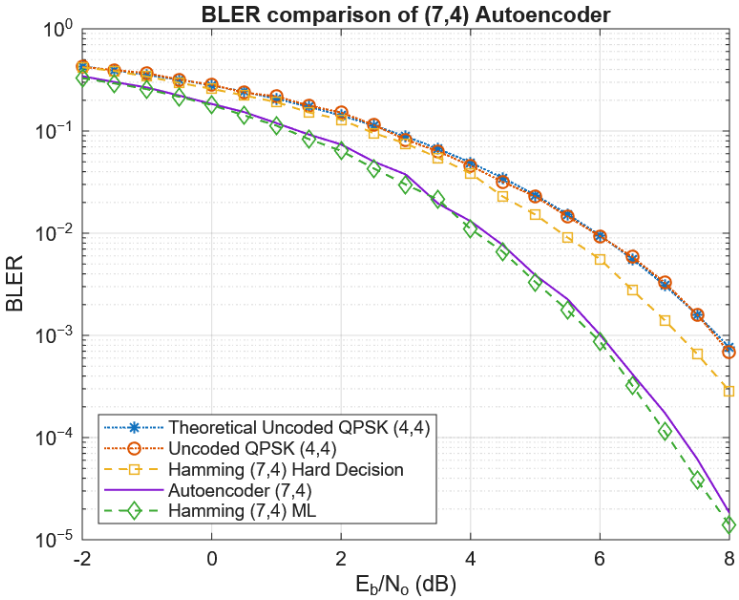

Figure 6. Performance comparison of (7,4) Hamming code and autoencoders.

Overall, it is highlighted that deep learning based communication systems can learn near optimal coding strategies directly from data, without relying on explicit algebraic code structures. The learned models bridge the gap between simple HDD and optimal maximum-likelihood decoding, thereby demonstrating the potential of neural autoencoders to serve as effective end-to-end communication components. Furthermore, the observed dependence of performance on the training SNR emphasizes the importance of selecting an appropriate noise level during learning to balance specialization and generalization.

4.4. Modulation Classification with Deep Learning

In order to evaluate the ability of deep learning techniques in identifying signal types, we implement a modulation classification framework based on MATLAB's Deep Learning Toolbox [23]. This framework utilizes a CNN to automatically extract the features and classify the modulation schemes. Following the existing setup while changing various parameters, we aim to highlight the importance of the channel's conditions in modulation classification.

To identify signal types, we implement a 28-layer CNN. Despite its depth, the architecture remains computationally efficient for real-time inference on the tested CPU-based system. To mitigate the risk of overfitting, we employed a dropout rate of 0.5 and a 10% validation split. Training with 10,000 frames per modulation type ensures the network captures generalized features rather than memorizing noise patterns. The specific simulation parameters used for this framework are summarized in Table 2.

Table 2. Simulation parameters for modulation classification.

Parameter	Value
Modulation types	BPSK, QPSK, 8PSK, 16QAM, 64QAM, PAM4, GFSK, CPFSK, B-FM, DSB-AM, SSB-AM
Number of symbols per frame	1024
Samples per symbol (sps)	8
Sample rate (fs)	200 kHz
Center frequencies	902 MHz (digital), 100 MHz (analog)
SNR	10/30 dB
Training frames per modulation	10,000
Training/validation/test split	80%/10%/10%
Network architecture	28-layer CNN
Training epochs	20
Mini-batch size	1024

We consider five different modulation schemes: BPSK, QPSK, 16QAM, 64QAM, and 8PSK. For each frame, 1024 modulation symbols are generated and used to train a convolutional neural network (CNN) designed to classify these modulation types. The CNN consists of 28 layers, including convolutional, ReLU activation, pooling, normalization, dropout, softmax, as well as input and output layers. Training is conducted under various signal-to-noise ratio (SNR) conditions to evaluate the network's modulation classification performance. The resulting classification outcomes are presented in Figure 7 for a low SNR scenario and Figure 8 for a high SNR scenario.

At low SNRs, the received signal constellation becomes highly dispersed due to strong noise interference, making it difficult to distinguish between different modulation schemes. This increased overlap among constellation points leads to a greater likelihood of misclassification. In contrast, at high SNR levels, the noise effect is significantly reduced, resulting in tightly clustered and clearly separated constellation points. Consequently, the modulation schemes become much easier for the CNN to identify accurately.

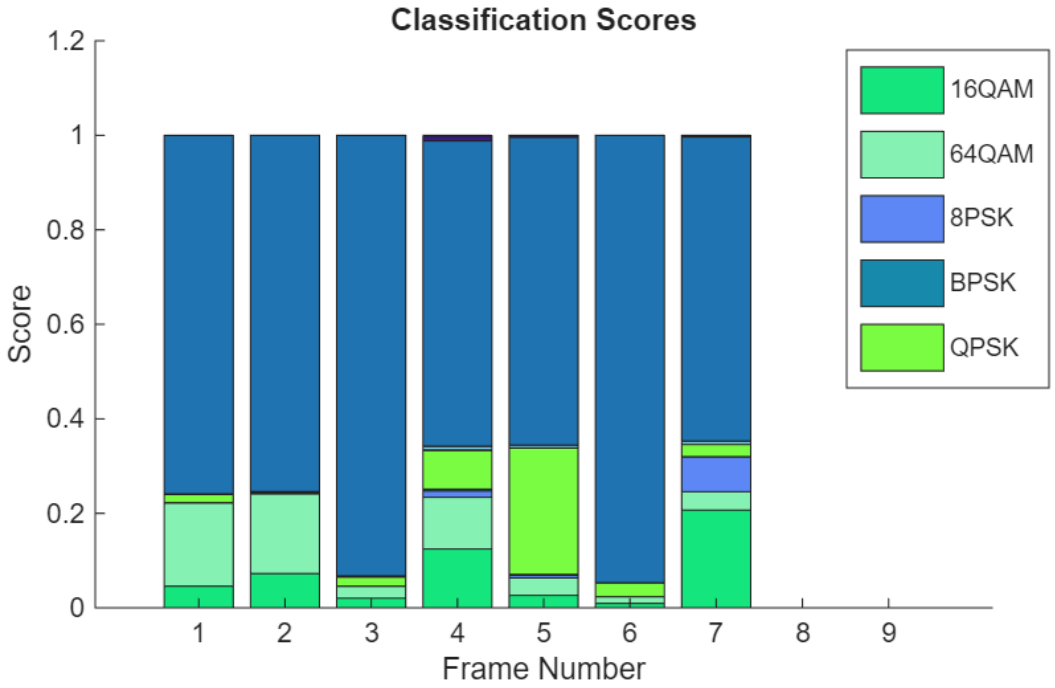

Figure 7. Modulation classification at 10 dB.

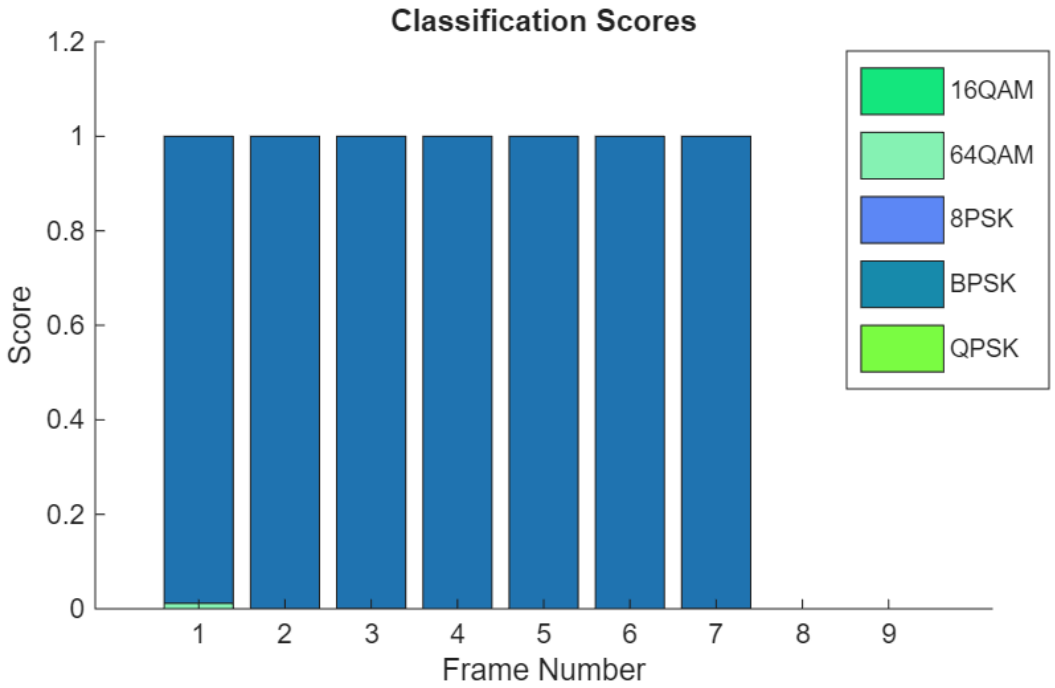

Figure 8. Modulation classification at 30 dB.

The CNN-based modulation classification framework demonstrates a strong dependence on the SNR ratio. At low SNR levels, noise-induced distortion causes significant scattering in the received constellations, reducing classification accuracy. However, as the SNR increases, the constellations become more compact and distinguishable, enabling the CNN to more reliably identify the underlying modulation schemes. This observation confirms that higher SNR conditions substantially enhance the performance and robustness of modulation recognition systems based on deep learning.

Figure 9, using the simulated data, demonstrates a high level of accuracy across the modulation schemes. However, we observe confusion among modulations with similar

characteristics like structure or spectral features (e.g., 16QAM and 64QAM). The overall classification accuracy exceeds 95%, confirming that most modulations were correctly identified.

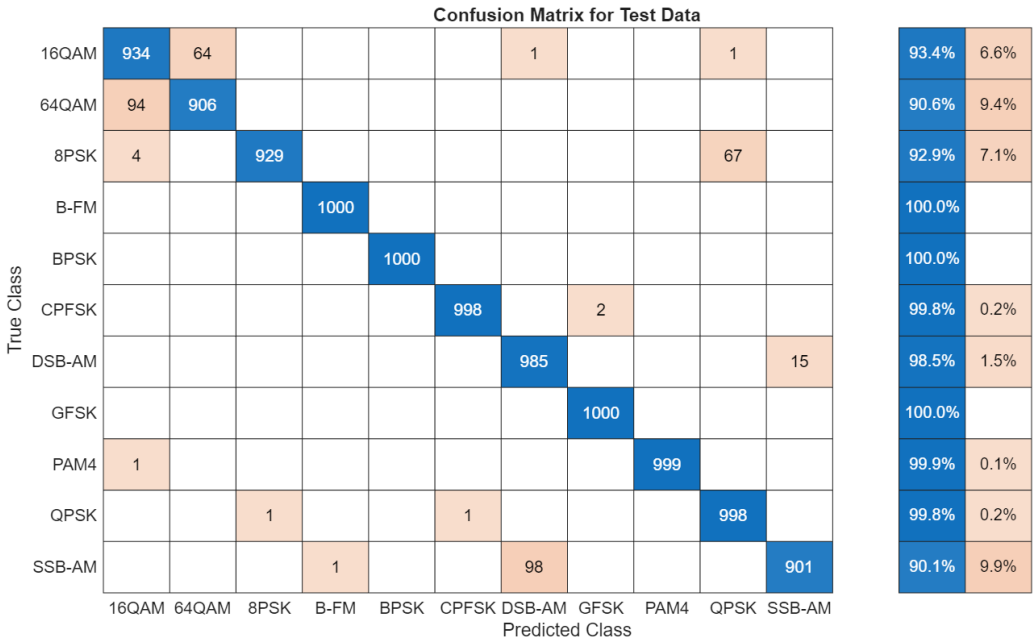

Figure 9. Confusion matrix using test data.

At lower SNRs, or even at 30dB, as seen in the confusion matrix, the inner constellation points of a 64QAM signal are geometrically identical to the points of a 16QAM signal. When noise blurs the outer boundaries of the constellation, the classifier struggles to distinguish whether a received symbol belongs to the inner ring of a 64QAM signal or a standard 16QAM transmission. Despite this structural ambiguity, the overall classification accuracy exceeds 95%, confirming that the system effectively distinguishes between diverse modulation types even when their spectral and spatial properties are closely related.

4.5. Deep Learning Data Synthesis for Channel Estimation

Additionally, we further investigate the effects of deep learning for channel estimation by employing a CNN to estimate the channel response between transmit and receive antennas, utilizing MATLAB's 5G Toolbox [20]. The CNN-based estimator predicts channel responses by learning from Demodulation Reference Signals' (DM-RS) pilot symbols. The pilot allocation pattern follows a sparse grid within the OFDM symbols. This process includes learning from specific pilot symbols, synthesizing a dataset of channel realizations using CDL, and then the network is trained to minimize the MSE between the estimated and actual paths. The key simulation and network training parameters for this investigation are summarized in Table 3.

We compare the deep learning model against two classical estimation techniques. The first is a baseline linear Interpolation, which performs simple interpolation between pilot subcarriers. The second, referred to as the Practical Estimator, represents a robust industry standard baseline utilizing MATLAB's nrChannelEstimate [20]. This estimator performs Least Squares (LS) estimation on the Demodulation Reference Signals (DM-RS), followed by a 2D averaging and interpolation filter across both time and frequency domains.

By averaging out noise across the resource grid, the Practical Estimator effectively mitigates Gaussian noise, providing a near ideal reference. The CNN's ability to approach this performance without explicit knowledge of the pilot grid structure demonstrates its capacity to learn the underlying channel statistics implicitly.

Table 3. Key parameters for neural network training and channel estimation.

Parameter	Value
Testing size	64
SNR (dB)	10/30
Training batch size	8
Number of transmit antennas	1
Number of receive antennas	1
Number of training examples	256
Pretrained training examples	16,384
OFDM subcarriers	612
OFDM symbols per slot	14
Training epochs	10

To assess the model's generalization ability, the CNN was tested on 64 previously unseen channel realizations. For each realization, the network predicted the channel response, and the corresponding mean squared error (MSE) was computed. The resulting MSE distribution, illustrated in Figure 10, indicates that most predictions exhibit errors below 0.06, with a dense concentration between 0.00 and 0.06. These results demonstrate that the CNN consistently delivers robust and dependable channel estimates under diverse conditions, though its precision is marginally lower compared to the practical estimator.

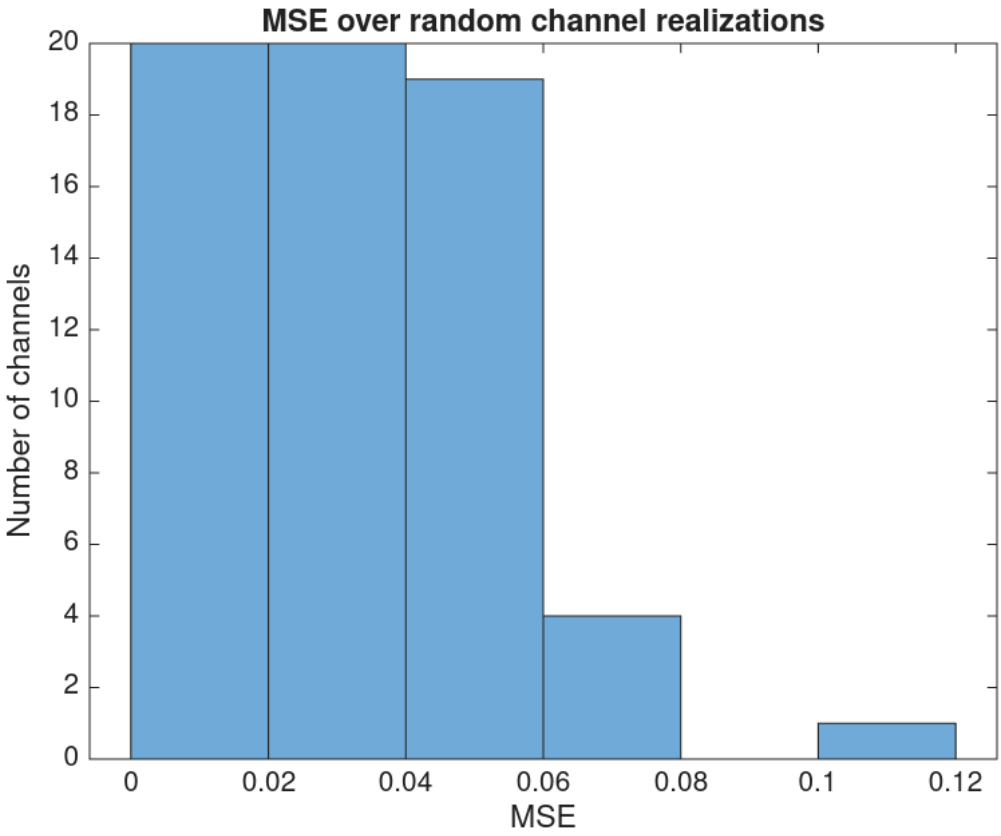

Figure 10. MSE over random channel realizations.

The performance of three different estimators is visualized below in both 10 and 30 dB. Figure 11 illustrates the reduced SNR, where we see an MSE of 0.18599 for the linear interpolation, while the practical estimator and neural network maintain a similar performance yielding 0.03229 and 0.0322413, respectively.

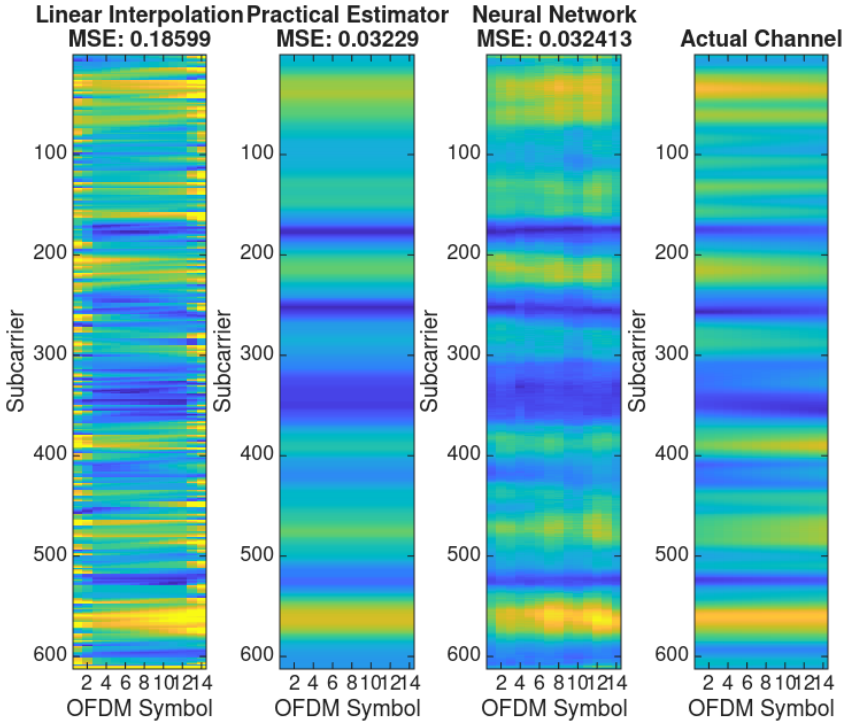

Figure 11. MSE at 10 dB.

Meanwhile, at an increased SNR of 30 dB as presented in Figure 12, we observe a significant improvement in the accuracy of the linear interpolation with an MSE of 0.019. The practical estimator demonstrates the highest accuracy with an MSE of approximately 0.00085, nearly replicating the actual channel. The CNN-based estimator achieves intermediate performance with an MSE of about 0.026, improving upon interpolation but falling short of the practical estimator's precision.

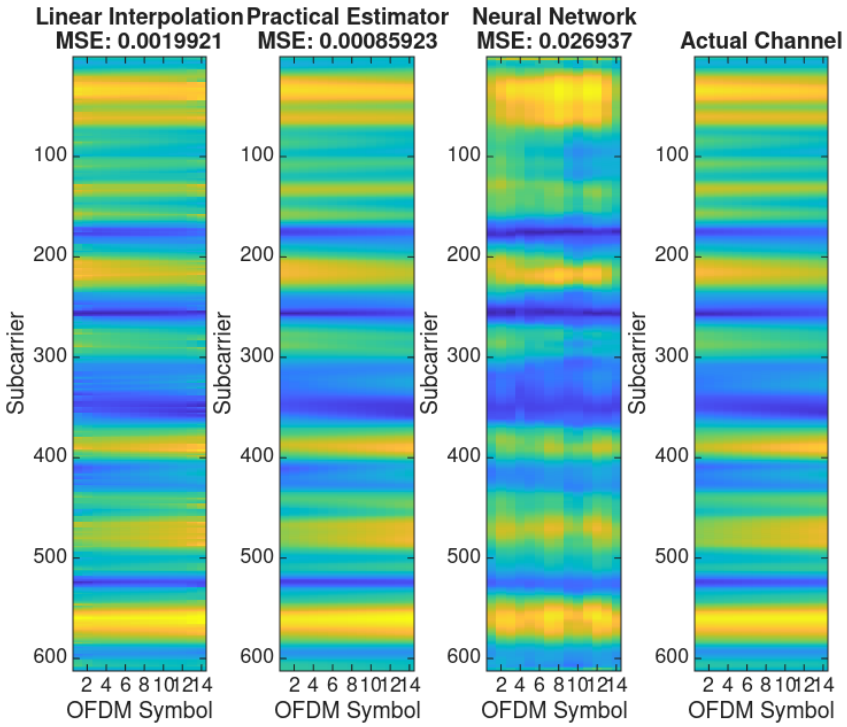

Figure 12. MSE at 30 dB.

Although the performance of the CNN aligns with traditional baselines in high-SNR regimes, its adoption is justified by its ability to remain functional without explicit knowledge of the pilot grid or channel statistics [7,12], offering superior robustness to model mismatch compared to standard estimators [3,19].

The results shown demonstrate that while the CNN-based estimator does not surpass the practical estimator, it achieves comparable and noticeable performance. Both of these methods outperform linear interpolation in low SNRs, which has high estimation errors and noise sensitivity. Another remarkable observation is the CNN's ability to maintain its accuracy across varying channel conditions and different SNR values, highlighting the robustness and potential as an alternative solution when the practical estimator is unavailable.

5. Conclusions

This paper explores key components of an AI-assisted physical layer, including autoencoders, deep learning-based modulation classification, and channel estimation. Several state-of-the-art methods are reviewed and extended by integrating trained neural networks into multiple physical layer functions. Unlike prior studies, this work experimentally demonstrates how multiple AI-enabled components can coexist and interact within a unified framework. Simulation results show consistent performance improvements across all evaluated tasks, highlighting the importance of effective training and demonstrating that deep learning can bridge the gap between theoretical baselines and practical implementations, offering a flexible alternative to conventional model-based signal processing chains.

Learned communication schemes such as autoencoders are shown to approach optimal decoding limits when design parameters, including normalization strategies and training noise levels, are carefully selected. A comparison between energy normalization and average power normalization reveals a trade-off between robustness across SNRs and constellation shaping gains at moderate-to-high SNRs. Neural network-based modulation classification and channel estimation successfully capture signal and channel characteristics without explicit analytical models. Modulation classification achieves high accuracy across diverse modulation schemes with SNR-dependent performance, while CNN-based channel estimators consistently outperform interpolation methods and closely approach practical industry baselines.

Overall, the results indicate a shift toward intelligent, data-driven physical layers capable of adapting to complex and dynamic channel conditions. The system-level evaluation highlights how improvements or limitations in individual AI-enabled modules propagate through the receiver chain when deployed jointly. Future work should focus on developing fully end-to-end learned communication systems that jointly optimize multiple physical-layer functions, as well as transitioning these models from simulation to over-the-air deployment to address real-world challenges such as hardware impairments and security concerns.

Author Contributions: Conceptualization, A.C., D.K., A.X. and C.C.; methodology, A.C., D.K., A.X. and C.C.; software, A.C., D.K., A.X. and C.C.; validation, A.C., D.K., A.X. and C.C.; formal analysis, A.C., D.K., A.X. and C.C.; investigation, A.C., D.K., A.X. and C.C.; resources, A.C., D.K., A.X. and C.C.; data curation, A.C., D.K., A.X. and C.C.; writing—original draft preparation, A.C.; writing—review and editing, D.K., A.X. and C.C.; visualization, A.C. and D.K.; supervision, A.X. and C.C.; project administration, D.K. and A.X.; funding acquisition, C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bazzi, A.; Bomfin, R.; Mezzavilla, M.; Rangan, S.; Rappaport, T.; Chafii, M. Upper Mid-Band Spectrum for 6G: Vision, Opportunity and Challenges. *IEEE Commun. Mag.* **2025**, *64*, 206–212. [CrossRef]
2. Kim, H. *Artificial Intelligence for 6G*; Springer International Publishing: Cham, Switzerland, 2022. [CrossRef]
3. O'Shea, T.J.; Hoydis, J. An Introduction to Deep Learning for the Physical Layer. *IEEE Trans. Cogn. Commun. Netw.* **2017**, *3*, 563–575. [CrossRef]
4. Felix, A.; Cammerer, S.; Dörner, S.; Hoydis, J.; Ten Brink, S. OFDM-Autoencoder for End-to-End Learning of Communications Systems. In *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*; IEEE: New York, NY, USA, 2018. [CrossRef]
5. Nachmani, E.; Marciano, E.; Lugosch, L.; Gross, W.J.; Burshtein, D.; Be'ery, Y. Deep Learning Methods for Improved Decoding of Linear Codes. *IEEE J. Sel. Top. Signal Process.* **2018**, *12*, 119–131. [CrossRef]
6. Dörner, S.; Cammerer, S.; Hoydis, J.; ten Brink, S. Deep Learning Based Communication Over the Air. *IEEE J. Sel. Top. Signal Process.* **2018**, *12*, 132–143. [CrossRef]
7. Ye, H.; Li, G.Y.; Juang, B.H.F. Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 114–117. [CrossRef]
8. Shen, Y.; Shi, Y.; Zhang, J.; Letaief, K.B. A Graph Neural Network Approach for Scalable Wireless Power Control. In *2019 IEEE Globecom Workshops (GC Wkshps)*; IEEE: New York, NY, USA, 2019. [CrossRef]
9. Rajendran, S.; Meert, W.; Giustiniano, D.; Lenders, V.; Pollin, S. Deep Learning Models for Wireless Signal Classification with Distributed Low-Cost Spectrum Sensors. *IEEE Trans. Cogn. Commun. Netw.* **2018**, *4*, 433–445. [CrossRef]
10. O'Shea, T.J.; Corgan, J.; Clancy, T.C. Convolutional Radio Modulation Recognition Networks. In *Engineering Applications of Neural Networks*; Communications in Computer and Information Science; Springer: Cham, Switzerland, 2016; Volume 629. [CrossRef]
11. West, N.E.; O'Shea, T. Deep Architectures for Modulation Recognition. In *2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*; IEEE: New York, NY, USA, 2017. [CrossRef]
12. Soltani, M.; Pourahmadi, V.; Mirzaei, A.; Sheikhzadeh, H. Deep Learning-Based Channel Estimation. *IEEE Commun. Lett.* **2019**, *23*, 652–655. [CrossRef]
13. He, H.; Wen, C.K.; Jin, S.; Li, G.Y. Deep Learning-Based Channel Estimation for Beamspace mmWave Massive MIMO Systems. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 852–855. [CrossRef]
14. Huang, H.; Yang, J.; Huang, H.; Song, Y.; Gui, G. Deep Learning for Super-Resolution Channel Estimation and DOA Estimation Based Massive MIMO System. *IEEE Trans. Veh. Technol.* **2018**, *67*, 8549–8560. [CrossRef]
15. Liu, X.; Yang, D.; Gamal, A.E. Deep Neural Network Architectures for Modulation Classification. In *2017 51st Asilomar Conference on Signals, Systems, and Computers*; IEEE: New York, NY, USA, 2017. [CrossRef]
16. O'Shea, T.J.; Roy, T.; Clancy, T.C. Over-the-Air Deep Learning Based Radio Signal Classification. *IEEE J. Sel. Top. Signal Process.* **2018**, *12*, 168–179. [CrossRef]
17. Mhawi, D.N.; Oleiwi, H.W.; Al-Raweshidy, H. Towards Intelligent Threat Detection in 6G Networks Using Deep Autoencoder. *Electronics* **2025**, *14*, 2983. [CrossRef]
18. Alnaseri, O.; Alzubaidi, L.; Himeur, Y.; Ala'anzy, M.A.; Timmermann, J.; Gismalla, M.S.M. A Review on Deep Learning Autoencoder in the Design of Next-Generation Communication Systems. *arXiv* **2024**, arXiv:2412.13843. [CrossRef]
19. Tse, D.; Viswanath, P. *Fundamentals of Wireless Communication*; Cambridge University Press: Cambridge, UK, 2005. [CrossRef]
20. MathWorks. Deep Learning Data Synthesis for 5G Channel Estimation. 2024. Available online: https://www.mathworks.com/help/5g/ug/deep-learning-data-synthesis-for-5g-channel-estimation.html (accessed on 15 June 2025).
21. Christopoulou, A.; Chaikalis, C.; Kosmanos, D.; Xenakis, A.; Chatzimisios, P. On the use of AI for 6G Signal Decoding in DVB – S.2 Applications. In *2024 IEEE Conference on Standards for Communications and Networking (CSCN)*; IEEE: New York, NY, USA, 2024. [CrossRef]
22. MathWorks. Autoencoders for Wireless Communications—MATLAB & Simulink. 2024. Available online: https://www.mathworks.com/help/comm/ug/autoencoders-for-wireless-communications.html (accessed on 15 June 2025).
23. MathWorks. Modulation Classification with Deep Learning. 2024. Available online: https://www.mathworks.com/help/deeplearning/ug/modulation-classification-with-deep-learning.html (accessed on 15 June 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.