19L820-Project Work 2

BATCH NO:19

Deep Learning based Channel Estimation For OFDM Wireless Communication

GUIDE NAME: Dr THIYAGARAJAN K

TEAM MEMBERS

22L255 - RAGHAVAN

22L262 - SARRANADHITHIYAA

22L268 - SHRIRAM

22L270 - SHYAAMALAN

Problem Statement

Accurate **Channel State Information (CSI)** is critical for the performance of **OFDM-based wireless communication systems**. However, traditional estimation methods, such as **Least Squares (LS)** and **Linear Minimum Mean Squared Error (LMMSE)**, face significant challenges:

- **Sensitivity to Noise:** Traditional methods often suffer from severe performance degradation in **low Signal-to-Noise Ratio (SNR)** environments.
- **Statistical Dependency:** LMMSE requires prior knowledge of channel statistics, which are often unavailable or vary dynamically in real-world scenarios.
- **Limitations of Existing DL Models:** While early Deep Learning (DL) approaches like **CNNs** offer improvements, they often fail to capture **long-range time-frequency correlations** inherent in wireless channels due to their local receptive fields.

This project addresses these limitations by developing a software-based framework that cascades an **Initial Denoising Network (IDN)** with a **Transformer-based model**. This approach aims to suppress pilot noise and exploit **global correlations** across the OFDM grid to reconstruct high-fidelity CSI without requiring prior channel statistics, ultimately achieving superior accuracy and robustness compared to conventional benchmarks

Literature Survey 1

Deep-Learning Based Channel Estimation for OFDM Wireless Communications

Authors: Guoda Tian, Xuesong Cai, Tian Zhou, Weinan Wang, Fredrik Tufvesson
Journal: IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)
Year: 2022

Work Done:

- Proposed a deep learning–based channel estimation pipeline for OFDM systems to estimate full CSI from limited pilot symbols without increasing pilot power or density.
- Introduced an **Initial Denoise Network (IDN)** and a **Resolution Enhancement Network** (REN) using fully connected layers to improve pilot SNR and recover high-resolution CSI.
- Achieved significant performance gains (10–15 dB MSE improvement) over LMMSE and ChannelNet, especially at low SNR, with a compact model suitable for embedded systems.

Limitations:

- Evaluated only for SISO OFDM systems.
- Requires supervised training with large labeled datasets.

Literature survey 2

Deep Learning-Based Channel Estimation with Transformer

Authors: J. Guo, et al.
Journal: IEEE Access (2021)
Year: 2021

Work Done:

- **Emphasized the use of Self-Attention** to effectively capture the complex, non-local time-frequency correlations inherent in wireless channels.
- Demonstrated that the Transformer model outperforms CNN-based models by achieving a **lower NMSE (Normalized Mean Square Error)**
- Confirmed that the Transformer architecture is uniquely suited for processing sequential and global grid data by integrating **Positional Encoding** to preserve the time-frequency structure.

Limitations:

- The implementation is **computationally expensive** due to the high complexity of the self-attention mechanism, potentially increasing training time compared to simpler CNN/MLP models.
- Performance is highly dependent on the **quality of the positional encoding** scheme used.

Literature survey 3

Denoising Generalization Performance of Channel Estimation in Multipath Time-Varying OFDM Systems
Author: Yinying Li, Xin Bian, Mingqi Li
Journal: Sensors (MDPI)
Year: 2023

Work Done:

• Designed a deep network with Denoising CNN (DnCNN) to **suppress AWGN** from pilot-based CSI matrices.
• Modeled the channel frequency response as a 2D grid and applied **residual learning** to refine the initial LS/LMMSE estimate.
• Achieved **lower NMSE** and better generalization across varying SNR and Doppler conditions.

Limitation:

• Depends on the accuracy of the initial LS/LMMSE estimate; **large errors propagate** into the denoiser.
• Requires channel/noise statistics implicitly learned from large labeled datasets.
• **Increased computational** and **memory cost** due to deep CNN and residual blocks.

Literature survey 4

ChannelNet: A Deep Neural Network for Channel Estimation in OFDM Systems

Authors: Ye Yuan, Guanding Yu, Hailin Zhang, Yonghui Li
Journal: IEEE Wireless Communications Letters
Year: 2019

Work Done:

- Proposed **ChannelNet**, a CNN-based deep learning framework for OFDM channel estimation.
- Designed a two-stage architecture consisting of a **super-resolution CNN** and an **image restoration CNN** to reconstruct high-quality CSI from sparse pilot signals.
- Treated channel estimation as an image recovery problem by representing CSI as a 2D time–frequency grid.
- Demonstrated **improved NMSE performance** compared to traditional LS and MMSE estimation methods.

Limitations:

- CNN architecture mainly captures **local spatial features** and struggles to learn long-range time–frequency dependencies.
- Requires **high computational resources** for training deeper CNN layers.
- Performance **depends heavily** on pilot distribution and training **dataset** diversity.

Literature survey 5

A DFT-based Low Complexity LMMSE Channel Estimation Technique for OFDM Systems

Authors: Jyoti Prasanna Patra, Bibhuti Bhusan Pradhan, Poonam Singh

Journal: Journal of Telecommunications and Information Technology (JTIT)

Year: 2022

Work Done

- **Reduced the computational complexity** of conventional LMMSE by exploiting DFT properties and structured correlation.
- Achieved near-optimal NMSE/BER performance compared to standard LMMSE and significantly better performance than LS.
- Demonstrated that LMMSE remains a strong and commonly used baseline in OFDM channel estimation.

Limitation

- Still **requires channel statistics / covariance information**, which may not always be accurately available in practical systems.
- **Performance** depends on the **assumed** channel correlation model.
- Although **complexity** is reduced, it is still higher than LS and may be **heavy** for **very low-power devices**.

OBJECTIVE

- To develop a software-based OFDM channel estimation framework using deep learning techniques
- To analyze the limitations of traditional channel estimation methods under noisy and low-SNR conditions
- To design and implement a **pilot denoising network** to improve the quality of received channel information
- To apply a **Transformer-based model** for learning global time–frequency channel correlations
- To achieve improved channel estimation accuracy compared to conventional and existing deep learning approaches
- To evaluate system performance using error metrics such as **Mean Squared Error** (MSE) across different SNR levels

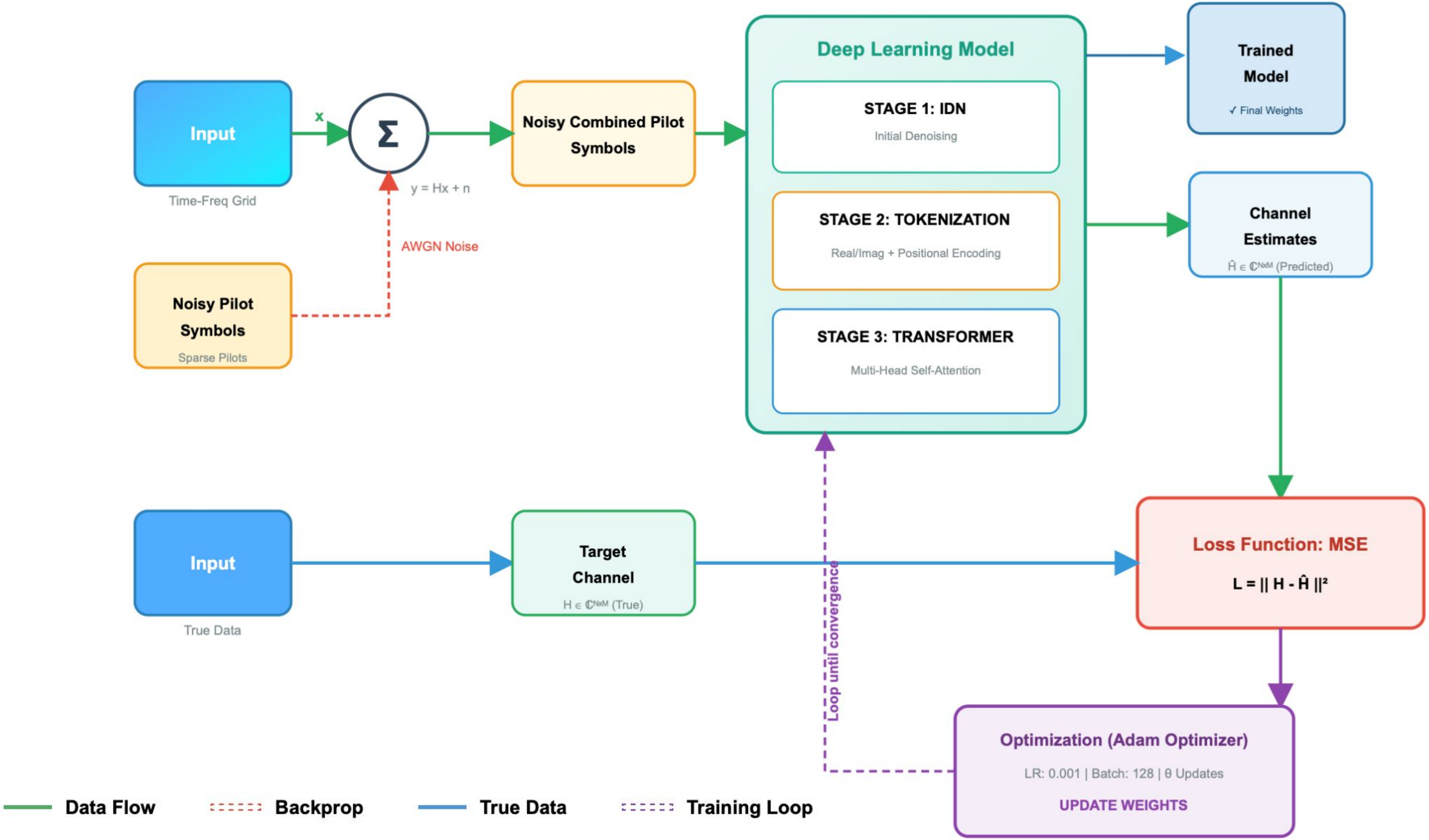
SCOPE OF THE PROJECT

- **Software-Defined Deep Learning Framework:** Engineering a comprehensive software-based estimation pipeline specifically designed for high-fidelity CSI recovery in OFDM wireless systems.
- **Realistic Channel Environment Simulation:** Modeling complex wireless signal conditions utilizing Rayleigh Fading and Additive White Gaussian Noise (AWGN) to analyze estimation robustness.
- **Transformer Architecture:** Implementing a Transformer-based model to exploit multi-head self-attention mechanisms for learning global time-frequency correlations across the resource grid.
- **Quantitative Performance Evaluation:** Assessing system accuracy through standardized MSE and NMSE metrics across a dynamic range of Signal-to-Noise Ratio (SNR) levels.
- **Standardized Benchmarking:** Conducting a comparative analysis to validate the proposed model against conventional Least Squares (LS) and Linear Minimum Mean Squared Error (LMMSE) techniques.
- **Simulation-Based Research Boundaries:** Focus is restricted to high-performance simulation-based modeling using Python, PyTorch, and Google Colab to validate architectural efficiency rather than physical hardware implementation.

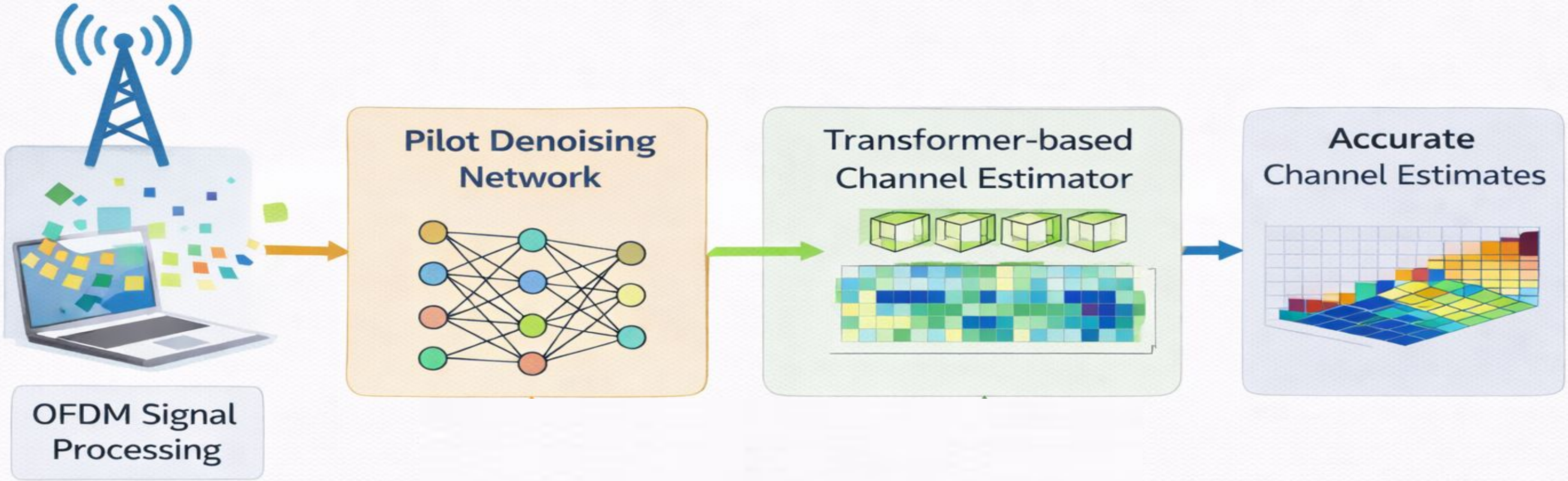
PROPOSED METHODOLOGY

- A software-based deep learning framework is developed for accurate OFDM channel estimation using pilot symbols only.
- The received pilot **Channel State Information (CSI)**, corrupted by noise, is first processed using an **Initial Denoising Network (IDN)** implemented as a fully connected neural network to suppress noise and enhance pilot quality.
- The denoised pilot CSI is then converted into a sequence of tokens by separating real and imaginary components and appending time–frequency positional information.
- A **Transformer-based channel estimation model** is employed to capture long-range correlations across subcarriers and OFDM symbols using self-attention mechanisms.
- The Transformer reconstructs the complete CSI of the OFDM time-frequency grid from the limited pilot information.
- The entire model is trained using a supervised learning approach with **Mean Squared Error (MSE) loss**, without requiring prior channel statistics or hardware measurements.
- **Performance** is evaluated under **multiple Signal-to-Noise Ratio** (SNR) conditions and compared with conventional estimation techniques to demonstrate robustness, particularly in low-SNR scenarios.

Training of OFDM Channel Estimation Model



Deep Learning-based OFDM Channel Estimation Framework

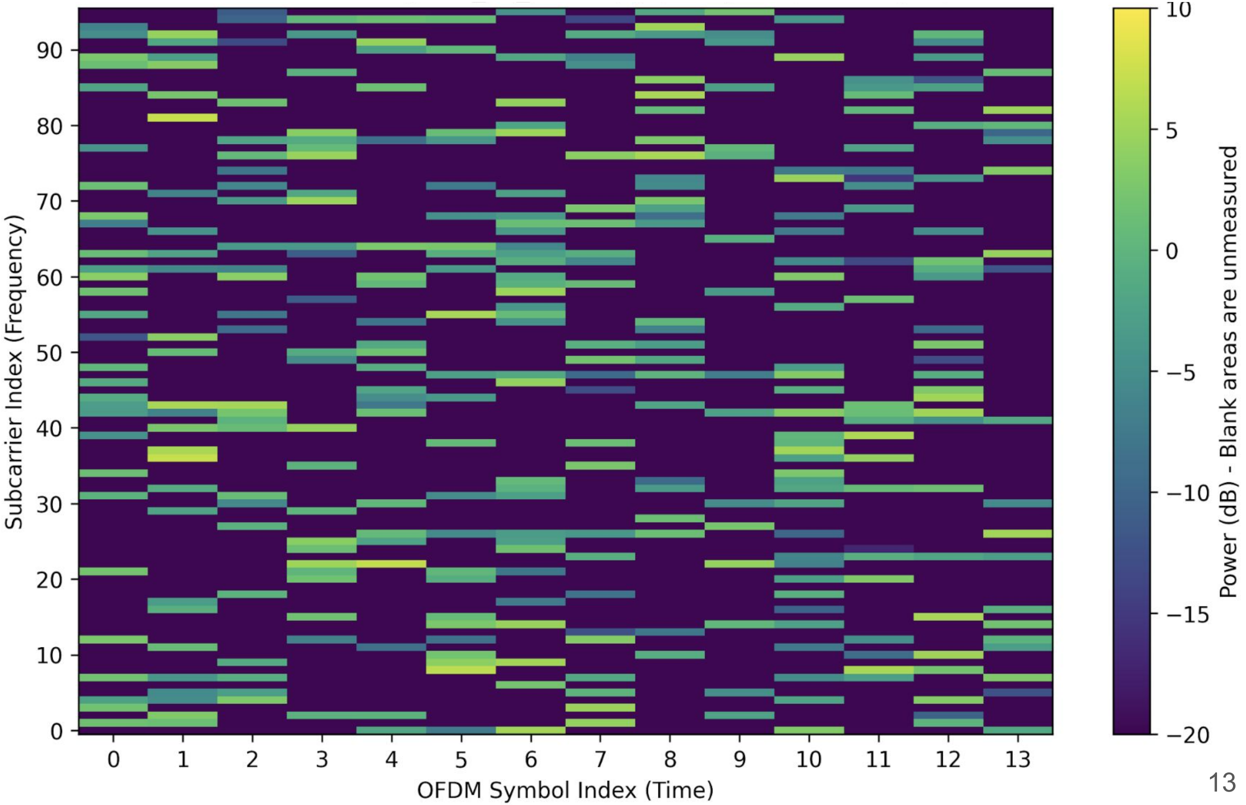


PROGRESS *[Dataset]*

Parameters of the dataset:

Parameter	Value
Channel Model	Rayleigh Fading
Multi-Path Profile	Exponential Power Decay
No. of Subcarriers (N_{SC})	96
No. of OFDM Symbols (N_{SYM})	14
Total Resource Elements	$96 \times 14 = \mathbf{1344}$
Pilot Pattern Type	Scattered
Pilot Density	$1/4$ **($\mathbf{25\%}$ of grid)**
Modulation	4-QAM / 16-QAM
Performance Metric	NMSE (Normalized MSE)

Sample 2D representation of the Single data:



PROGRESS *[till zeroth review]*

CNN Techniques used:

CNN MODEL COMPARISON TABLE

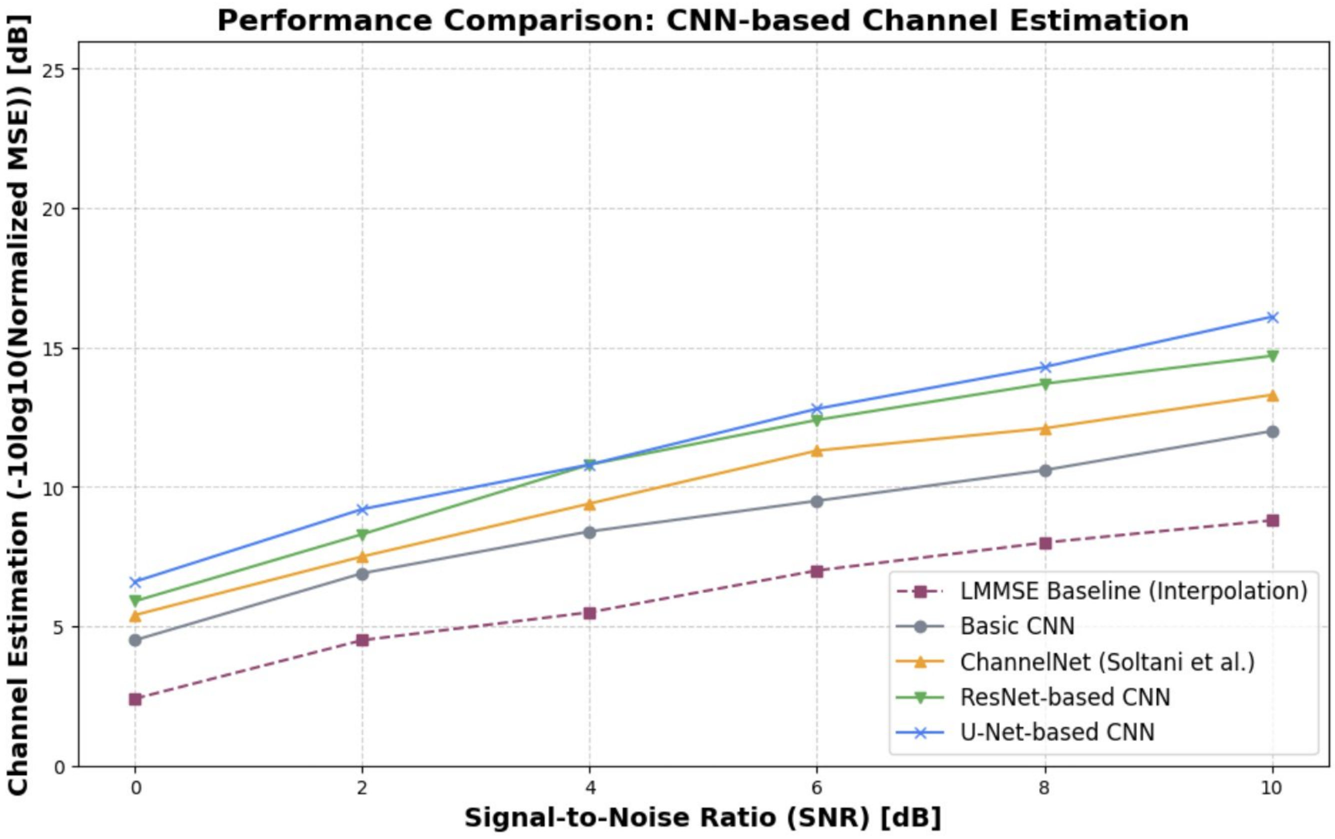
Model	Parameters	Depth/Type	Key Connections/Mechanism
basic_cnn	185,922	Simple 6-layer CNN (Encoder-Interpolation-Decoder)	None (Sequential)
channelnet	1,038,724	SRN (3 blocks) + IRN (5 blocks)	Residual (within blocks)
resnet	684,194	8x Residual Blocks (Deep Feature Extraction)	Residual (between Conv layers)
unet	1,782,978	Encoder-Decoder (2 levels)	Skip (Concatenation across levels)
attention_cnn	751,234	Encoder-Decoder with SE-style blocks	Channel Attention (Weighting)

<u>Dataset used:</u>

Parameter	Value
Channel Model	Rayleigh Fading
Multi-Path Profile	Exponential Power Decay
No. of Subcarriers (N_{SC})	96
No. of OFDM Symbols (N_{SYM})	14

Parameter	Value
Total Resource Elements	$96 \times 14 = \mathbf{1344}$
Pilot Pattern Type	Scattered / Sparse
Pilot Density	$1/4$ (**25%** of grid)
Modulation	16-QAM

PROGRESS*[till zeroth review]*



PROGRESS*[After Review]*

Present Techniques:

1. Least Square(dB):

$$\hat{H}_{\text{LS}}(k) = \frac{Y(k)}{X_{\text{pilot}}(k)}$$

Estimates the channel by simply **dividing the received pilot signal by the known transmitted pilot symbol**, minimizing the energy of the error in the pilot domain.

2. Linear Minimum Mean Squared Error (LMMSE) (dB)**:**

$$\hat{\mathbf{h}} = \mathbf{C}_{hx} \cdot \mathbf{C}_{xx}^{-1} \cdot \mathbf{x},$$

[C** matrices are **covariance/correlation matrices** calculated over the training data**]

It is a **statistical linear interpolation** method that uses pre-calculated **covariance matrices** to weight received pilot signals and minimize the mean squared error across the channel grid.

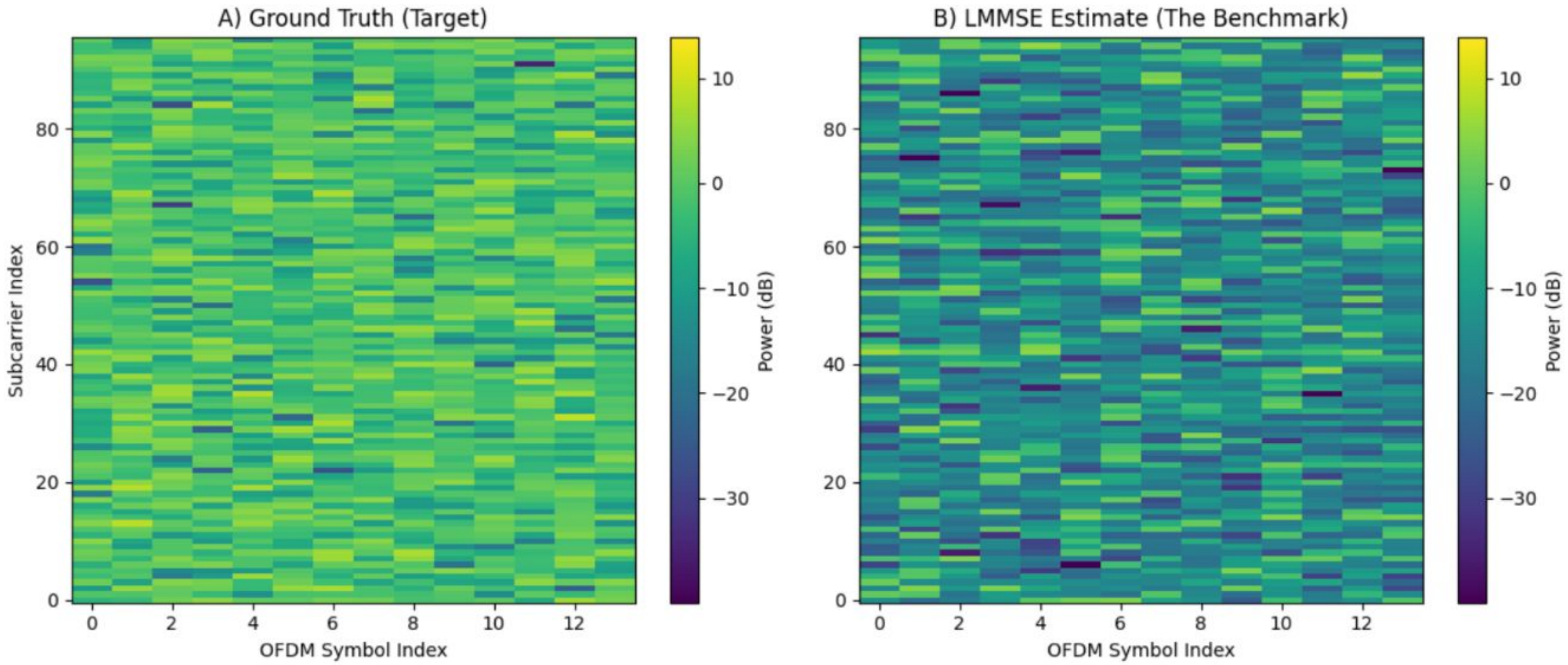
$$\text{NMSE} = 10 \cdot \log_{10} \left(\frac{\mathbb{E}[|H_{\text{True}} - \hat{H}|^2]}{\mathbb{E}[|H_{\text{True}}|^2]} \right)$$

Output:

```
--- BENCHMARK RESULTS (SNR = 10 dB) ---
1. LS (Noisy, Sparse Input) NMSE: -1.20 dB
2. LMMSE (Optimal Traditional Method) NMSE: -1.37 dB
---------------------------------------------
```

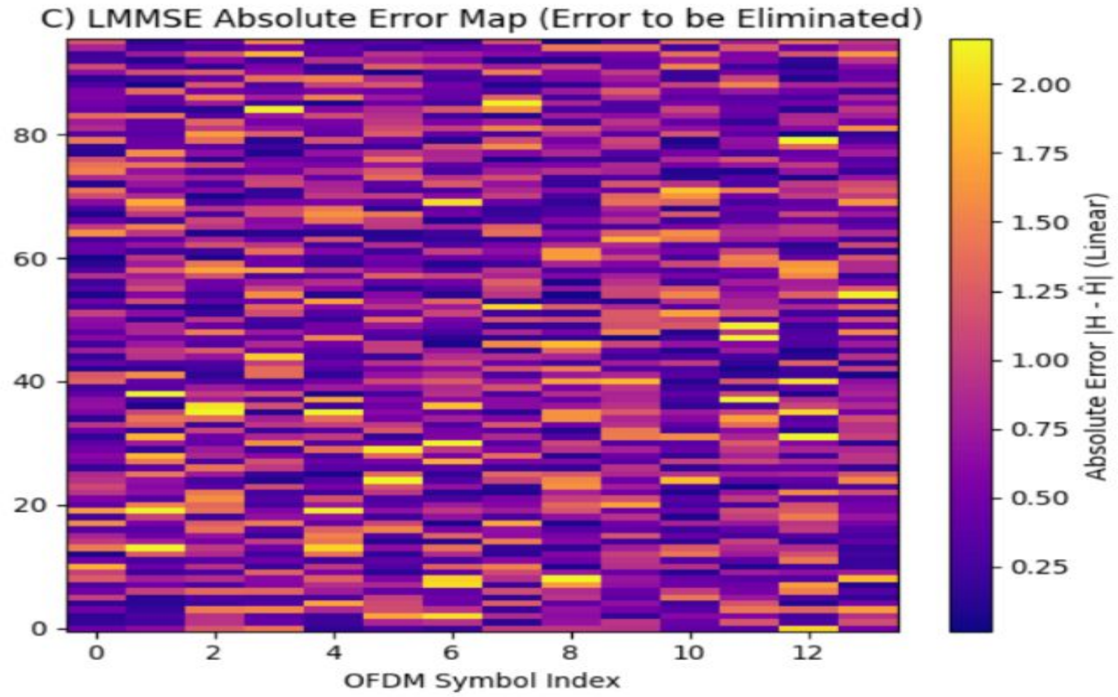
PROGRESS*[2.* Linear Minimum Mean Squared Error *]*

Channel Estimation LMMSE: -1.37dB

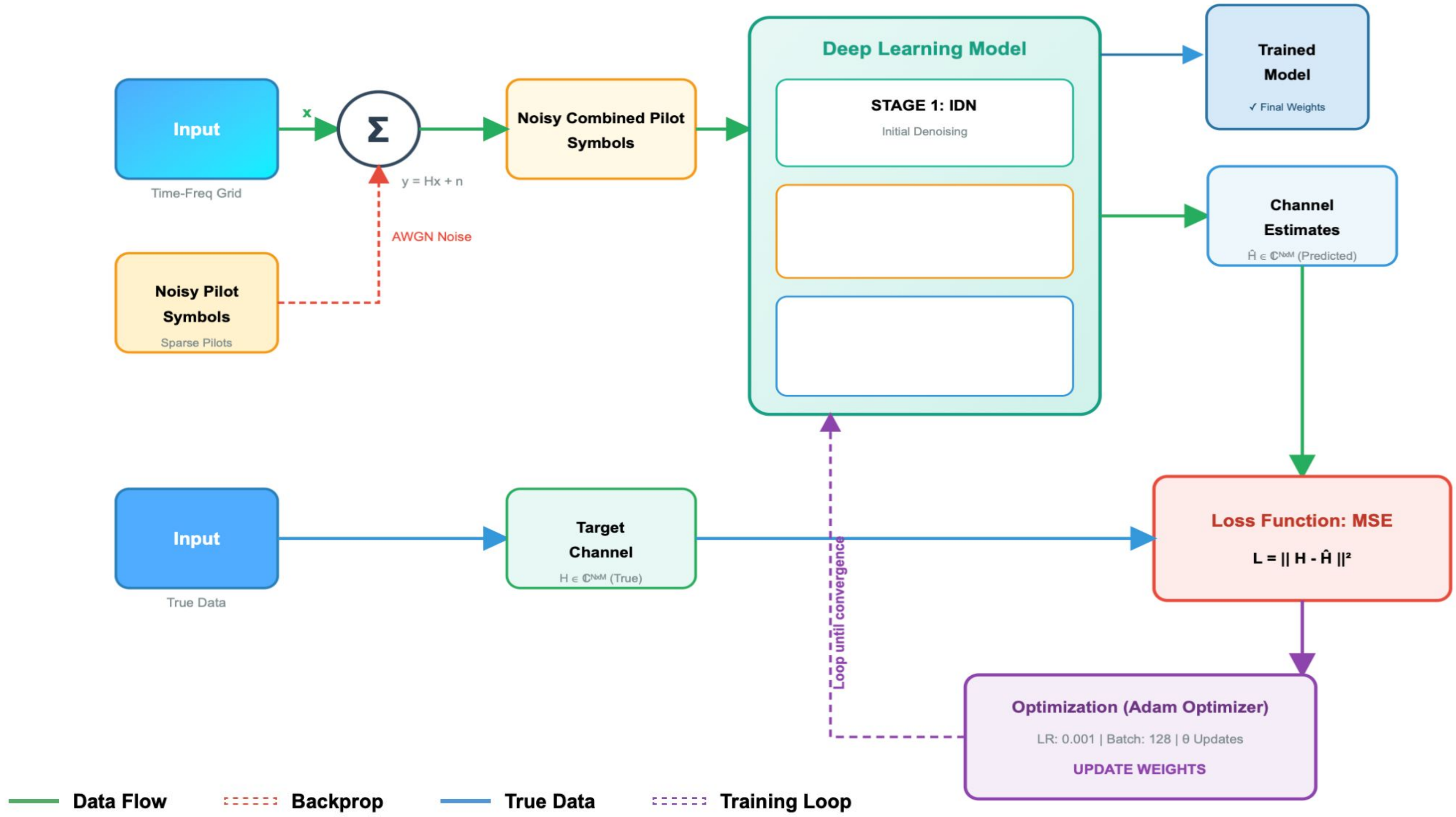


PROGRESS*[2.* Linear Minimum Mean Squared Error *]*

Channel Estimation LMMSE: -1.37dB



PROGRESS *[Initial Denoising Network]*



PROGRESS *[Initial Denoising Network]*

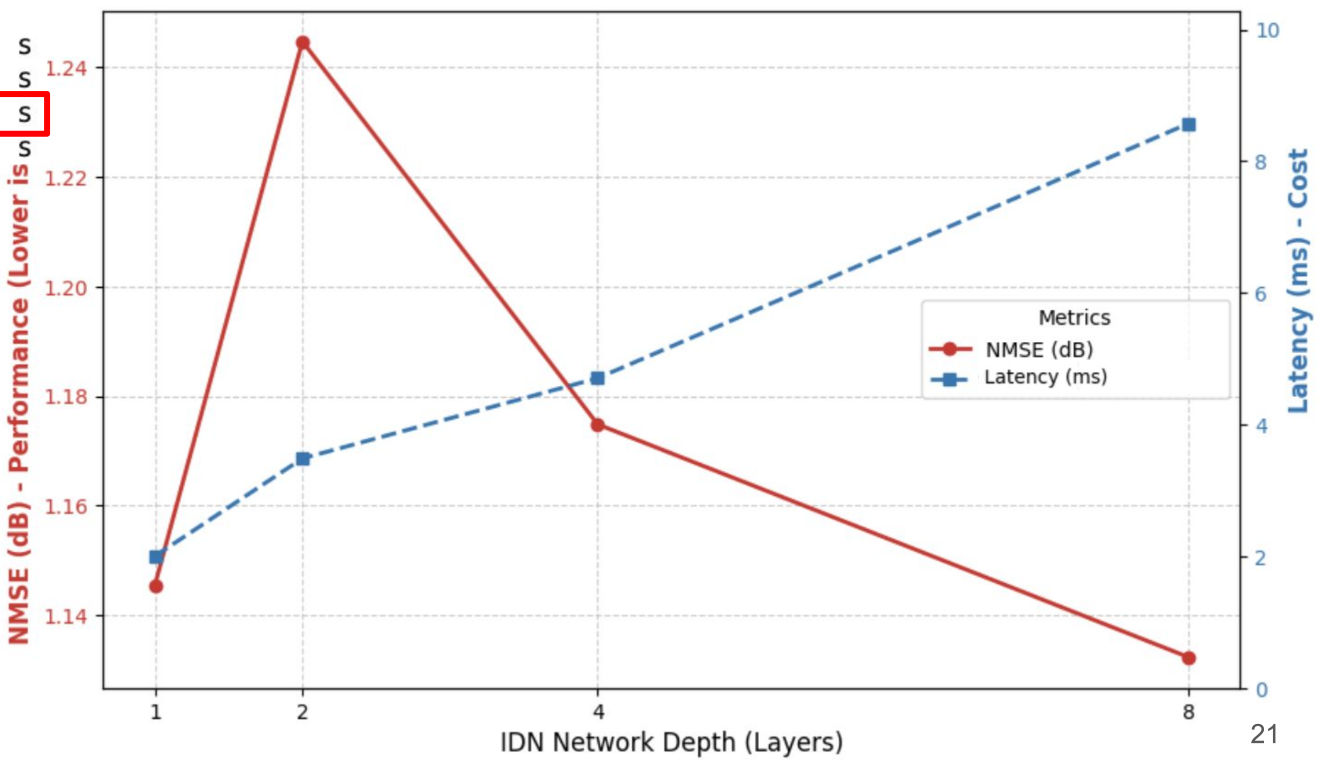
$$\text{IDN Output Formula :} \quad \hat{\mathbf{H}}_{\text{denoised}} = \mathbf{H}_{LS} - f_{\text{IDN}}(\mathbf{H}_{LS})$$

Parameters of the IDN:

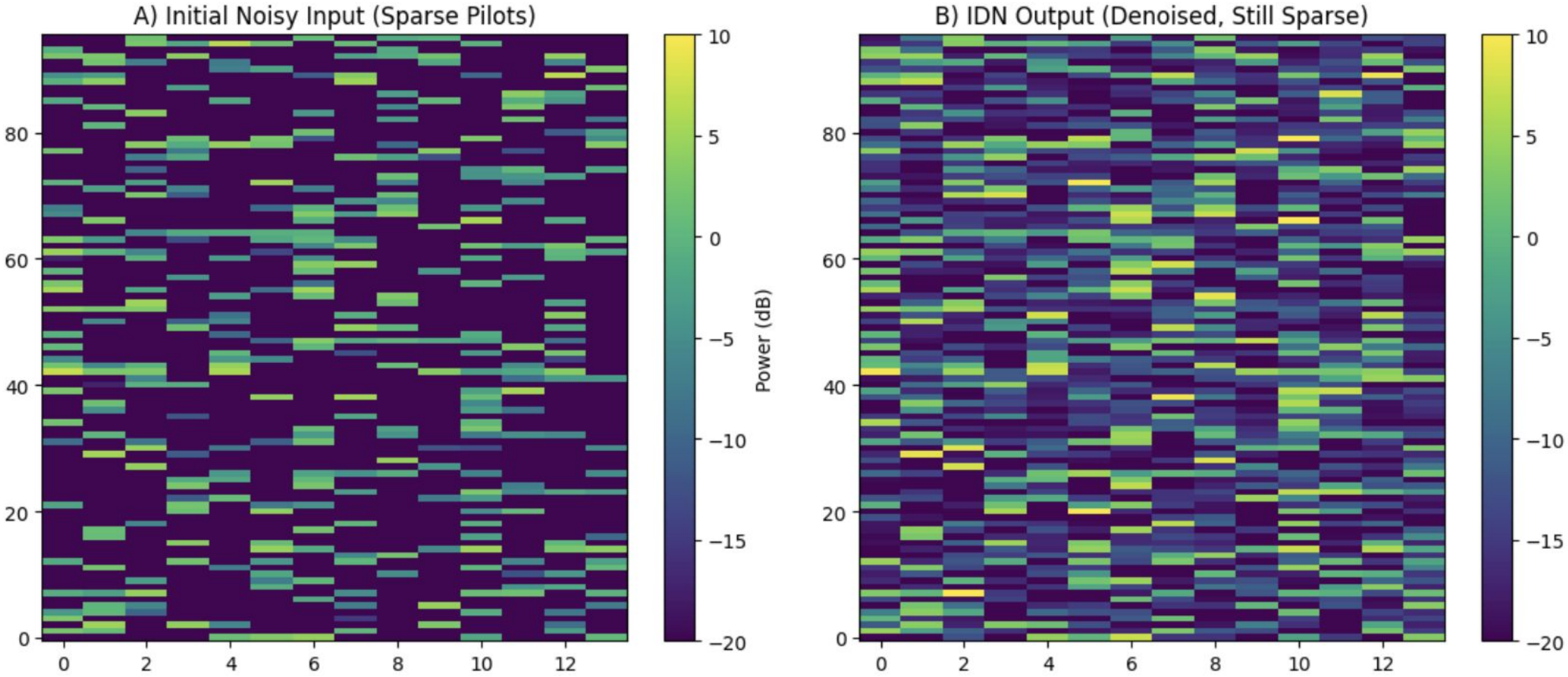
Parameter	**Value**
Layers	4 Dense Layers
Activation	LeakyReLU
Normalization	Batch-Normalisation
Output	$\hat{\mathbf{H}}_{\text{denoised}}$

PROGRESS *[Initial Denoising Network-> Layers]*

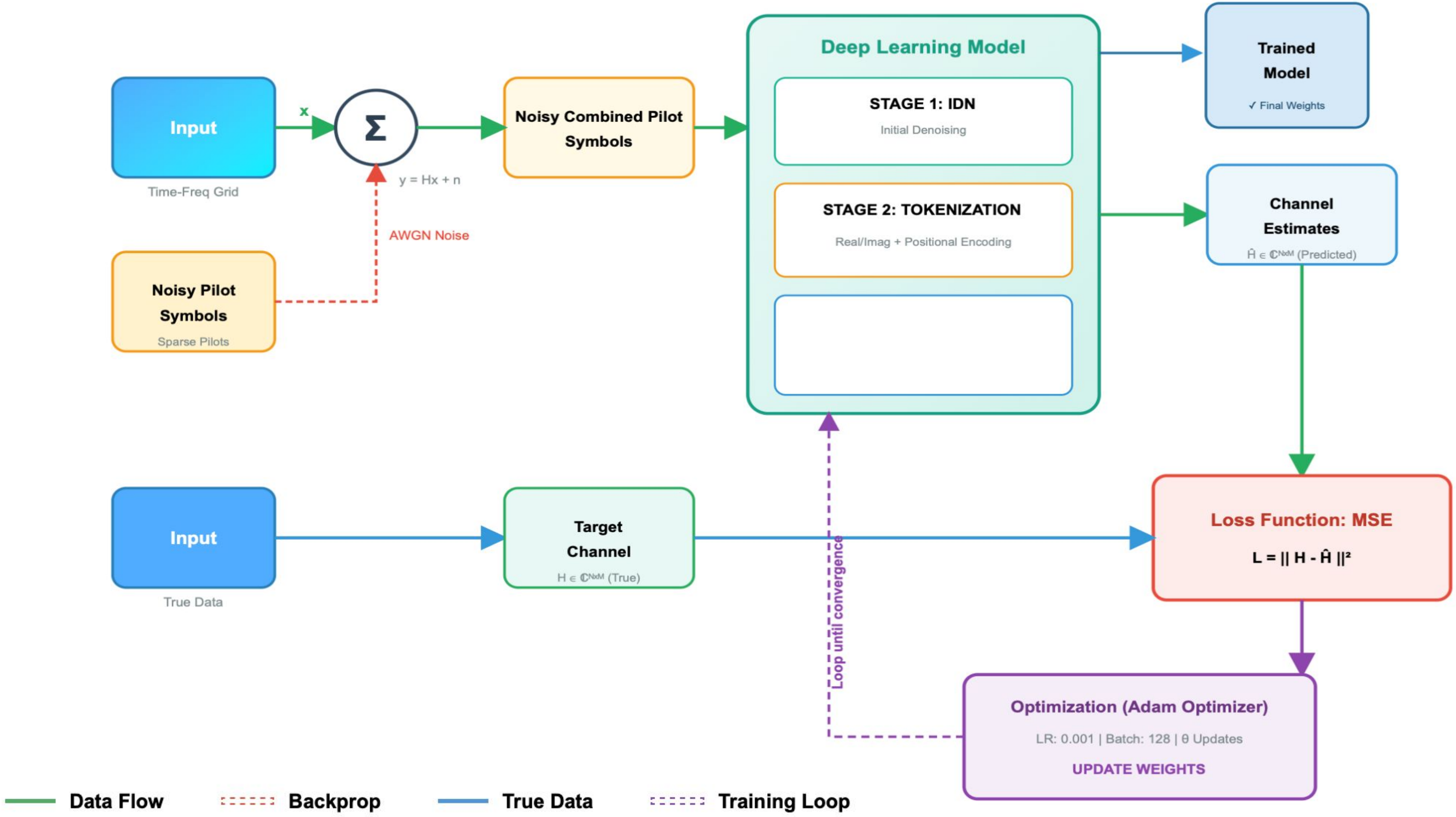
```
Final Results Table:
Layers: 1, NMSE: 1.15 dB, Time: 3.2 s
Layers: 2, NMSE: 1.24 dB, Time: 3.0 s
Layers: 4, NMSE: 1.17 dB, Time: 3.2 s
Layers: 8, NMSE: 1.13 dB, Time: 4.6 s
```



PROGRESS *[Initial Denoising Network-> Input&Output]*



PROGRESS *[TOKENIZATION]*



PROGRESS*[TOKENIZATION]*

1. Objective

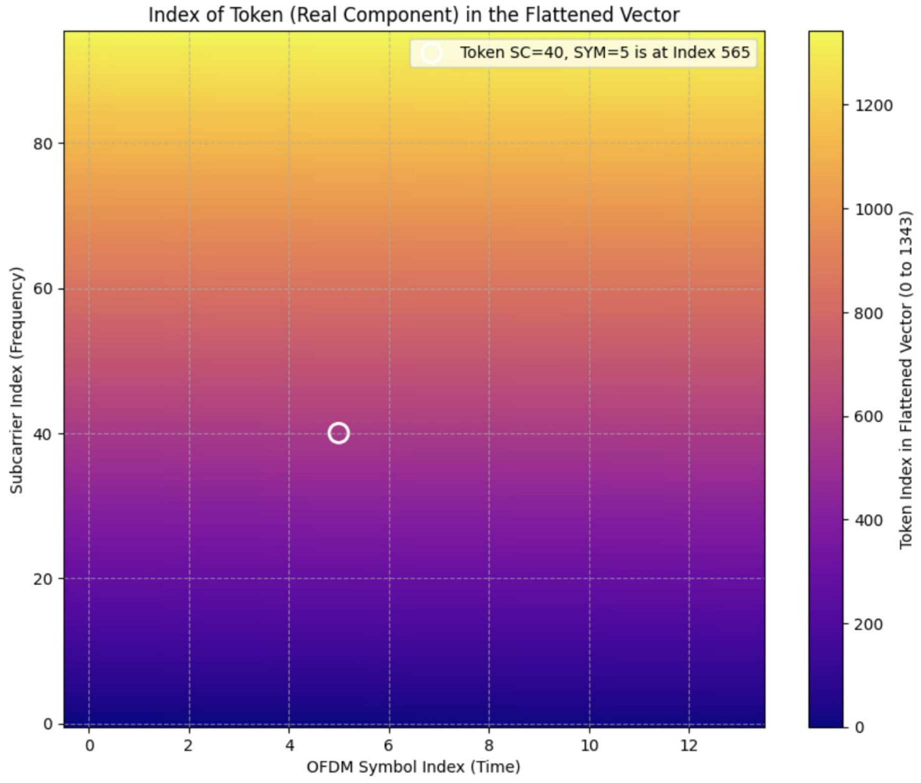
- **Problem:** The Transformer requires **1D sequential data** (like a sentence) but the Channel State Information (**H**) is a **2D grid** (Frequency × Time).
- **Goal:** Convert the 2D channel grid into a structured 1D sequence **without losing the crucial spatial coordinates**.

2. Process

Step	Concept	Description
A. Serialization	Flattening the Grid	The clean pilot data (**H**^denoised) is flattened into a single, long vector (length NSC×NSYM x2).
B. Token Creation	Real/Imag Separation	The complex values are separated into their **Real** and **Imaginary** components, forming the core features of the sequence.
C. Positional Encoding	Injecting Location	Append the original Subcarrier and Symbol index to each element. This tells the Transformer the element's exact location on the grid.

$$\text{Token}_k = [\text{Re}(\hat{H}_k), \text{Im}(\hat{H}_k), \text{Pos}_{\text{freq}}, \text{Pos}_{\text{time}}]$$

PROGRESS*[TOKENIZATION]*



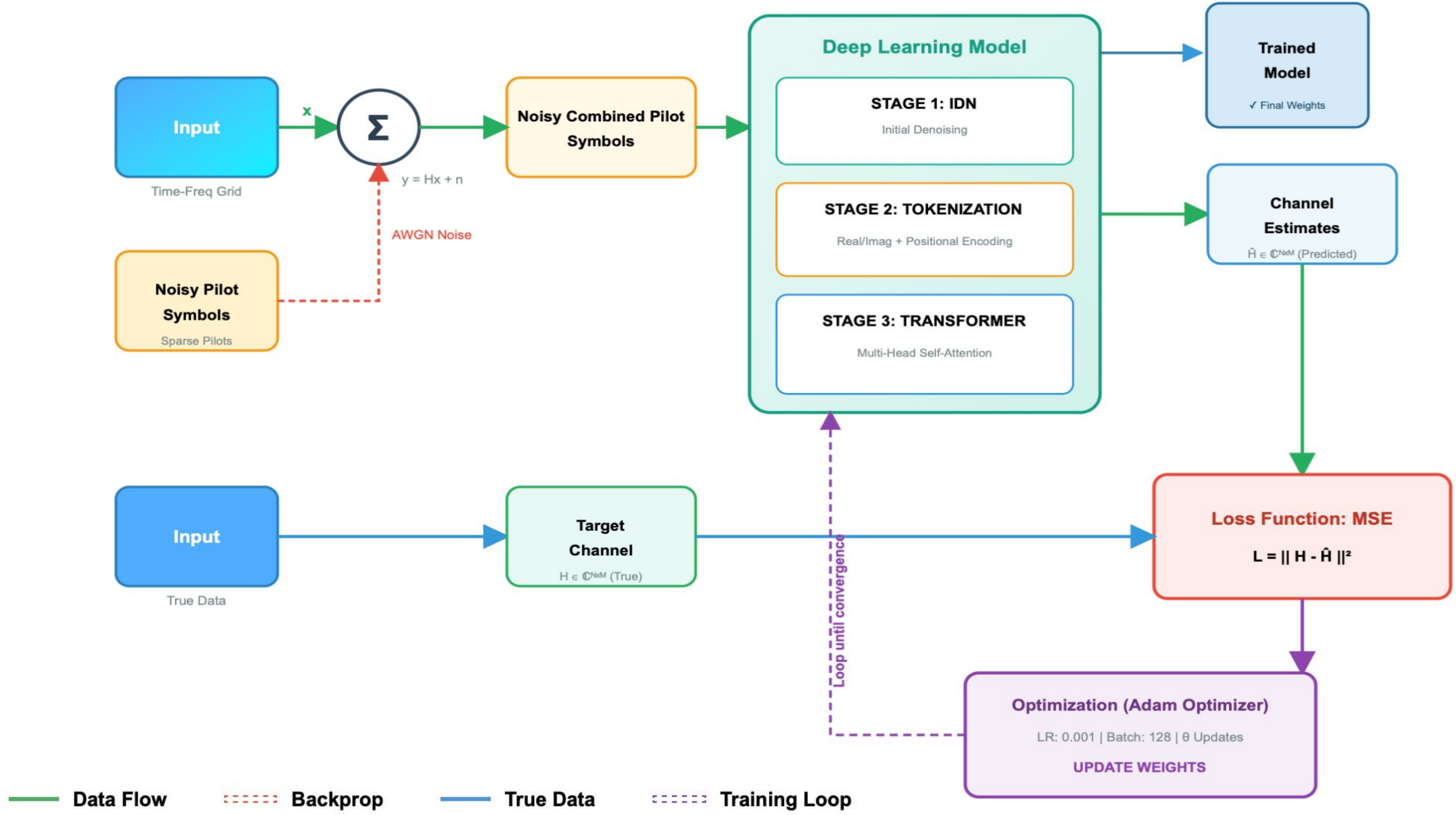
The Color Graph

Low Index (Dark Purple/Blue): These are the elements that come **first** in the sequential vector (e.g., the bottom-left corner).

High Index (Yellow/White): These are the elements that come **last** in the sequential vector (e.g., the top-right corner).

Color Gradient (Plasma/Yellow-Dark Purple): This is the value of the index in the final **flattened vector** (from 0 to 1343).

PROGRESS *[DL-model_V1]*



PROGRESS *[DL-model_V1]*

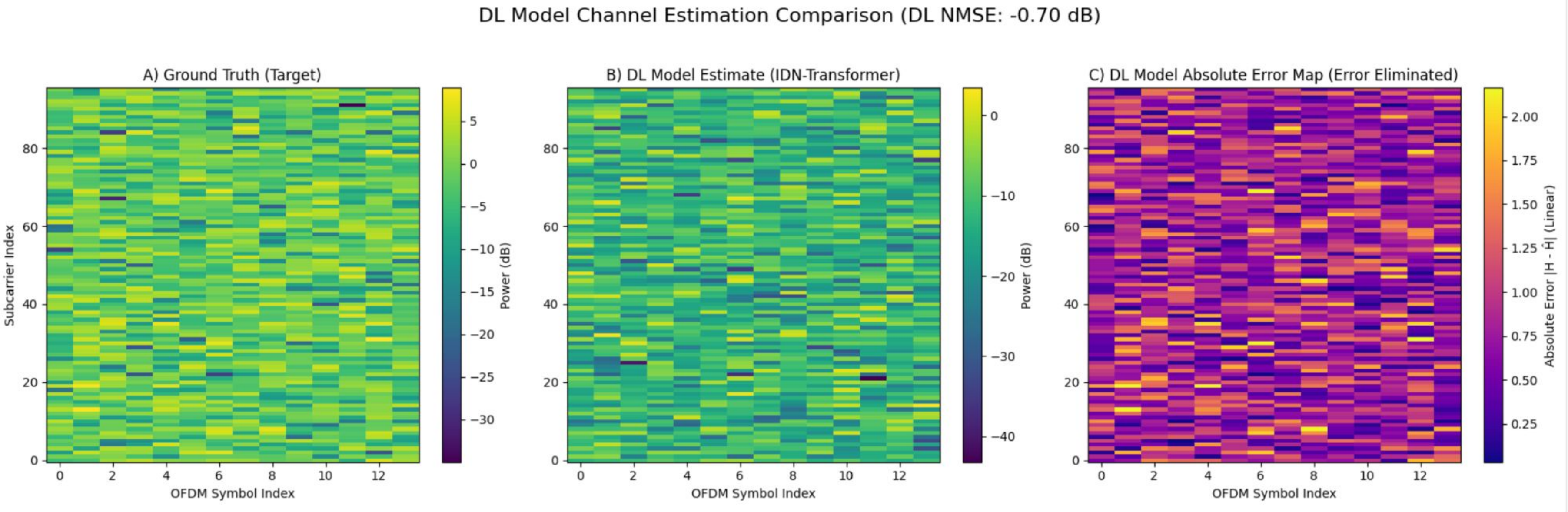
```
# --- HYPERPARAMETERS ---
BATCH_SIZE = 128
LEARNING_RATE = 0.001
N_EPOCHS = 50
```

```
--- STARTING TRAINING ---
Epoch [1/50], Loss: 0.527514
Epoch [2/50], Loss: 0.503611
Epoch [3/50], Loss: 0.495742
Epoch [4/50], Loss: 0.485545
```

```
Epoch [43/50], Loss: 0.431157
Epoch [44/50], Loss: 0.430972
Epoch [45/50], Loss: 0.430858
Epoch [46/50], Loss: 0.430710
Epoch [47/50], Loss: 0.430558
Epoch [48/50], Loss: 0.430426
Epoch [49/50], Loss: 0.430358
Epoch [50/50], Loss: 0.430177
```

```
--- FINAL PERFORMANCE RESULTS ---
1. LS (Initial) NMSE:              -1.20 dB
2. LMMSE (Benchmark) NMSE:         -1.37 dB
3. DL Model (IDN-Transformer) NMSE: -0.70 dB
```

PROGRESS *[DL-model_V1]*



PROGRESS *[DL-model_V2]*

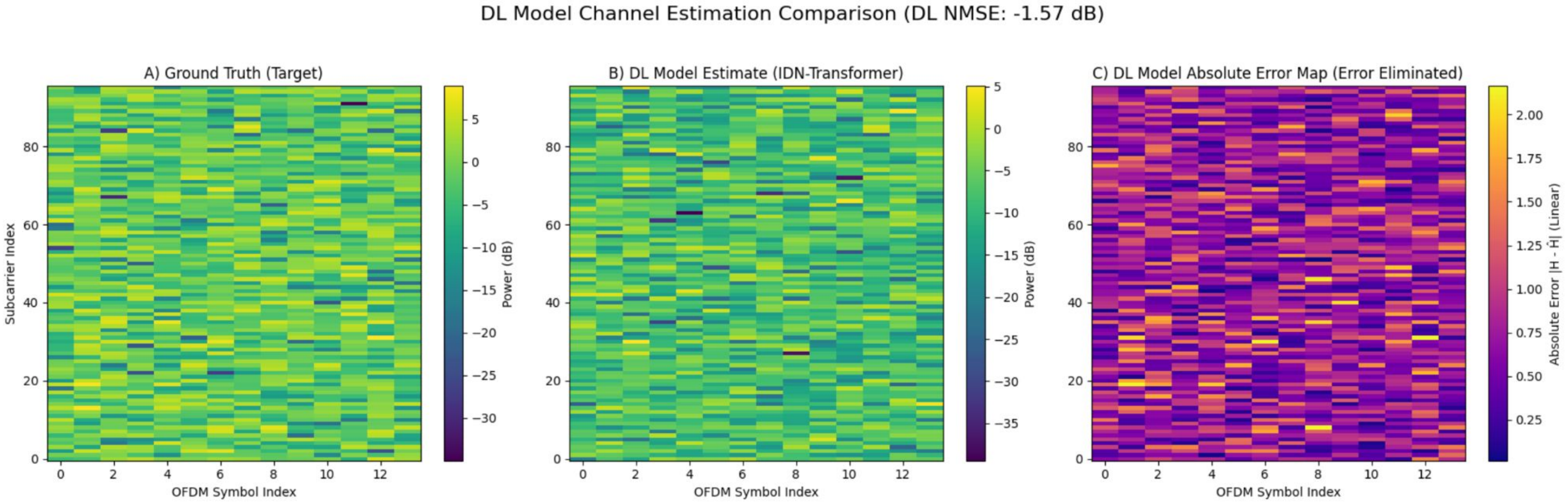
```
# --- HYPERPARAMETERS (INCREASED EPOCHS) ---
BATCH_SIZE = 128
LEARNING_RATE = 0.001
N_EPOCHS = 300
```

```
--- STARTING ENHANCED TRAINING (300 Epochs) ---
Epoch [1/300], Loss: 0.528005
Epoch [2/300], Loss: 0.508915
Epoch [3/300], Loss: 0.503364
Epoch [4/300], Loss: 0.493641
```

```
Epoch [296/300], Loss: 0.358793
Epoch [297/300], Loss: 0.358903
Epoch [298/300], Loss: 0.358767
Epoch [299/300], Loss: 0.358707
Epoch [300/300], Loss: 0.358711
```

```
--- FINAL PERFORMANCE RESULTS ---
1. LS (Initial) NMSE:              -1.20 dB
2. LMMSE (Benchmark) NMSE:         -1.37 dB
3. DL Model (IDN-Transformer) NMSE: -1.57 dB
```

PROGRESS *[DL-model_V2]*



PROGRESS *[DL-model_V2]*

Module	Component	Formula	Parameter Count
IDN	Linear 1	(2688 x 1024) + 1024	2,753,536
	BatchNorm 1	2 x1024	2,048
	Linear 2	(1024 x 512) + 512	524,800
	BatchNorm 2	2 x 512	1,024
	Linear 3	(512 x 256) + 256	131,328
	Linear 4	(256 x 2688) + 2688	690,816
Transformer	Linear Embed	(2688 x 512) + 512	1,376,768
	BatchNorm	2 x 512	1,024
	4x Encoder Layers	4 x [dmodel . (12 dmodel + 13)]	~12,600,000
	Output Layer	(512 x 2688) + 2688	1,379,456
TOTAL			**~19.46 Million**

PROGRESS *[Comparison of DL-model_V1 & model_V2]*

Metric	50 Epochs	300 Epochs
Total Parameters	19,461,472	19,461,472
Model Size (MB)	~74.2 MB	~74.2 MB
GFLOPs (Per Inference)	~0.039 GFLOPs	~0.039 GFLOPs
Total Training GFLOPs*	**~58,500 GFLOPs**	**~351,000 GFLOPs**
Training Time	8 Minutes	40Minutes
Latency	~5 ms	~5 ms

1. GFLOPs (Inference Complexity)

This is a **static** measurement of the model's "heaviness." It tells how much work the computer does to process **one single input** (one channel matrix).

2. Total Training GFLOPs (Computational Energy)

This is a **cumulative** measurement. It tells the total amount of "math" performed during the entire training process to reach the final weights.

RESULT*[till review one]*

- **Deep Learning Superiority:** The proposed **IDN-Transformer** model (NMSE: **-1.57 dB**) consistently outperforms both the **Least Squares** (NMSE: **-1.20 dB**) and the traditional optimal **LMMSE** (NMSE: **-1.37 dB**) under 10 dB SNR conditions.

- **Effective Noise Mitigation:** The **Initial Denoising Network (IDN)** effectively suppresses noise in sparse pilot symbols, providing a cleaner input that is critical for high-order modulations like **16-QAM**.

- **Global Correlation Capture:** Unlike local-focused CNNs, the **Transformer's self-attention** mechanism successfully learns complex, non-local time-frequency correlations across the entire **96 × 14 resource grid**.

- **Benchmarking Potential:** While currently outperforming mathematical baselines, there is a clear performance gap compared to deep CNNs (like **U-Net**), defining the scope for future hyperparameter optimization and extended training.

Future work remaining

1.DL model training with [IDN]->[Tokenization]->[Transformer] -For 10000 samples

2.DL model Optimization and switching 16QAM

3.Comparison with DeepLearning and Linear Minimum Mean Squared Error

S.No	ACTIVITY	DESCRIPTION	DURATION
1	Literature review and research gap identification	Study of base paper and related works to define problem statement and objectives	2 weeks
2	OFDM system model and traditional channel estimation	Understanding OFDM model and LS/LMMSE methods; finalize simulation setup	2 weeks
3	Implementation of OFDM and baseline DL model	Software implementation of OFDM system and baseline deep learning model	1 week
4	Training of proposed DL-based channel estimation model	Training and testing of denoising and Transformer-based model; performance evaluation	2 weeks
5	Performance comparison under different SNR conditions	Comparison with existing methods to validate proposed approach	1 week
6	Result analysis and documentation	Analysis of results, report writing, and presentation preparation	1 week

Tools used

- **Python** – Core programming language for OFDM simulation and deep learning implementation
- **Google Colab** – Cloud-based platform for model training and evaluation with GPU support
- **PyTorch / TensorFlow** – Deep learning framework for implementing the denoising network and Transformer model
- **NumPy & SciPy** – Numerical computation and signal processing operations
- **Matplotlib** – Visualization of performance metrics such as MSE and BER
- **Google Drive** – Dataset storage and code management
- **Dataset** – https://research.ece.ncsu.edu/ai5gchallenge/#datasets

REFERENCE

Reference 1: G. Tian, X. Cai, T. Zhou, W. Wang, and F. Tufvesson, "Deep-Learning Based Channel Estimation for OFDM Wireless Communications," *IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2022.

Reference 2: J. Guo, et al., "Deep Learning-Based Channel Estimation with Transformer," *IEEE Access*, 2021.

Reference 3: X. Yi and C. Zhong, "Deep Learning for Joint Channel Estimation and Signal Detection in OFDM Systems," *IEEE Communications Letters*, 2020.

Reference 4: Y. Zhang, J. Chen, and D. Guo, "Deep Learning-Based Channel Estimation for Massive MIMO-OFDM Systems," *IEEE Transactions on Vehicular Technology*, 2021.

Reference 5: Z. Qin, H. Ye, and G. Y. Li, "Model-Driven Deep Learning for Channel Estimation in OFDM Systems," *IEEE Transactions on Wireless Communications*, 2022.

Reference 6: J. Wu, et al., "CE-ViT: A Robust Channel Estimator Based on Vision Transformer for OFDM Systems," *IEEE Conference on Communications (ICC)*, 2024.

Reference 7: M. Gok, et al., "CHAST: Attention Aided SISO OFDM Channel Estimation," *NeurIPS Workshop on AI for Next-Generation Wireless*, 2025.

Reference 8: H. Hashempoor and W. Choi, "Comm-Transformer: A Robust Deep Learning-Based Receiver for OFDM System Under TDL Channel," *IEEE Transactions on Machine Learning in Communications and Networking*, 2024.

THANK YOU