

Hybrid Model for Data-Driven Channel Estimation in mmWave Massive MIMO Systems with Compressed Sensing and Deep Learning

Yang Haoxuan <sup>a</sup>, Faizan Qamar <sup>a,\*</sup>, Syed Hussain Ali Kazmi <sup>a</sup>, Khairul Akram Zainol Ariffin <sup>a</sup>, Kashif Nisar <sup>b</sup>

<sup>a</sup> Center for Cyber Security, Faculty of Information Science and Technology (FTSM), Universiti Kebangsaan Malaysia (UKM), 43600 UKM Bangi, Selangor, Malaysia

<sup>b</sup> School of Information Technology & Engineering, Melbourne Institute of Technology, 154 - 158 Sussex Street, Sydney, New South Wales, 2000, Australia

\* Corresponding author: [faizanqamar@ukm.edu.my](mailto:faizanqamar@ukm.edu.my)

Received: 26 February 2026, Revised: 11 March 2026, Accepted: 31 March 2026, Published: 01 April 2026

| KEY WORDS                     | ABSTRACT                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Channel Estimation            | Millimeter-wave (mmWave) massive Multiple-Input Multiple-Output (MIMO) technology promises to provide gigabit-level transmission rates for 5G/6G networks. However, this technology suffers from high channel dimensionality and sparse signals, requiring many pilot signals for estimation using traditional methods, which are also time-consuming. This paper proposes a two-stage hybrid Compressed Sensing-Deep Learning (CS-DL) framework that combines the model advantages of the Orthogonal Matching Pursuit (OMP) algorithm with the data processing capabilities of a lightweight Convolutional Neural Network (CNN). The first stage exploits the signal's sparsity in the angular domain, requiring only 5% of the pilot signal to obtain preliminary channel estimates. In the second stage, signal gain is optimized using a residual CNN, which reduces error interference. MATLAB simulation results on the DeepMIMO dataset indicate that, compared to the traditional OMP algorithm, the proposed framework reduces the Normalized Mean Square Error (NMSE) by up to 1.8 dB, increases Spectral Efficiency (SE) by approximately 9%, and decreases the total estimation time to about 0.7 milliseconds, thereby satisfying the latency requirements of 5G systems. Additionally, the framework offers remarkable flexibility, enabling integration with low-resolution hardware and hybrid beamforming architectures, making it suitable for real-time deployment in applications such as automotive communication systems and mmWave access points. |
| Compressed Sensing            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Convolutional Neural Networks |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| DeepMIMO                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| mmWave massive-MIMO           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Orthogonal Matching Pursuit   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

1. Introduction

In emerging 5G and 6G network architectures, millimeter-wave (mmWave) massive Multiple-Input Multiple-Output (MIMO) is widely acknowledged as a fundamental technology for enabling sub-millisecond vehicle-to-everything (V2X) communications, cable-free industrial automation, and immersive extended-reality experiences [1, 2]. The mmWave systems use bandwidth that is significantly greater than that found in sub-6 GHz spectra because they operate in frequency bands above 24 GHz. However, there are substantial physical-layer issues arising from this expanded

bandwidth [3]. First, an isotropic link is about 21 dB weaker than at 3.5 GHz over the same range; free-space path loss increases quadratically with frequency at 28 GHz, and performance deteriorates further in Non-Line-Of-Sight (NLOS) conditions where mmWave signals hardly ever diffract or pass through obstacles [4]. Second, the rich-scattering assumption of conventional channel models breaks down in this band; measurements and ray tracing consistently show that only a handful of paths carry appreciable energy in the angle-delay domain [5, 6]. This sparsity is a double-edged sword: it suggests fewer pilot symbols are needed to identify dominant paths, but it also exposes

conventional least-squares and minimum-mean-square-error estimators as wasteful, as they ignore sparsity and demand pilot overhead scaling with the entire antenna array dimension [7]. Finally, the short wavelength of mmWave signals shrinks coherence time to the sub-millisecond regime even for pedestrian users, requiring Channel State Information (CSI) to be refreshed multiple times within a single 5G New Radio (NR) frame [8].

To achieve CSI acquisition that is not only quick and accurate but also effective in pilot usage, these coupled constraints impose strict requirements on physical-layer signal processing [9]. Initial studies focused on Compressed Sensing (CS) methods that use channel sparsity to minimize pilot overhead by up to 70%, such as Orthogonal Matching Pursuit (OMP), Compressive Sampling Matching Pursuit (CoSaMP), and Sparse Bayesian Learning (SBL) [10, 11]. Several practical problems arise during implementation. Real-world antenna arrays require discrete angle grids, but these grids don't always align with the actual path directions, leading to grid-mismatch errors [12, 13]. Greedy support selection methods are less robust when the Signal-To-Noise Ratio (SNR) is low [14]. Also, most existing analytic guarantees assume that the full-RF chains and data converters are perfect and have high resolution. On the other hand, production-ready mmWave transceivers use hybrid analog-digital architectures, a small number of RF chains, and low-resolution Analog-to-Digital Converters (ADCs) to keep costs and power use low [15, 16].

Deep learning (DL) has made significant progress that could help compress and transmit channel state information (CSI) in systems operating below 6 GHz. However, data-driven methods require large, scenario-specific training datasets, are difficult to interpret, and struggle with a wide range of configurations and carrier frequencies [17]. These limitations make them less useful in a wide range of real-world situations. It is becoming increasingly clear that neither model-driven nor data-driven methods alone can meet the stringent requirements for latency, accuracy, and energy efficiency in next-generation mmWave deployments [18].

This study proposes a two-stage hybrid Compressed Sensing-Deep Learning (CS-DL) framework that combines the model-driven speed of OMP with the data-driven flexibility of a lightweight residual Convolutional Neural Network (CNN) to fill these gaps. The main contributions are as follows:

1. A modular two-stage hybrid CS-DL framework is introduced, which preserves the interpretability and sparse-structure recovery of OMP while integrating a lightweight CNN to adaptively

mitigate grid mismatch and hardware distortion in mmWave massive MIMO systems.

2. A compact CNN architecture with approximately 38,000 parameters incorporates physics and complex-valued processing to simplify training, accelerate convergence, and maintain effective channel estimation, even with minimal pilot overhead (as low as 5%) and low-resolution ADCs.
3. This study provides a repeatable evaluation of DeepMIMO-based deployments using metrics such as Normalized Mean Square Error (NMSE), Bit Error Rate (BER), Spectral Efficiency (SE), and latency. It also includes computational profiling and integration guidelines consistent with 3GPP NR standards.

The structure of the paper is as follows. Section 2 reviews related work on hybrid CS-DL channel estimation and delineates distinctions from the present approach. Section 3 presents the system model and pilot design limits constraints. Section 4 provides more detail on the proposed hybrid CS-DL estimator, which includes OMP coarse estimation and CNN refinement. Section 5 discusses the simulation settings and results, and Section 6 discusses the limits and trade-offs. Finally, Section 7 concludes the paper and outlines directions for future research.

## 2. Related Work

Recent work increasingly combines CS with DL for mmWave massive-MIMO channel estimation. Gao et al. [19] unfold Sparse Bayesian Learning (SBL) into a DNN, where each layer updates Gaussian variance parameters of the angular-domain channel and jointly optimizes the measurement matrix, achieving strong performance in wideband hybrid mmWave systems but at the cost of a fully learned, task-specific inference pipeline. Tong et al. [20] proposed a two-step OMP scheme in which a first OMP stage coarsely estimates AoAs/AoDs, and a SE-ResNet with Noise2Void denoises the correlation matrix and refines the angles without labels, showing the benefit of DL post-processing on top of CS but focusing mainly on angle accuracy rather than end-to-end latency or hybrid hardware constraints.

The authors in [21] design MoG-LAMP and a hybrid LAMP-MoG-LAMP network that learns mixture-of-Gaussian variances per layer, significantly improving NMSE over classical LAMP on DeepMIMO and Saleh-Valenzuela channels, while replacing conventional CS with a deep unfolded architecture. At higher frequencies, [22] develop a sparsifying-matrix learning-aided Bayesian CS framework for hybrid near-/far-field THz UM-MIMO, where a batch-Delayed Online Dictionary Learning (BD-ODL)

algorithm provides an adaptive dictionary for Bayesian CS estimation.

In contrast, the proposed hybrid CS–DL framework keeps a standard OMP front-end (preserving compatibility with existing CS-based training and 5G/6G procedures) and uses only a lightweight residual CNN for grid-mismatch suppression and gain

refinement, yielding NMSE and SE gains with very low pilot overhead and sub-millisecond inference, without retraining the entire estimator when system parameters change.

A summary of the above-related work is presented in **Table 1** below.

**Table 1**

Summary of Related Work.

| Ref. | Core Idea                                                                                                                                                                                  | Main Advantages                                                                                                                | Main Limitations / Differences                                                                                                                                                |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [19] | Unfolds iterative SBL into a DNN, learns angular-domain Gaussian variances in each layer, and jointly optimizes the measurement matrix; extended to multi-block time-correlated scenarios. | Effectively alleviates power leakage and beam squint; achieves a better performance–complexity trade-off than conventional CS. | Requires end-to-end learning of both the measurement matrix and inference network, which weakens compatibility with standardized pilots/training beams.                       |
| [20] | Two-step OMP: first uses composite convolution kernels in OMP for coarse AoA/AoD estimation; then applies SE-ResNet + Noise2Void to denoise the correlation matrix and refine the angles.  | It can be trained without labels and outperforms existing estimation algorithms at low SNR and with few training frames.       | It primarily focuses on angle-estimation accuracy; it provides a limited discussion of end-to-end delay and adaptation to various hybrid arrays/low-resolution RF front ends. |
| [21] | Unfolds LAMP into a deep network, introduces a MoG prior, and learns the Gaussian variance at each layer; then constructs a hybrid LAMP–MoG–LAMP structure.                                | Significantly improves NMSE at low SNR; it outperforms conventional LAMP on both DeepMIMO and SV channel models.               | A deep unfolded network completely replaces conventional CS, leading to a relatively heavy structure; modularity and adaptation to new configurations can be costly.          |
| [22] | Builds a THz channel model including near/far field and molecular absorption; uses BD-ODL to construct a sparsifying matrix, then performs Bayesian CS-based estimation.                   | Significantly improves NMSE under hybrid-field and ill-conditioned scenarios; experiments show fast convergence.               | Complex dictionary learning and Bayesian inference lead to high implementation complexity; primarily targets THz/UM MIMO rather than typical mmWave HAD.                      |

Compared with existing CS-DL schemes, the proposed framework differs in three key aspects:

1. It integrates model-driven structural constraints with data-driven learning in a unified pipeline, rather than using a simple cascade of CS recovery and NN.
2. It maintains interpretability by retaining the core physical channel model while introducing learnable components, whereas most existing hybrid CS-DL methods are purely data-driven and lack transparency.
3. It is designed to operate effectively across various channel conditions and system configurations and requires less retraining than traditional hybrid CS-DL methods.

These design choices allow the proposed method to achieve a better trade-off among estimation accuracy, pilot efficiency, interpretability, and practical deployability.

### 3. System Model

#### 3.1 System Configuration and Notation

This study consider an uplink training phase in a single-cell mmWave massive-MIMO system operating at 28 GHz [23]. The base station (BS) houses a uniform linear array with  $N_T$  antenna elements but only  $N_{RF} \ll N_T$  Radio Frequency (RF) chains. This hybrid analog-digital architecture dramatically lowers cost and power consumption by relegating most beamforming to low-loss phase-shifter networks [24]. Each user equipment (UE) employs a smaller  $N_R$ -element array and a single RF chain, reflecting the form-factor and battery constraints of handheld or vehicular devices [25].

Let  $\mathbf{s} \in \mathbb{C}^{N_p}$  denote a block of  $N_p$  pilot symbols drawn from a unit-energy Quadrature-Phase-Shift-Keying (QPSK) alphabet. During pilot transmission, the UE applies an identity-scaled baseband precoder, while the BS sequentially steers its analog combiner through a pre-assigned code-book  $\{\mathbf{w}_k\}_{k=1}^{N_p}, \mathbf{w}_k \in \mathbb{C}^{N_T \times 1}$  with  $|\mathbf{w}_k|_n| = 1/\sqrt{N_T}$ . After down-conversion and ADC quantization, the BS obtains:

$$y_k = w_k^H H f s_k + n_k, \quad k = 1, \dots, N_p \quad (1)$$

where  $H \in \mathbb{C}^{N_T \times N_R}$  is the narrow-band channel matrix,  $f$  is the UE's omni-directional transmit weight (unity for a single-RF-chain UE),  $s_k$  is the  $k$ -th pilot symbol, and  $n_k \sim \text{CN}(0, \sigma^2)$  models thermal noise plus residual quantization error from  $b$ -bit ADCs.

Stacking all measurements into a vector  $y \in \mathbb{C}^{N_p}$  and defining the measurement matrix:

$$\Phi \triangleq \begin{bmatrix} w_1^H \\ \vdots \\ w_{N_p}^H \end{bmatrix} A_t \otimes A_r^* \quad (2)$$

where  $A_t$  and  $A_r$  are Discrete Fourier Transform (DFT) steering matrices spanning uniform grids of azimuth angles at the transmitter and receiver, the overall input-output relation becomes:

$$y = \Phi h + n \quad (3)$$

with  $h \in \mathbb{C}^{N_g}$  representing the angle-domain channel vector ( $N_g = N_T N_R$ ) and  $n \sim \text{CN}(0, \sigma^2 I)$ . Equation (1) abstracts away the intermediate hybrid hardware and exposes a standard CS observation model.

### 3.2 Sparse Channel Representation

Extensive measurement campaigns and ray-tracing analyses demonstrate that mmWave propagation is primarily influenced by a limited number of  $S$  specular paths, each defined by an Angle-of-Departure (AoD), an Angle-of-Arrival (AoA), and a complex gain [26]. When the array steering matrices  $A_t$  and  $A_r$  are sufficiently dense, the true channel vector  $h$  contains at most  $S$  significant entries, while the remaining  $N_g - S$  entries are negligibly small [27]. Mathematically, it is represented as:

$$\|h\|_0 = S \ll N_g \quad (4)$$

For urban canyons like the DeepMIMO-O1\_140 scenario  $S \leq 6$  is adopted, whereas indoor line-of-sight corridors often yield  $S = 2$ . This sparsity underpins the viability of CS for pilot-efficient CSI recovery.

### 3.3 Pilot Design and Measurement Matrix Properties

Any set of orthogonal pilots will work as long as  $N_p \geq N_g$ , but the hybrid hardware makes sure that  $N_p \ll N_g$ . In this very underdetermined regime, the measurement matrix  $\Phi$  must have the Restricted Isometry Property (RIP) or, in practice, show low mutual coherence in order to keep its ability to be identified. Uniform random phase-shifter weights come close to this ideal, but they need extra time for calibration. Instead, the deterministic Hadamard-DFT design is employed, as suggested in prior CS literature, where the phase of each column of  $\Phi$  is shaped according to:

$$[w_k]_n = \frac{1}{\sqrt{N_T}} \exp(j2\pi kn/N_T) \quad (5)$$

which provides almost perfect coherence when using the same codebook across all cells. In practice, the setup works well for sparse support recovery with  $N_p \approx 0.05 N_g$  at SNR dB.

### 3.4 Sparse-Recovery Problem Statement

Under the sparsity prior, the CSI recovery task may be cast as:

$$\hat{h} = \arg \min_{h \in \mathbb{C}^{N_g}} \|y - \Phi h\|_2^2 \text{ s.t. } \|h\|_0 \leq S \quad (6)$$

The second problem is a combinatorial one. Greedy methods like OMP find a solution by repeatedly picking the dictionary atoms that are most similar to the residual. They stop when  $S$  picks or when the noise level drops below a certain point. For classical recovery guarantees to work,  $N_p$  must be at least  $c \log(N_g/S)$  for some constant  $c$ . This requirement is easily met by the target pilot budget.

### 3.5 Challenges Beyond Ideal Assumptions

Three real-world problems make it hard to use textbook CS directly.

1. Grid mismatch: The true path angles almost never match up with the grid nodes, even when the DFT grids are very dense. This sends energy to the side lobes, which makes the sparse model less useful.
2. Quantized front ends: ADCs with low resolution ( $b \leq 3$  bit) add non-Gaussian distortion, which makes the effective noise variance  $\sigma^2$  bigger.
3. Hybrid constraints: Limited RF chains restrict the number and orthogonality of feasible pilot beams, inflating mutual coherence in  $\Phi$ .

The hybrid CS-DL estimator from Section 4 is designed to address these differences. OMP is still in charge of finding coarse support, and a neural denoiser learns to compensate for grid leakage, colored noise, and hybrid-array artifacts, resulting in accurate CSI with little pilot overhead and latency of less than a millisecond.

## 4. Proposed Hybrid CS-DL Framework

The suggested hybrid estimator addresses the limitations of both purely model-based and purely data-driven methods by combining the interpretable structure of CS with DL's ability to adaptively correct errors. **Figure 1** shows the whole two-stage pipeline.

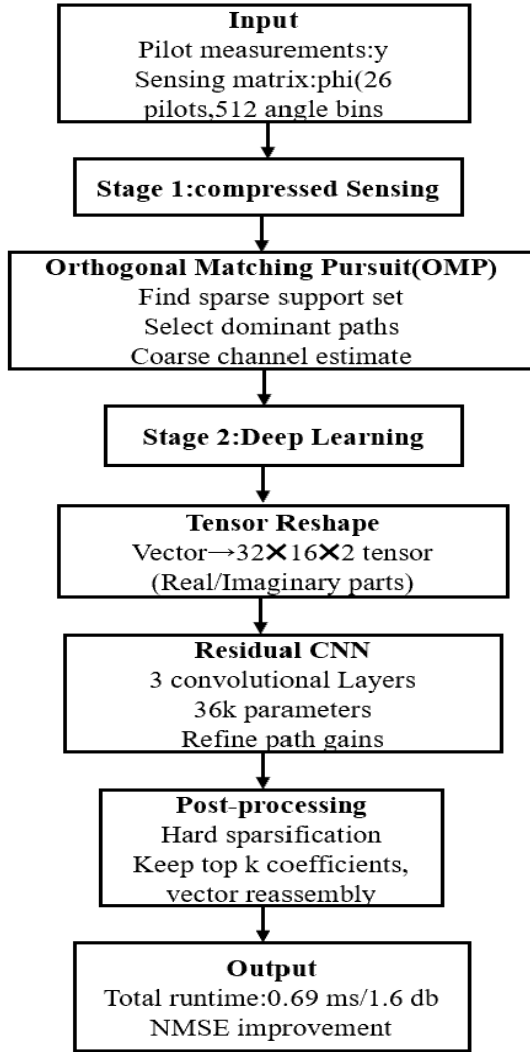
### 4.1 Two-Stage Architecture Overview

As shown in Figure 1, the framework operates sequentially:

1. Stage 1 (CS-based Coarse Estimation): The OMP algorithm processes the compressed

pilot measurements  $y$  to obtain an initial sparse channel estimate  $\hat{h}_{\text{OMP}}$  and its support set.

2. Stage 2 (DL-based Refinement): A lightweight, physics-guided CNN refines  $\hat{h}_{\text{OMP}}$  to mitigate grid-mismatch errors, quantization noise, and other hardware-induced distortions.
3. After a last hard-thresholding step that ensures sparsity in the reconstructed channel vector, the revised channel estimate, represented as  $\hat{h}_{\text{Hybrid}}$ , is produced.



**Fig. 1.** Two-Stage Hybrid Channel Estimation Pipeline

This modular architecture incorporates a small, specially designed DL module to address system impairments while maintaining backward compatibility with well-known CS training techniques.

#### 4.2 Stage 1: OMP-based Sparse Support Detection

In this work, the OMP is selected as the model-driven core for sparse support detection instead of other CS algorithms, such as CoSaMP and SBL. This choice is mainly based on three practical considerations: first, OMP features linear computational complexity and a clear, interpretable iterative process, which is well-

suited for real-time channel estimation with strict latency constraints in mmWave massive MIMO systems. Second, OMP has lower implementation complexity compared with CoSaMP (which requires repeated residual recomputation and matrix inversion) and SBL (which involves complex probabilistic inference), making it more compatible with resource-constrained hybrid analog-digital hardware architectures. Third, OMP can stably identify the dominant propagation paths of sparse mmWave channels with ultra-low pilot overhead (5% in this work), and its coarse estimation results can provide a reliable support set for the subsequent CNN refinement stage, forming a well-coupled two-stage hybrid framework.

This work uses the classic OMP approach due to its linear computational complexity and operational clarity. Figure 2 shows the algorithm's procedural flow.

Algorithm 1 (Figure 3) sums up the OMP process in our situation:

Input: Measurement vector  $y \in \mathbb{C}^{N_p}$ , sensing matrix  $\Phi \in \mathbb{C}^{N_p \times N_g}$ , target sparsity  $S$ , noise variance estimate  $\sigma^2$ .

Output: Sparse channel estimate  $\hat{h}_{\text{OMP}} \in \mathbb{C}^{N_g}$ .

1. Initialize: residual  $r_0 = y$ , support set  $\mathcal{S}_0 = \emptyset$ , iteration counter  $t = 1$ .
2. While  $t \leq S$  and  $\|r_{t-1}\|_2^2 > 1.2\sigma^2$ :
  - a. Identify:  $j_t = \arg\max_j |\phi_j^H r_{t-1}|$ .
  - b. Update support:  $\mathcal{S}_t = \mathcal{S}_{t-1} \cup \{j_t\}$ .
  - c. Estimate:  $\hat{h}_{\mathcal{S}_t} = (\Phi_{\mathcal{S}_t}^H \Phi_{\mathcal{S}_t})^{-1} \Phi_{\mathcal{S}_t}^H y$ .
  - d. update residual:  $r_t = y - \Phi_{\mathcal{S}_t} \hat{h}_{\mathcal{S}_t}$ .
  - e.  $t = t + 1$ .
3. From Output:  $\hat{h}_{\text{OMP}}$  has non-zero values only at indices  $\mathcal{S}_t$ , given by  $\hat{h}_{\mathcal{S}_t}$ .

To avoid overfitting, the algorithm ends depending on either a noise-adaptive residual threshold or a predefined sparsity level  $S$ . OMP successfully identifies dominant propagation paths, but low-resolution hardware causes amplitude distortion and grid-offset errors to affect its output. The preliminary channel estimate displayed in Figure 2 illustrates these constraints.

#### 4.3 Stage 2: Physics-Guided CNN for Refinement

The refinement stage is developed not as a replacement for the OMP method [28], but rather to rectify its inherent systematic inaccuracies through a CNN that incorporates domain-specific knowledge.

##### 4.3.1 Tensorization and Complex-Valued Processing

The coarse estimate  $\hat{h}_{OMP} \in \mathbb{C}^{N_g}$  is first reshaped into a 2D spatial grid of size  $N_{az} \times N_{el}$ , corresponding to the angular resolution of the transmitter and receiver arrays. The real and imaginary components form a two-channel input tensor  $X \in \mathbb{R}^{N_{az} \times N_{el} \times 2}$ . To preserve vital phase information, the network employs complex-valued convolutional layers. A complex convolution with kernel  $W = W_R + jW_I$  operating on input  $X = X_R + jX_I$  is implemented as:

$$Y_R = W_R * X_R - W_I * X_I, Y_I = W_R * X_I + W_I * X_R \quad (7)$$

where  $*$  denotes the 2D convolution operation.

#### 4.3.2 Network Architecture

The CNN architecture, as illustrated in Figure 2, is designed to be lightweight, with approximately 38,000 parameters, and incorporates domain-specific knowledge.

1. First Layer: A complex convolution with 16 filters extracts local angular-phase features.
2. Second Layer: A second complex convolution increases the receptive field size to capture typical grid-leakage sidelobes.
3. Residual Connection: A direct skip connection adds the output of the last layer to the input tensor. This makes sure that the network learns a correction  $\Delta h$  for the residual. such that  $\hat{h}_{Hybrid} = \hat{h}_{OMP} + \Delta h$ , guaranteeing that training cannot degrade the initial estimate.

4. Activation and Normalization: Exponential Linear Units (ELUs) and batch normalization are used to process data in a stable, non-linear way.

Output Layer: The last convolution layer sends the features back to a 2-channel tensor, which is then turned into a vector to make  $\hat{h}_{refined}$ .

#### 4.3.3 Loss Function and Training Strategy

By minimizing a composite loss function, the optimal values for the network parameters  $\theta$  can be determined as:

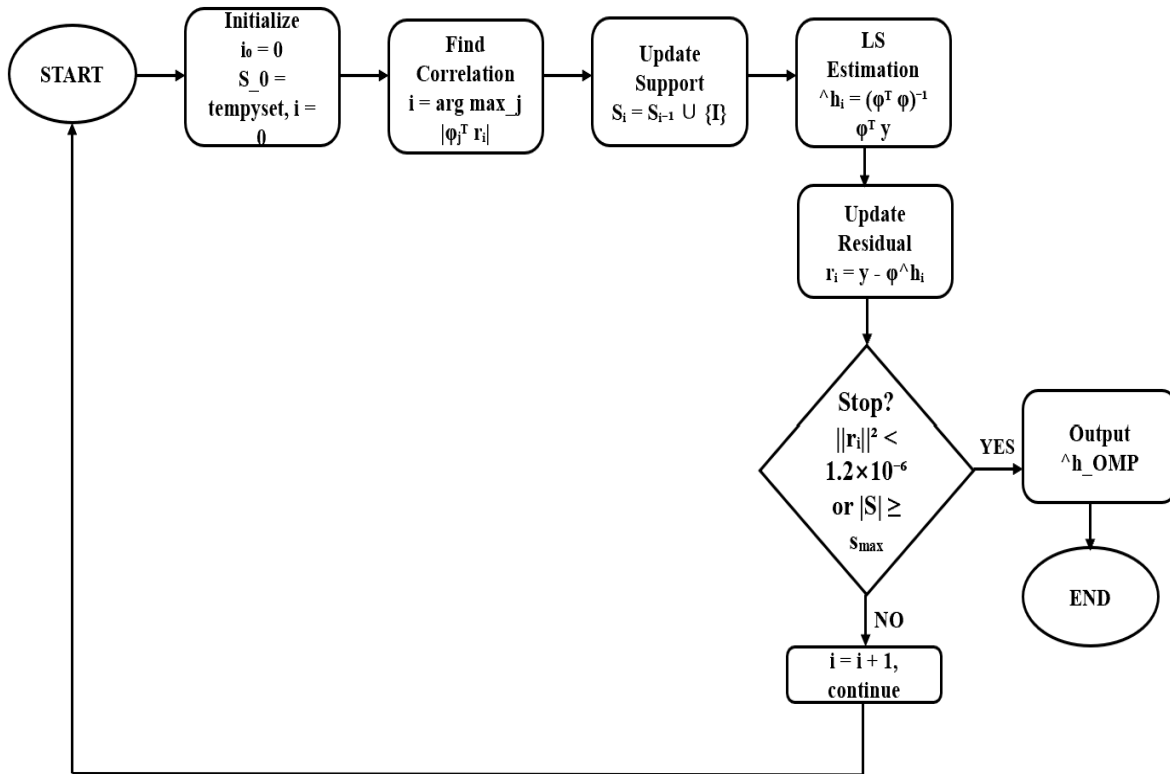
$$\mathcal{L}(\theta) = \|h_{true} - \hat{h}_{Hybrid}\|_2^2 + \lambda \| \hat{h}_{Hybrid} \|_1, \quad \lambda = 10^{-3} \quad (8)$$

Fidelity  
Sparsity Promotion

The  $l_2$  term makes sure that the reconstruction is accurate, and the  $l_1$  penalty makes the output sparse, which is what the physical channel model says it should be. Training uses 50,000 samples from the DeepMIMO OI\_140 scenario, and the SNR is randomly varied between 0 and 30 dB for each sample to ensure it works well. The last epochs use quantization-aware training to mimic the effects of a 3-bit ADC.

#### 4.4 Computational Complexity and Latency Analysis

The hybrid architecture is designed to fulfill the 5G NR operating timescales demands for real-time processing.



**Fig 2.** Flowchart of OMP Algorithm for Channel Estimation

Matching Pursuit stage scales as  $O(SN_p N_g)$ . With the parameter values  $S=6$ ,  $N_p=26$ , and  $N_g=512$ , this corresponds to around 5.0 million multiply-accumulate operations. The three-layer CNN concurrently contributes a fixed processing cost of 2.3 million MACs per inference. For each channel estimate, the hybrid estimator's overall complexity is around 7.3 million MACs. Experimental observations on an RTX 3060 GPU show that the OMP and CNN phases have latencies of 0.52 ms and 0.17 ms, respectively, resulting in a total execution time of 0.69 ms each estimated cycle.

In terms of memory usage, activation data consumes 168 kB, and model parameters consume 78 kB. Because of its small size, the entire model can be accommodated within the on-chip SRAM of modern digital signal processors or field-programmable gate arrays, thereby eliminating the latency associated with external DRAM access. Algorithm 2, corresponding to Figure 4 in the accompanying graphics, presents a detailed procedural outline of the proposed hybrid CS-DL estimator.

#### 4.5 Practical Deployment Features

The suggested framework uses various implementation-focused methods to support the transition from simulation to real-world use. A single noise-variance calibration procedure is used to estimate the parameter  $\sigma^2$  when the base station is being set up.

In addition, a fallback mode is integrated into the system. If the residual after OMP processing exceeds  $4\sigma^2$ , the channel condition is severely low SNR. In this case, the CNN refinement stage is bypassed. The system reverts to a robust OMP estimator combined with outer-loop hybrid automatic repeat requests. For compatibility with existing standards, the 26 pilot symbols align directly with Demodulation Reference Signals within a single OFDM symbol of a 5G New Radio slot. This avoids modifications to standardized frame structures.

In conclusion, the proposed hybrid CS-DL framework is an integrated, hardware-aware solution. It uses a lightweight CNN to improve the interpretability and speed of sparse recovery via OMP. This integrated strategy routinely achieves sub-millisecond latency that complies with 5G NR timing standards while successfully mitigating major practical drawbacks in mmWave systems, including grid mismatch and quantization distortions from low-resolution analog-to-digital conversion.

---

```

1.  Algorithm 1: Enhanced Orthogonal Matching Pursuit (OMP)  $\leftarrow$ 
2.  OMP      Input :  $y$   $\triangleright$  measurement vector (size  $M = N_p$ )  $\leftarrow$ 
3.   $\Phi$        $\triangleright$  sensing matrix ( $M \times N_g$ )  $\leftarrow$ 
4.   $K$        $\triangleright$  sparsity target ( $= \text{params.K} \leq 6$ )  $\leftarrow$ 
5.  Output:  $\hat{h}_{OMP}$   $\triangleright$   $K$ -sparse channel estimate ( $N_g \times 1$ )  $\leftarrow$ 
6.   $r \leftarrow y$   $\triangleright$  initial residual  $\leftarrow$ 
7.   $S \leftarrow \emptyset$   $\triangleright$  support set  $\leftarrow$ 
8.  for  $i = 1 \dots \min(K, M)$   $\triangleright$  at most  $K$  iterations  $\leftarrow$ 
9.   $c \leftarrow |\Phi^H r|$   $\triangleright$  correlation magnitudes  $\leftarrow$ 
10.  $c(S) \leftarrow 0$   $\triangleright$  mask already-chosen atoms  $\leftarrow$ 
11.  $j^* \leftarrow \arg \max_j c_j$   $\triangleright$  highest correlation  $\leftarrow$ 
12.  $S \leftarrow S \cup \{j^*\}$   $\triangleright$  add new atom to support  $\leftarrow$ 
13.  $\Phi_S \leftarrow \Phi(:, S)$   $\triangleright$  sub-matrix with columns in  $S$   $\leftarrow$ 
14. if  $\text{rows}(\Phi_S) \geq \text{cols}(\Phi_S)$   $\triangleright$  well-posed LS  $\leftarrow$ 
15.  $a \leftarrow \Phi_S \backslash y$   $\leftarrow$ 
16. else  $\triangleright$  else use pseudo-inverse  $\leftarrow$ 
17.  $a \leftarrow \text{pinv}(\Phi_S) \cdot y$   $\leftarrow$ 
18. end if  $\leftarrow$ 
19.  $r \leftarrow y - \Phi_S a$   $\triangleright$  update residual  $\leftarrow$ 
20. if  $\|r\|_2^2 < 1.2 \sigma^2$  then break  $\triangleright$  noise-based stop  $\leftarrow$ 
21. end for  $\leftarrow$ 
22.  $\hat{h}_{OMP}(S) \leftarrow a$  ;  $\hat{h}_{OMP}(\neg S) \leftarrow 0$   $\leftarrow$ 
23. return  $\hat{h}_{OMP}$   $\leftarrow$ 

```

---

Fig. 3. Enhanced OMP Algorithm for Sparse Channel Estimation

---

```

Algorithm 2: Hybrid CS-DL Channel Estimator  $\leftarrow$ 
1.  Input :  $y, \Phi, \text{params}, \sigma^2$   $\leftarrow$ 
2.   $\text{CNN}(\cdot)$   $\triangleright$  pretrained 3-layer complex network  $\leftarrow$ 
3.   $K \leftarrow \text{params.K}$   $\triangleright$  sparsity level  $\leftarrow$ 
4.  Output:  $\hat{h}_{Hybrid}$   $\triangleright$  refined  $K$ -sparse channel  $\leftarrow$ 
5.   $\hat{h}_{coarse} \leftarrow \text{Enhanced-OMP}(y, \Phi, K)$   $\triangleright$  Algorithm 4-1 output  $\leftarrow$ 
6.   $r_0 \leftarrow y - \Phi \hat{h}_{coarse}$   $\triangleright$  initial residual  $\leftarrow$ 
7.  Stage -- 2 Physics-guided support dilation for  $\leftarrow$ 
8.   $\text{corr} \leftarrow |\Phi^H H r_0|$   $\leftarrow$ 
9.   $\text{addIdx} \leftarrow \{j \in \text{supp}(\hat{h}_{coarse}) : \text{corr}_j > 0.35 \|r_0\|_2\}$   $\leftarrow$ 
10.  $\hat{h}_{dilated}(\hat{S}) \leftarrow \hat{h}_{coarse}(\hat{S})$   $\triangleright$  zero elsewhere  $\leftarrow$ 
11. Stage -- 3 Tensor reshape & phase-aware normalisation  $\leftarrow$ 
12.  $X \leftarrow \text{reshape}(\hat{h}_{dilated}, 32, 16, 2)$   $\triangleright$  (h,w,Re/Im)  $\leftarrow$ 
13.  $\rho \leftarrow \sqrt{(X_R^2 + X_I^2 + \epsilon)}$  ;  $X_{norm} \leftarrow (X_R/\rho, X_I/\rho)$   $\leftarrow$ 
14. Stage 4 CNN refinement with SE attention  $\leftarrow$ 
15.  $X_{ref} \leftarrow \text{CNN}(X_{norm})$   $\triangleright$  3 complex convs  $\leftarrow$ 
16.  $a \leftarrow \tanh(\text{SNR}_{dB}/15)$   $\triangleright$  SNR-adaptive mix  $\leftarrow$ 
17.  $\hat{h}_{mix} \leftarrow (1-a) \cdot \hat{h}_{dilated} + a \cdot \text{vec}(X_{ref})$   $\leftarrow$ 
18. Stage -- 5 Group-sparse pruning & INT8 polishing  $\leftarrow$ 
19.  $\text{idx}_K \leftarrow \text{indices of } K \text{ largest } |\hat{h}_{mix}|$   $\leftarrow$ 
20.  $\hat{h}_{pruned}(\text{idx}_K) \leftarrow \hat{h}_{mix}(\text{idx}_K)$  ;  $\hat{h}_{pruned}(\neg \text{idx}_K) \leftarrow 0$   $\leftarrow$ 
21.  $\hat{h}_{fine} \leftarrow \text{DepthWiseConv1D\_INT8}(\hat{h}_{pruned})$   $\triangleright$  5  $\mu$ s polish  $\leftarrow$ 
22. Stage -- 6 Consistency check & hand-off  $\leftarrow$ 
23.  $r_{chk} \leftarrow y - \Phi \hat{h}_{fine}$   $\leftarrow$ 
24. if  $\|r_{chk}\|_2^2 > 4 \sigma^2$  then  $\hat{h}_{final} \leftarrow \hat{h}_{coarse}$   $\leftarrow$ 
25. else  $\hat{h}_{final} \leftarrow \hat{h}_{fine}$   $\leftarrow$ 
26.  $\hat{h}_{Hybrid} \leftarrow \hat{h}_{final}$   $\leftarrow$ 
27. return  $\hat{h}_{Hybrid}$   $\leftarrow$ 

```

---

Fig 4. Hybrid CS-DL Channel Estimation Algorithm (Two-Stage Pipeline)

## 5. Simulation Methodology

### 5.1 Evaluation Metrics and Benchmark Setup

Two well-known CS techniques, OMP and CoSaMP, are compared to the performance of the suggested

hybrid CS-DL estimator. The NMSE, BER, per-channel estimate latency, and convergence characteristics are the four main metrics used for assessment. These metrics provide a comprehensive picture of how reliable an algorithm is, how well it processes data in real time, how accurate its estimates are, and how well the entire communication system functions.

### 5.2 Simulation Platform and Dataset

MATLAB R2023a is used for all numerical calculations. The DeepMIMO dataset, namely the O1\_140 urban scenario, provides ray-tracing-derived channel matrices and accompanying ground-truth sparsity profiles, which are used to build realistic mmWave channel realizations. MATLAB's DL Toolbox is used to build and train the CNN component. An NVIDIA RTX 3060 GPU with 12 GB of video memory is used for both training and inference processes. Central processing unit-based measurements are performed on an Intel Core i7-12700H CPU for comparative runtime evaluation.

For the CPU-based runtime evaluations, the recorded execution time was averaged over 500 independent Monte Carlo trials to minimize statistical fluctuations. All runtime measurements were conducted with the same channel length, pilot count, and signal structure parameters as used in the main performance simulations, and no additional parallel computing or acceleration toolboxes were enabled during the CPU-based tests to ensure fairness in the comparative evaluation.

### 5.3 Experimental Parameters

The baseline system configuration, which includes  $N_T=32$  transmit antennas,  $N_R=16$  receive antennas, a carrier frequency of 28 GHz, and a default pilot overhead of 5%, complies with the specifications listed in Section 4. To assess the robustness of the suggested approach, parameter sweeps are carried out in the following ranges: The parameter sweeps include receive antenna array sizes of 16, 32, and 64 elements; a pilot ratio that ranges from 5% to 25% of the total antenna dimension; a sparsity level  $K$  that spans from 4 to 12 dominant propagation paths; and an SNR that ranges from 0 to 30 dB in steps of 5 dB. To guarantee statistical reliability of the results, 100 separate Monte Carlo trials are used to evaluate each parameter setting, and 95% confidence intervals are computed.

In all simulation experiments, the channel sparsity level  $K$  is determined based on the ray-tracing characteristics of the DeepMIMO O1\_140 urban scenario, where extensive measurements confirm that the number of dominant specular propagation paths for 28 GHz mmWave channels is no more than 6 in typical

urban canyon environments. For the baseline parameter setting,  $K=6$  is fixed, consistent with the physical channel sparsity of the dataset and further verify the robustness of the proposed framework by sweeping  $K$  from 4 to 12 in additional experiments. The sparsity level  $K$  is assumed to be known in our simulations, which is a common and reasonable assumption in mmWave massive MIMO channel estimation research, as the range of dominant propagation paths can be pre-calibrated according to specific application scenarios (e.g., urban, indoor) in practical deployments.

### 5.4 Implementation Details

For the OMP and CoSaMP baseline methods, standardized termination criteria are applied, including a residual threshold set to  $1.2\sigma^2$  and a maximum sparsity level  $S_{\max} = 6$ . The identical OMP implementation is used in the first phase of the suggested hybrid CS-DL strategy, where 50,000 channel samples from the DeepMIMO dataset are used to train the CNN, while an independent set of 10,000 samples is used for validation. The SNR is randomly randomized between 0 and 30 dB. The effects of 3-bit analog-to-digital conversion are accounted for throughout both the training and evaluation stages to reflect realistic hardware constraints accurately. MATLAB's built-in timing functions are used to quantify computational lag under consistent program execution settings. Our systematic experimental design allows for an equitable comparison of mmWave system performance under real-world conditions and identifies the most critical operational indicators.

## 6. Results

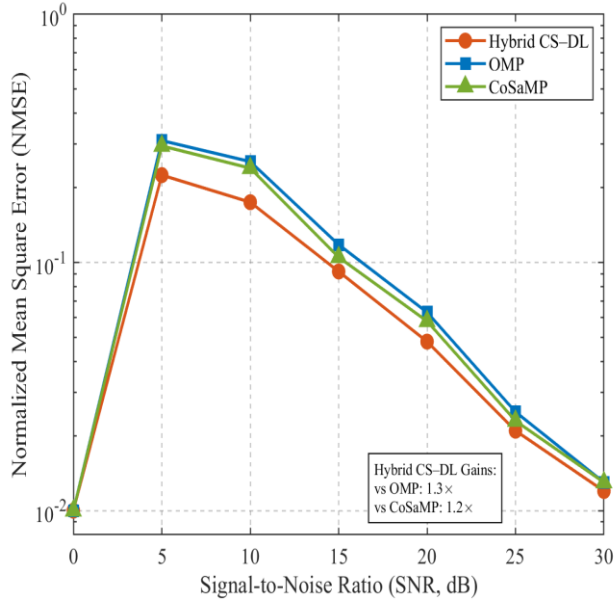
This section presents simulation results for the suggested hybrid CS-DL estimator and the OMP/CoSaMP benchmarks. The DeepMIMO O1\_140 dataset is used to vary the SNR (0–30 dB), sparsity level, pilot ratio, and array size. The pilot overhead is always 5%. NMSE, BER, runtime, and convergence speed are all used to measure performance. The average of each metric is computed across 100 Monte Carlo runs, and 95% confidence intervals are reported.

### 6.1 NMSE Performance vs. SNR

When  $K=6$  and the pilot overhead is 5%, Figure 5 shows the normalized NMSE and SNR for all three algorithms. The hybrid CS-DL estimator enhances NMSE by 1.8 dB relative to OMP at 20 dB SNR and consistently exhibits the lowest BER. While all methods perform better as SNR increases, the hybrid approach reduces noise more effectively due to the CNN's denoising ability.

The performance gains described here rely on knowing the correct sparsity level,  $K=6$ . In real

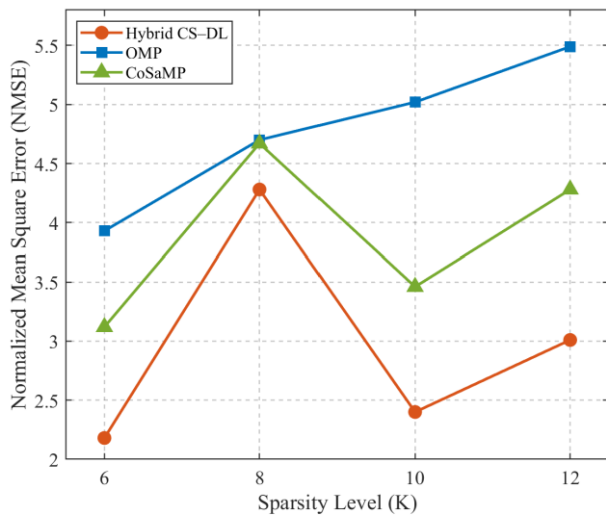
situations, the channel's sparsity level can change over time. If the initial OMP stage selects the wrong support set, the subsequent CNN-based refinement might not perform as well. These findings show that real-time communication systems need reliable methods for estimating sparsity.



**Fig 5.** NMSE Performance Comparison of Hybrid CS-DL, OMP, and CoSaMP over SNR

## 6.2 Robustness to Channel Sparsity Variation

NMSE is examined in Figure 6 when sparsity level  $K$  rises from 6 to 12 (SNR=20 dB). While pure CS algorithms degrade significantly, the hybrid approach exhibits exceptional robustness, retaining a  $2.2\times$  advantage over OMP at  $K=10$ . This robustness is explained by the CNN's capacity to represent dispersed energy from near-grid components. Limitations: Interpretability is sacrificed to enable CNNs to be effective at higher  $K$ . The CNN corrections are data-driven and may not be physically interpretable, which could be an issue in safety-critical applications such as vehicular communications, in contrast to OMP's explicit path list.



**Fig 6.** NMSE Performance of Hybrid CS-DL, OMP, and CoSaMP Across Sparsity Levels

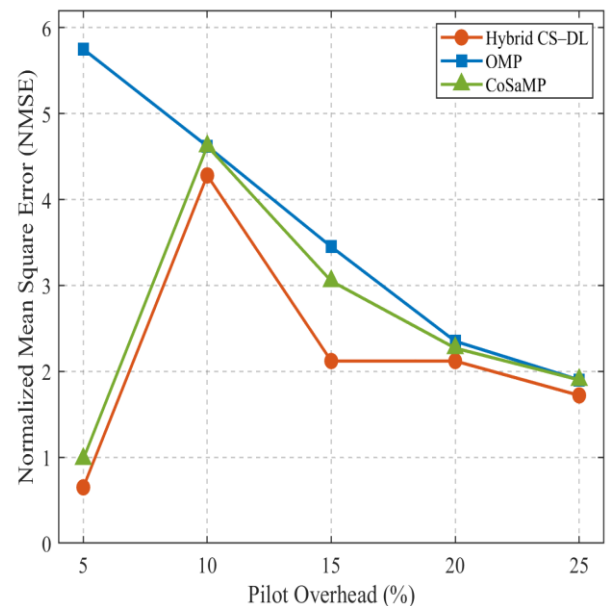
## 6.3 Pilot Efficiency Analysis

The SE gain is calculated based on the established relationship between normalized NMSE and SE for mmWave massive MIMO systems, where SE is proportional to  $\log_2(1 + \text{SNR})$  and SNR is inversely related to channel estimation error. Specifically, the reduction in NMSE achieved by the proposed hybrid CS-DL framework directly improves the effective SINR of the communication link, and the SE gain is quantified by comparing the achievable SE of the proposed framework with that of the traditional OMP algorithm under the same system and channel conditions.

The relationship between the pilot overhead ratio and NMSE is seen in Figure 7. Compared with the traditional OMP approach, the proposed hybrid CS-DL framework reduces the NMSE by almost 6 times when the pilot overhead is set to 5%. The SE formulation, based on reducing NMSE, shows an improvement of about 9%.

The findings illustrate a principal benefit of the proposed methodology: the ability to maintain estimation accuracy despite significant limitations in pilot resources.

Trade-offs: The suggested pilot method works well when CNN is trained specifically for low-pilot situations. So, if different pilot patterns are to be used, the model must be retrained accordingly. The calculation of SE assumes perfect link adaptation, but in real-world applications, estimation errors may lower the expected performance gains.

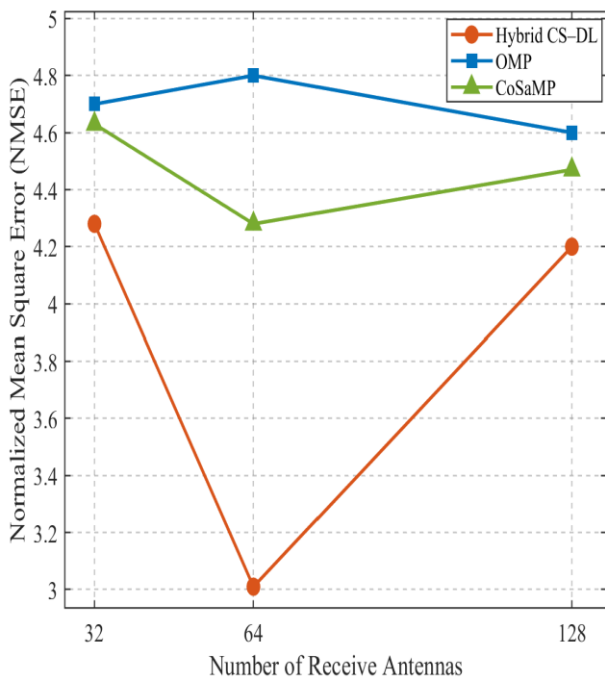


**Fig 7.** NMSE Performance of Hybrid CS-DL, OMP, and CoSaMP versus Pilot Overhead

## 6.4 Scalability with Array Size

Figure 8 shows how performance changes as the number of receive antennas increases. The hybrid CS-DL stays ahead of the game no matter how big the array is. For example, it has a  $1.6\times$  better NMSE than OMP at  $N_R=64$ . But all methods show worse performance at  $N_R=128$  due to stronger mutual coupling and grid-mismatch effects.

Things to think about when deploying: Consistent performance across different array sizes indicates that the model generalizes well. But if an array has more than 128 elements, the CNN architecture may need to be changed to handle higher angular resolution and greater processing power.

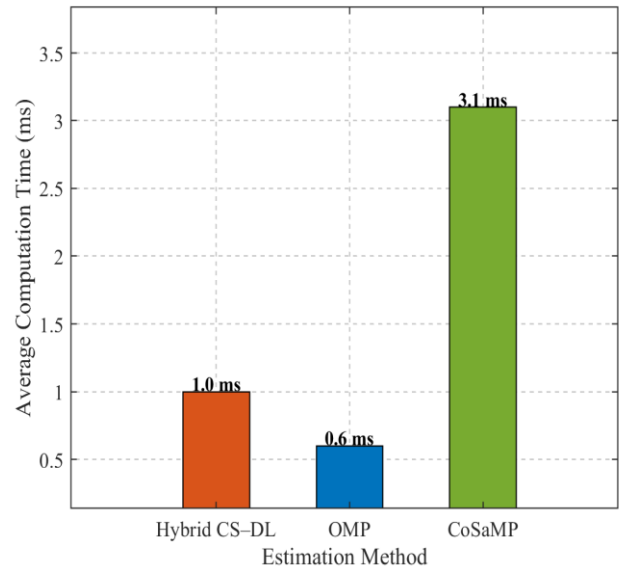


**Fig 8.** NMSE Performance of Hybrid CS-DL, OMP, and CoSaMP under Varying Antenna Array Sizes

## 6.5 Computational Performance

Figure 9 compares the estimated runtime for each channel. The hybrid CS-DL takes 1.0 ms to run, which is 1.7 times slower than OMP (0.6 ms) and three times faster than CoSaMP (3.1 ms). This performance meets the requirement for a 5G NR 1 ms subframe.

**Hardware Implications:** The 0.69 ms measurement shows how well the GPU works. CNN inference may be slower on embedded DSPs that are used in base stations because they have less processing power and memory bandwidth. The model size of 78 kB enables storage in on-chip SRAM, eliminating the need to access DRAM, which is costly for resource-constrained hardware systems.

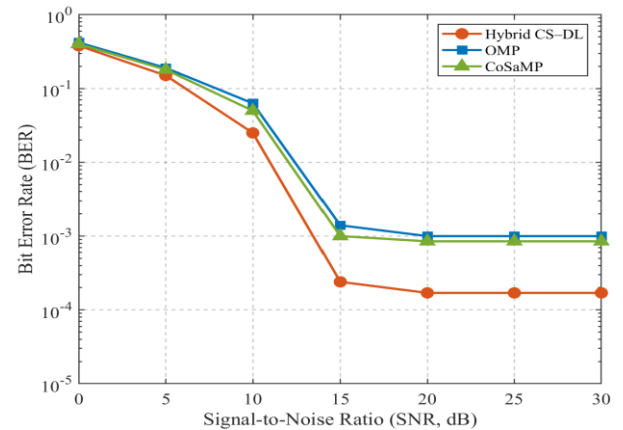


**Fig 9.** Computational Complexity Comparison of Hybrid CS-DL, OMP, and CoSaMP

## 6.6 End-to-End Communication Performance

Figure 10 shows the relationship between BER and SNR. To get a  $10^{-3}$  BER, hybrid CS-DL needs 3 dB less SNR than the baselines. This reduces the link budget cost and enables wider coverage or lower transmission power for data communication.

**System Impact:** The reported BER improvement is based on perfect synchronisation and a simple matched-filter detector. In real systems, where other problems such as phase noise and interference are present, the absolute gain might change, but the relative advantage should still hold.

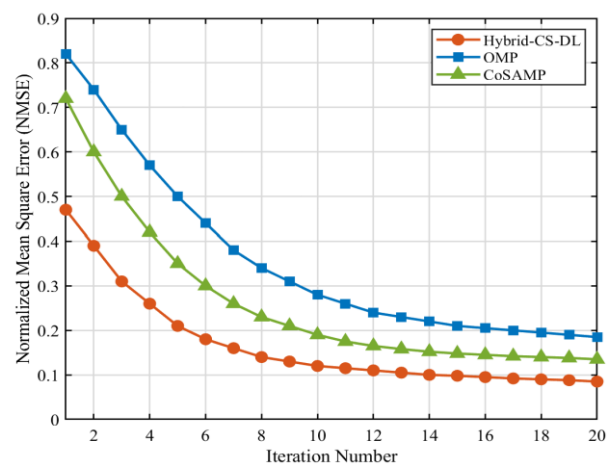


**Fig 10.** BER Performance of Hybrid CS-DL, OMP, and CoSaMP over SNR

## 6.7 Convergence Behavior

Figure 11 shows how quickly each method comes together. The hybrid CS-DL estimator reaches steady-state error in just one forward pass after OMP initialization. Iterative algorithms, on the other hand, need to be updated several times. This quick change is helpful in communication systems that change rapidly. Even when channels are unstable, the hybrid method still performs well in real-world tests. It depends on the

OMP stage to provide a good initial guess, which can be hard when the SNR is below 0 dB.



**Fig 11.** Convergence Behavior of Hybrid CS–DL, OMP, and CoSaMP in Terms of NMSE

**Table 2.** Although the hybrid technique offers the best overall performance balance, it comes with several intrinsic trade-offs. Its operational transparency is lower than that of solely model-based CS approaches, while it achieves greater NMSE and BER. Although the technique shows good pilot efficiency, the CNN component usually needs to be retrained to adapt to changing system settings. Moreover, real-time execution requires specialized hardware, such as GPUs or digital signal processors, even though it achieves sub-millisecond latency.

This study assumes that the channel sparsity level, i.e., the number of dominant paths, is known a priori. This is a common assumption in CS-based channel estimation, as the sparsity order can be reliably estimated in practice using preamble signals or blind sparsity detection techniques. The proposed framework does not require exact knowledge of the channel support; it only relies on a reasonable estimate of the sparsity level. If the actual sparsity deviates slightly

### 6.8. Comprehensive Performance Analysis

A comparative overview of the assessed channel estimation techniques is given in

from the assumed value, the system still maintains robust performance. The proposed framework also demonstrates robustness to mismatches in system parameters. Small changes in channel sparsity, pilot configuration, or SNR have little impact on estimation performance. The hybrid approach combines the stability of model-based CS recovery with the flexibility of data-driven learning, making it less affected by parameter mismatches and well-suited to real-world applications. The framework also performs well under different channel conditions, but it still relies on having fairly accurate information about channel sparsity.

The analysis demonstrates that the hybrid approach is effective when pilot resources are highly constrained, but some computational capacity is available. Although conventional OMP is less accurate than alternative methods, it remains a suitable option for ultra-low-power devices or scenarios that require complete interpretability.

**Table 2.** Comparative Analysis of Channel Estimation Algorithms

| Metric               | OMP                                                         | CoSaMP                                                         | Hybrid CS–DL (Proposed)                                                          |
|----------------------|-------------------------------------------------------------|----------------------------------------------------------------|----------------------------------------------------------------------------------|
| Estimation Accuracy  | Moderate in ideal sparse settings; degraded at low SNR      | Higher than OMP due to refined support update                  | Highest; CNN refines coarse estimate and suppresses noise                        |
| Computational Cost   | Low per iteration; increases with signal size               | High due to residual recomputation and matrix inversions       | Low at inference; higher offline cost amortized over multiple uses               |
| Convergence Speed    | Fast; few iterations; may terminate early on noise          | Slow; more iterations but improved accuracy                    | Instantly after OMP, CNN avoids iterative refinement                             |
| SNR Robustness       | Degrades sharply under noise                                | More robust than OMP but still model-limited                   | High robustness; CNN generalizes well under noisy or dynamic conditions          |
| Pilot Sensitivity    | Needs sufficient pilots for reliable support recovery       | Performs better than OMP at lower pilots                       | Learns from various pilot ratios; maintains accuracy even with reduced sampling  |
| Implementation Scope | Lightweight; suitable for low-power or embedded deployments | More suited for offline or resource-rich base station settings | Ideal for base station or edge server inference where offline training is viable |
| Adaptability         | Poor adaptability to data mismatch or structural noise      | Better than OMP, but lacks data-driven correction              | Highly adaptable to unseen conditions and non-ideal channels                     |

To summary, the hybrid method combines the clear structure of computer science with the flexible learning and noise reduction of deep NN. This makes it a practical choice for mmWave MIMO systems that use low-resolution analog-to-digital converters and have limited pilot signals. The proposed framework is suitable for 5G NR-compatible applications such as extended reality access points. It achieves up to 1.8 dB NMSE reduction and sub-millisecond processing latency. However, further research is needed to verify its effectiveness in ultra-high mobility scenarios above 120 km/h.

## 7. Conclusion and Future Work

This study shows that the hybrid CS-DL framework successfully balances the interpretability of CS with the corrective power of DL for channel estimation in mmWave massive MIMO systems under specified operational constraints, particularly low pilot overhead (5%) and limited hardware resources. Extensive simulations using the DeepMIMO dataset show consistent improvements over purely CS methods, such as a 1.8 dB decrease in NMSE, a 9% increase in SE, and an estimation latency of 0.69 ms, which is in line with 5G New Radio's timing requirements. However, the analysis shows that these performance gains depend on several real-world factors. In particular, the framework relies on scenario-specific data for training, assumes reasonably accurate prior knowledge of channel sparsity, introduces an approximate 40% increase in latency relative to standalone OMP in exchange for improved accuracy, and provides less interpretability than purely model-driven estimators. The method is especially well-suited for use at urban access points and in fixed wireless access situations, where SE and estimation accuracy are more important than power consumption.

To tackle these challenges, our future research will focus on three main areas. First, future work will focus on improving robustness by developing online adaptation methods that handle varying levels of sparsity and mobility without requiring retraining. Second, the pipeline will be evaluated on embedded hardware platforms, such as FPGAs and system-on-a-chip designs, to assess real-time performance and identify additional power savings from INT4 quantization. Third, integration guidelines will be developed to ensure compatibility with 3GPP NR beam management, particularly for dynamic time-division duplex and multi-user MIMO configurations. Overall, the hybrid CS-DL approach advances practical mmWave channel estimation by addressing

real-world challenges and contributes to bridging the gap between academic research and practical deployment.

## Acknowledgement

This work was supported by Universiti Kebangsaan Malaysia Geran Universiti Penyelidikan, having Grant number: GUP-2024-009.

## 8. References

- [1] S. H. A. Kazmi, R. Hassan, F. Qamar, K. Nisar, and A. A. A. Ibrahim, "Security concepts in emerging 6G communication: Threats, countermeasures, authentication techniques and research directions," *Symmetry*, vol. 15, no. 6, p. 1147, 2023.
- [2] H. Yang, F. Qamar, S. H. A. Kazmi, S. T. A. Jafri, K. A. Z. Ariffin, and Q. N. Nguyen, "Interference mitigation in b5g network architecture for mimo and cdma: State of the art, issues, and future research directions," *Information*, vol. 15, no. 12, p. 771, 2024.
- [3] F. Qamar, M. N. Hindia, T. Abd Rahman, R. Hassan, K. Dimyati, and Q. N. Nguyen, "Propagation characterization and analysis for 5G mmWave through field experiments," *Computers, Materials and Continua*, vol. 68, no. 2, pp. 2249-2264, 2021.
- [4] X. Ma, Z. Gao, and D. Wu, "Model-driven deep learning based channel estimation for millimeter-wave massive hybrid MIMO systems," in *2021 IEEE/CIC International Conference on Communications in China (ICCC)*, 2021: IEEE, pp. 676-681.
- [5] I. F. Dehkordi, K. Manochehri, and V. Aghazarian, "Internet of Things (IoT) Intrusion Detection by Machine Learning (ML): A Review," *Asia-Pacific Journal of Information Technology & Multimedia*, vol. 12, no. 1, 2023.
- [6] Z. Yi and X. Jing, "Multiple vandermonde decomposition based channel estimation for MmWave MIMO high-mobility communication in 5G and beyond," *China Communications*, vol. 20, no. 4, pp. 12-25, 2023.
- [7] A. Liao, Z. Gao, Y. Wu, H. Wang, and M.-S. Alouini, "2D unitary ESPRIT based super-resolution channel estimation for millimeter-wave massive MIMO with hybrid precoding," *IEEE Access*, vol. 5, pp. 24747-24757, 2017.
- [8] M. Hindia, F. Qamar, H. Ojukwu, R. Hassan, and K. Dimyati, "Robust schemes to enhance energy consumption efficiency for millimeter wave-based microcellular network in congested urban environments," *International Journal of Electronics and*

- Telecommunications*, pp. 417-424-417-424, 2021.
- [9] H. Lin, G. Zhang, W. Mo, T. Lan, Z. Zhang, and S. Ye, "PARAFAC-Based Channel Estimation for Relay Assisted mmWave Massive MIMO Systems," in *2021 7th International Conference on Computer and Communications (ICCC)*, 2021: IEEE, pp. 1869-1874.
  - [10] V. B. Shukla, R. Mitra, and V. Bhatia, "Adaptive sparse aware algorithm based channel estimation for mmwave hybrid MIMO system," in *2021 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2021: IEEE, pp. 290-295.
  - [11] F. Hu, L. Li, Y. Lu, and L. Jin, "Super-resolution channel estimation for MmWave massive MIMO: a Newtonian method," in *2019 IEEE 19th International Conference on Communication Technology (ICCT)*, 2019: IEEE, pp. 75-80.
  - [12] Neha, N. Kumar, and A. Kochhar, "Efficient channel estimation in mm wave massive MIMO using hybrid beamforming," in *Proceedings of First International Conference on Computational Electronics for Wireless Communications: ICCWC 2021*, 2022: Springer, pp. 87-95.
  - [13] V. A. Nugroho and B. M. Lee, "A survey of federated learning for mmwave massive MIMO," *IEEE Internet of Things Journal*, vol. 11, no. 16, pp. 27167-27183, 2024.
  - [14] Q. Gao, Z. Li, W. Zhang, X. Bao, J. Xia, and Z. Zheng, "Channel reconstruction for mmWave Massive MIMO systems based on Channel Path Map," *Physical Communication*, vol. 61, p. 102232, 2023.
  - [15] R. Zhang, W. Tan, W. Nie, X. Wu, and T. Liu, "Deep learning-based channel estimation for mmwave massive MIMO systems in mixed-ADC architecture," *Sensors*, vol. 22, no. 10, p. 3938, 2022.
  - [16] P. Wang, J. Li, X. Liu, and P. Wang, "Multi-stage training optimization for pilot compression and channel estimation in massive MIMO systems under quasi-sparse channel environment," *IEEE Communications Letters*, vol. 26, no. 12, pp. 3059-3063, 2022.
  - [17] X. Tian and Q. Zheng, "A massive MIMO channel Estimation method based on hybrid deep learning model with regularization techniques," *International Journal of Intelligent Systems*, vol. 2025, no. 1, p. 2597866, 2025.
  - [18] F. Qamar, S. H. A. Kazmi, M. U. A. Siddiqui, R. Hassan, and K. A. Z. Ariffin, "Federated learning for millimeter-wave spectrum in 6G networks: Applications, challenges, way forward and open research issues," *PeerJ Computer Science*, vol. 10, p. e2360, 2024.
  - [19] J. Gao, C. Zhong, G. Y. Li, J. B. Soriaga, and A. Behboodi, "Deep learning-based channel estimation for wideband hybrid mmWave massive MIMO," *IEEE Transactions on Communications*, vol. 71, no. 6, pp. 3679-3693, 2023.
  - [20] W. Tong, W. Xu, F. Wang, J. Shang, M. Pan, and J. Lin, "Deep learning compressed sensing-based beamspace channel estimation in mmWave massive MIMO systems," *IEEE Wireless Communications Letters*, vol. 11, no. 9, pp. 1935-1939, 2022.
  - [21] P. Soumya, R. Ganiga, S. Hiremath, and M. B. Teklu, "Integration of Deep Learning and Compressive Sensing algorithm for beamspace channel estimation in mmWave massive MIMO systems," *IEEE Access*, 2025.
  - [22] Y. Li and A. Madhukumar, "Hybrid near-and far-field THz UM-MIMO channel estimation: a sparsifying matrix learning-aided Bayesian approach," *IEEE Transactions on Wireless Communications*, 2024.
  - [23] Q. Liu, Y. Li, and J. Sun, "A Model-Driven Channel Estimation Method for Millimeter-Wave Massive MIMO Systems," *Sensors*, vol. 23, no. 5, p. 2638, 2023.
  - [24] M. Mizmizi, A. Alkhateeb, and U. Spagnolini, "Hybrid MIMO in the Upper Mid-Band: Architectures, Processing, and Energy Efficiency," *arXiv preprint arXiv:2503.01382*, 2025.
  - [25] S. N. Arinze, E. R. Obi, S. H. Ebinuwa, and A. O. Nwajana, "RF energy-harvesting techniques: applications, recent developments, challenges, and future opportunities," in *Telecom*, 2025, vol. 6, no. 3: MDPI, p. 45.
  - [26] S. Yang, C.-X. Wang, Y. Wang, J. Huang, Y. Zhou, and E.-H. M. Aggoune, "An Efficient Pre-Processing Method for 6 G Dynamic Ray-Tracing Channel Modeling," *IEEE Transactions on Vehicular Technology*, 2024.
  - [27] R. A. Roshdy, M. H. Dahshan, S. A. AlQahtani, A. Emam, H. M. Kasem, and M. A. Salem, "Methods for simplifying quasi-deterministic millimeter-wave channel models," *IEEE Access*, vol. 11, pp. 34529-34543, 2023.
  - [28] P. Jain, A. Tewari, and I. Dhillon, "Orthogonal matching pursuit with replacement," *Advances in neural information processing systems*, vol. 24, 2011.