A Spatio-temporal Residual Attention Network with SNR-adaptive Weighting for MIMO-OFDM Channel Estimation

Ousama El-azizi[1], Rachid Fateh[1,2], and Safi Said[1]

[1] *Sultan Moulay Slimane University, Beni Mellal, Morocco,*
[2] *Ibn Zohr University, Agadir, Morocco*

https://doi.org/10.26636/jtit.2026.3.2688

Abstract — This paper addresses channel estimation in MIMO-OFDM systems under time-varying Rician fading and spatially correlated propagation conditions, where reliable coherent detection requires accurate channel state information. Although least-squares (LS) estimation is simple and computationally efficient, it remains highly sensitive to noise. In addition, conventional deep learning-based estimators may suffer from tracking bias when temporal channel variations are not properly handled. To overcome these limitations, this paper proposes SMART-CE, a spatio-temporal residual attention network with SNR-adaptive weighting. The proposed approach refines a sequence of LS estimates by extracting spatial-frequency features, exploiting temporal channel correlation, and adaptively balancing historical and current channel information according to the operating SNR. The final channel estimate is obtained through a residual correction of the current LS estimate. Simulation results on an 8 × 8 MIMO-OFDM system show that SMART-CE generally achieves lower NMSE and BER than CNN, residual CNN, residual-attention CNN, recursive least-squares (RLS), and normalized least-mean-square (NLMS) baselines, with the most pronounced gains observed in the low-to-medium SNR regime.

Keywords — channel estimation, deep learning, MIMO-OFDM, NLMS, NMSE, residual attention, Rician fading, RLS, SNR-adaptive weighting

1. Introduction

Multiple-input multiple-output orthogonal frequency-division multiplexing (MIMO-OFDM) is a key technology for modern broadband wireless communication systems. MIMO improves spectral efficiency and link reliability by exploiting the spatial domain, whereas OFDM mitigates frequency-selective fading by dividing a wideband channel into multiple narrowband subcarriers. Owing to these advantages, MIMO-OFDM has become a suitable transmission scheme for high-data-rate and reliable wireless networks [1], [2].

The performance of MIMO-OFDM systems strongly depends on the availability of accurate channel state information (CSI) at the receiver, which is required for coherent detection, equalization, beamforming, and resource allocation. In real systems, CSI is commonly acquired through pilot-aided estimation. The least-squares (LS) estimator is widely used because of its simplicity and low computational cost. However, LS is highly sensitive to noise and does not exploit channel statistics. More advanced estimators, such as minimum mean square error (MMSE), can improve estimation accuracy, but they require prior knowledge of the channel covariance matrix and noise variance, which may be unavailable or difficult to estimate in practical time-varying environments [3], [4].

In mobile scenarios, the channel varies over time due to Doppler effects, while successive channel realizations remain temporally correlated. This correlation can be exploited to enhance estimation accuracy and mitigate channel aging effects [5], [6]. Adaptive filtering techniques, such as normalized least-mean-square (NLMS) and recursive least-squares (RLS) [7]–[9], have been proposed to track time-varying channels by recursively updating the channel estimate [10].

Although these methods can follow slow channel variations, their performance is sensitive to the step size, pilot structure, and noise level. Moreover, excessive temporal smoothing may introduce tracking bias at high signal-to-noise ratio (SNR), where the current LS estimate is already relatively accurate. Consequently, a naive use of adaptive estimates may degrade rather than improve the final estimation performance.

Recent advances in deep learning (DL) have motivated a new research direction for wireless physical-layer design. DL-based channel estimation methods have shown promising performance by learning nonlinear mappings between noisy observations and the corresponding channel responses. In particular, convolutional neural networks (CNNs) are well suited for OFDM-based systems because the channel response across antennas and subcarriers can be represented as a structured two-dimensional feature map [1], [3], [11].

Residual learning further simplifies the estimation task by enabling the network to learn a correction over a conventional estimator instead of reconstructing the entire channel directly [4], [12]. In addition, attention mechanisms improve feature representation by emphasizing the most informative spatial-frequency components [12], [14], [15].

To capture temporal dynamics, recurrent architectures and temporal learning modules have also been combined with CNN-based estimators to model inter-slot channel correlation in time-varying scenarios [16]–[18].

Despite these advances, two important limitations remain. First, many DL-based estimators rely on a single-snapshot LS estimate and therefore do not fully exploit the temporal correlation available across successive slots. Second, although hybrid model-driven and data-driven schemes have been investigated [2], [13], [19], a fixed or additive fusion strategy may be suboptimal when the reliability of temporal information changes with the operating SNR and channel dynamics.

In particular, an adaptive estimate that is useful at low SNR may introduce motion-induced bias at high SNR. This observation motivates the need for an SNR-aware temporal aggregation mechanism that can adaptively balance noise reduction and bias avoidance [20].

In this paper, we propose SMART-CE, a spatio-temporal residual attention network with SNR-adaptive temporal fusion for MIMO-OFDM channel estimation. Unlike single-snapshot estimators, SMART-CE processes a temporal sequence of LS estimates using a shared residual attention encoder to exploit inter-slot channel correlation. A lightweight SNR-aware hyper-network dynamically modulates the temporal attention weights, allowing the network to average past slots at low SNR for noise reduction while emphasizing the current slot at high SNR to mitigate motion-induced bias.

A convolutional block attention module (CBAM) is then applied to the fused spatio-temporal features to enhance informative spatial-frequency components. In parallel, an exponential moving average (EMA)-based auxiliary branch provides causally smoothed contextual information through feature-level fusion. Finally, the channel estimate is obtained as a residual correction over the current-slot LS estimate, which stabilizes learning and preserves a direct connection to the conventional baseline.

The SNR-conditioned temporal weighting mechanism is the main methodological contribution of SMART-CE. It adjusts the use of historical observations according to the operating SNR, reducing noise at low SNR while limiting tracking bias as the current observation becomes more reliable. The residual encoder, CBAM refinement, EMA context branch, and residual output connection are established components used to implement and support this temporal fusion. They are not presented as separate methodological contributions.

The main contributions of this paper are summarized as follows:

- we formulate the tradeoff between temporal noise reduction and motion-induced tracking bias, explaining why a fixed temporal averaging rule may not be suitable across different SNR regimes,
- an SNR-conditioned temporal weighting mechanism is introduced, that controls the contributions of the current and previous LS estimates according to the operating SNR,
- the SNR-adaptive temporal fusion with a shared residual encoder is integrated, CBAM-based feature refinement, and an EMA-derived auxiliary context branch,
- we compare SMART-CE with the classical NLMS and RLS adaptive estimators and with CNN, Res-CNN, and RA-CNN learning-based baselines. All methods are evaluated using the same channel realizations, pilot observations, noise conditions, and current-slot channel targets.

The remainder of this paper is organized as follows. Section 2 presents the MIMO-OFDM system model and formulates the channel estimation problem. Section 3 details the proposed SMART-CE framework, including the shared temporal residual attention encoder, the SNR-aware hyper-network, the CBAM module, the EMA auxiliary branch, and the residual output stage. Section 4 reports and discusses the numerical simulation results. Finally, Section 5 concludes the paper.

2. System Model and Problem Formulation

2.1. MIMO-OFDM System Model

Consider a MIMO-OFDM system in which a transmitter equipped with N_t antennas communicates with a receiver equipped with N_r antennas over a time-varying wireless channel. The number of active subcarriers is denoted by N_{sc}. The channel is assumed to remain approximately constant within one OFDM symbol but may vary from one time slot to another due to mobility and Doppler effects. Perfect time and frequency synchronization is assumed.

Let $\mathbf{s}_a^\kappa \in \mathbb{C}^{N_{\mathrm{sc}}\times 1}$ denote the vector of frequency-domain symbols transmitted from the a-th transmit antenna during the κ-th OFDM symbol, where $a = 1,\ldots,N_t$. The corresponding time-domain signal before cyclic prefix insertion is obtained as:

$$\mathbf{x}_a^\kappa = \mathbf{F}_{N_{\mathrm{sc}}}^H \mathbf{s}_a^\kappa \,, \tag{1}$$

where $\mathbf{F}_{N_{\mathrm{sc}}}$ is the normalized N_{sc}-point DFT matrix and $(\cdot)^H$ denotes the Hermitian transpose.

A cyclic prefix longer than the maximum delay spread is appended to preserve the circular convolution property and avoid inter-symbol interference.

After cyclic prefix removal and DFT operation at the receiver, the received signal at the n-th subcarrier and the κ-th time slot is written as:

$$\mathbf{y}^\kappa[n] = \mathbf{H}^\kappa[n]\mathbf{s}^\kappa[n] + \mathbf{w}^\kappa[n] \,, \tag{2}$$

where $\mathbf{y}^\kappa[n] \in \mathbb{C}^{N_r\times 1}$ is the received signal vector, $\mathbf{s}^\kappa[n] \in \mathbb{C}^{N_t\times 1}$ is the transmitted signal vector, and $\mathbf{H}^\kappa[n] \in \mathbb{C}^{N_r\times N_t}$ is the MIMO channel frequency response at subcarrier n. The vector $\mathbf{w}^\kappa[n] \in \mathbb{C}^{N_r\times 1}$ denotes additive complex Gaussian noise with independent entries following $\mathcal{CN}(0,\sigma_w^2)$.

The channel frequency response matrix is expressed as:

$$\mathbf{H}^\kappa[n] = \begin{bmatrix} H_{1,1}^\kappa[n] & \cdots & H_{1,N_t}^\kappa[n] \\ \vdots & \ddots & \vdots \\ H_{N_r,1}^\kappa[n] & \cdots & H_{N_r,N_t}^\kappa[n] \end{bmatrix} , \tag{3}$$

where $H_{r,a}^\kappa[n]$ denotes the channel coefficient between the a-th transmit antenna and the r-th receive antenna.

For a frequency-selective channel with L taps, this coefficient can be written as:

$$H_{r,a}^{\kappa}[n] = \sum_{\ell=0}^{L-1} h_{r,a}^{\kappa}[\ell]\, e^{-j\frac{2\pi n\ell}{N_{\mathrm{sc}}}} , \tag{4}$$

where $h_{r,a}^{\kappa}[\ell]$ is the ℓ-th channel impulse response tap.

2.2. Time-varying Rician Channel Model

To capture mobility-induced temporal variations, one of the considered propagation scenarios is modeled as a time-varying Rician channel [21], [22]. The channel matrix at slot κ is decomposed into a line-of-sight (LoS) component and a non-line-of-sight (NLoS) component as:

$$\mathbf{H}^{\kappa}[n] = \sqrt{\frac{K}{K+1}}\,\mathbf{H}_{\mathrm{LoS}}[n] + \sqrt{\frac{1}{K+1}}\,\mathbf{H}_{\mathrm{NLoS}}^{\kappa}[n] , \tag{5}$$

where K denotes the Rician factor.

The NLoS component evolves according to a first-order autoregressive process inspired by the Jakes Doppler model [21], [23]:

$$\mathbf{H}_{\mathrm{NLoS}}^{\kappa}[n] = \alpha\,\mathbf{H}_{\mathrm{NLoS}}^{\kappa-1}[n] + \sqrt{1-\alpha^2}\,\mathbf{U}^{\kappa}[n] , \tag{6}$$

where $\mathbf{U}^{\kappa}[n]$ is a complex Gaussian innovation matrix with i.i.d. $\mathcal{CN}(0,1)$ entries.

The temporal correlation coefficient is given by:

$$\alpha = J_0(2\pi f_d T_s) , \tag{7}$$

where $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind, f_d is the maximum Doppler frequency, and T_s is the slot duration.

The Doppler frequency is related to the mobile velocity v and the carrier frequency f_c by $f_d = v f_c / c$, where c is the speed of light. A larger α corresponds to stronger temporal correlation between consecutive slots, which can be exploited to improve channel estimation.

2.3. Spatially Correlated MIMO Channel Model

The second considered propagation scenario accounts for spatial correlation between antenna elements. Following the Kronecker correlation model, the channel matrix is expressed as:

$$\mathbf{H}^{\kappa}[n] = \mathbf{R}_r^{1/2}\,\mathbf{H}_{\mathrm{iid}}^{\kappa}[n]\,\mathbf{R}_t^{1/2}, \tag{8}$$

where $\mathbf{H}_{\mathrm{iid}}^{\kappa}[n]$ is an independent complex Gaussian channel matrix, while $\mathbf{R}_t$ and $\mathbf{R}_r$ denote the transmit and receive spatial correlation matrices, respectively.

In the exponential correlation model, their entries are given by:

$$[\mathbf{R}_t]_{i,j} = \rho_t^{|i-j|}, \qquad [\mathbf{R}_r]_{i,j} = \rho_r^{|i-j|} , \tag{9}$$

where ρ_t and ρ_r control the correlation levels at the transmitter and receiver sides.

This model is useful for evaluating the ability of the estimator to exploit structured spatial dependencies across antenna arrays.

2.4. Pilot-aided LS Estimation

During pilot transmission, known pilot symbols are transmitted to estimate the channel response. Let T_p denote the number of pilot symbols. The received pilot matrix at subcarrier n and slot κ is written as:

$$\mathbf{Y}_p^{\kappa}[n] = \mathbf{H}^{\kappa}[n]\,\mathbf{S}_p[n] + \mathbf{W}_p^{\kappa}[n] , \tag{10}$$

where $\mathbf{Y}_p^{\kappa}[n] \in \mathbb{C}^{N_r \times T_p}$ is the received pilot matrix, $\mathbf{S}_p[n] \in \mathbb{C}^{N_t \times T_p}$ is the known pilot matrix, and $\mathbf{W}_p^{\kappa}[n]$ is the corresponding noise matrix.

The pilot matrix is constructed from orthogonal DFT sequences satisfying:

$$\mathbf{S}_p[n]\,\mathbf{S}_p^H[n] = \mathbf{I}_{N_t} , \qquad T_p \geqslant N_t, \tag{11}$$

which limits noise enhancement in the LS estimate.

The LS estimate of the channel at slot κ and subcarrier n is obtained as:

$$\hat{\mathbf{H}}_{\mathrm{LS}}^{\kappa}[n] = \mathbf{Y}_p^{\kappa}[n]\,\mathbf{S}_p^H[n]\left(\mathbf{S}_p[n]\,\mathbf{S}_p^H[n]\right)^{-1} . \tag{12}$$

Although LS has low computational complexity and does not require channel statistics, its accuracy is strongly affected by noise, especially at low SNR.

2.5. Temporal Input Sequence

Unlike single-snapshot estimators, the proposed framework exploits a temporal observation window of T consecutive slots. The sequence of LS estimates is collected as:

$$\mathcal{S}_{\mathrm{LS}} = \left\{\hat{\mathbf{H}}_{\mathrm{LS}}^{(t)}[n]\right\}_{t=1}^{T} , \tag{13}$$

where t denotes the relative slot index within the observation window and $\hat{\mathbf{H}}_{\mathrm{LS}}^{(T)}[n]$ corresponds to the current slot to be estimated.

This sequence provides the spatio-temporal input used by SMART-CE to exploit both the current pilot observation and the correlation across previous slots.

For the auxiliary temporal branch, a causal EMA of the LS sequence is computed as:

$$\hat{\mathbf{H}}_{\mathrm{EMA}}^{(t)}[n] = \beta\,\hat{\mathbf{H}}_{\mathrm{EMA}}^{(t-1)}[n] + (1-\beta)\,\hat{\mathbf{H}}_{\mathrm{LS}}^{(t)}[n],\ t = 2,\ldots,T , \tag{14}$$

with initialization $\hat{\mathbf{H}}_{\mathrm{EMA}}^{(1)}[n] = \hat{\mathbf{H}}_{\mathrm{LS}}^{(1)}[n]$, where $\beta \in (0,1)$ is the smoothing factor.

The final smoothed estimate is denoted by:

$$\hat{\mathbf{H}}_{\mathrm{adapt}}[n] = \hat{\mathbf{H}}_{\mathrm{EMA}}^{(T)}[n] . \tag{15}$$

The EMA branch provides a causal temporal context that can reduce noise when successive channels are strongly correlated. However, because the channel may evolve across the window, excessive smoothing can introduce tracking bias. This tradeoff motivates the SNR-adaptive temporal fusion mechanism introduced in the next section.

2.6. Real-valued Tensor Representation

For the learning-based estimator, the complex-valued channel matrices are converted into real-valued tensors by stacking their real and imaginary parts. For each time slot, the LS estimate is represented as:

$$\hat{\mathcal{H}}_{\mathrm{LS}}^{(t)} \in \mathbb{R}^{2\times N_{\mathrm{sc}}\times N_r N_t}, \tag{16}$$

where the first dimension contains the real and imaginary components, the second dimension corresponds to the OFDM subcarriers, and the last dimension contains the vectorized MIMO channel coefficients.

The temporal sequence of LS estimates used by SMART-CE is then written as:

$$\mathcal{H}_{\mathrm{seq}} = \left\{\hat{\mathcal{H}}_{\mathrm{LS}}^{(t)}\right\}_{t=1}^{T} \in \mathbb{R}^{T\times 2\times N_{\mathrm{sc}}\times N_r N_t} . \tag{17}$$

The current-slot LS estimate corresponds to $\hat{\mathcal{H}}_{\mathrm{LS}}^{(T)}$. Similarly, the EMA auxiliary estimate is represented as:

$$\hat{\mathcal{H}}_{\mathrm{adapt}} \in \mathbb{R}^{2\times N_{\mathrm{sc}}\times N_r N_t} . \tag{18}$$

The true channel tensor used as the training target is denoted by $\mathcal{H} \in \mathbb{R}^{2\times N_{\mathrm{sc}}\times N_r N_t}$ and corresponds to the channel at the current slot.

2.7. Problem Formulation

The objective of channel estimation is to recover the current channel response from noisy pilot-based observations. In the proposed framework, the final estimate is formed as a residual correction over the current-slot LS estimate:

$$\hat{\mathcal{H}} = \hat{\mathcal{H}}_{\mathrm{LS}}^{(T)} + f_\theta\left(\mathcal{H}_{\mathrm{seq}}, \hat{\mathcal{H}}_{\mathrm{adapt}}, \gamma\right), \tag{19}$$

where $f_\theta(\cdot)$ denotes the SMART-CE estimator with trainable parameters θ and γ is the SNR value provided to the SNR-aware module.

The residual formulation allows the network to learn the estimation error of the current LS input rather than reconstructing the full channel from scratch. This design provides a direct connection to the conventional LS baseline, stabilizes training, and helps preserve the fine structure of the current channel observation. The detailed architecture of $f_\theta(\cdot)$ is presented in the next section.

Notation: Matrices, vectors, and scalars are denoted by boldface uppercase, boldface lowercase, and normal letters, respectively. Superscripts $(\cdot)^T$, $(\cdot)^H$, and $(\cdot)^{-1}$ denote transpose, Hermitian transpose, and matrix inverse, respectively. The operator $\mathbb{E}\{\cdot\}$ denotes expectation, $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind, and $\|\cdot\|_F$ represents the Frobenius norm.

3. Proposed Channel Estimation Technique

This section presents SMART-CE, the proposed spatio-temporal residual attention network with SNR-adaptive temporal fusion. Unlike single-snapshot estimators that rely only on the current LS estimate, SMART-CE exploits three complementary sources of information: temporal correlation from a sequence of LS estimates, spatial-frequency structure through residual attention and CBAM, and instantaneous SNR information through a lightweight hyper-network that adaptively controls temporal aggregation. The overall architecture is shown in Fig. 1.

The proposed estimator takes three inputs: the temporal LS sequence $\mathcal{H}_{\mathrm{seq}} \in \mathbb{R}^{T\times 2\times N_{\mathrm{sc}}\times N_r N_t}$, the EMA-based auxiliary estimate $\hat{\mathcal{H}}_{\mathrm{adapt}} \in \mathbb{R}^{2\times N_{\mathrm{sc}}\times N_r N_t}$, and the instantaneous SNR value γ. It outputs a residual correction $\Delta\mathcal{H}$, which is added to the current-slot LS estimate $\hat{\mathcal{H}}_{\mathrm{LS}}^{(T)}$ to obtain the final channel estimate.

3.1. Shared Residual Attention Encoder

The first component of SMART-CE is a residual attention encoder that is applied independently to each slot of the LS sequence using shared weights. Weight sharing reduces the number of trainable parameters and enforces a consistent feature extraction process across time [16].

For each slot $t = 1, \ldots, T$, the corresponding LS tensor $\hat{\mathcal{H}}_{\mathrm{LS}}^{(t)} \in \mathbb{R}^{2\times N_{\mathrm{sc}}\times N_r N_t}$ is first projected into a C-dimensional feature space as:

$$\mathbf{Z}_0^{(t)} = \phi\left(\mathcal{B}\left(\mathbf{W}_0 * \hat{\mathcal{H}}_{\mathrm{LS}}^{(t)}\right)\right) , \tag{20}$$

where $*$ denotes convolution, $\mathbf{W}_0$ is the input convolution kernel, $\mathcal{B}(\cdot)$ denotes batch normalization, and $\phi(\cdot)$ is the ReLU activation function.

The input projection uses a 3×3 convolution from 2 to 64 channels, followed by batch normalization and ReLU. The shared encoder contains six residual attention blocks. Each block uses two 3×3 convolutions with 64 input and output channels, batch normalization, ReLU after the first convolution, and a squeeze-and-excitation channel gate with a reduction ratio of 8.

The feature map is then refined by N_{RB} residual attention blocks. For the i-th block, the transformation can be written as:

$$\tilde{\mathbf{Z}}_i^{(t)} = \mathcal{B}\left(\mathbf{W}_{i,2} * \phi\left(\mathcal{B}\left(\mathbf{W}_{i,1} * \mathbf{Z}_{i-1}^{(t)}\right)\right)\right), \tag{21}$$

followed by channel-attention recalibration and residual addition:

$$\mathbf{Z}_i^{(t)} = \phi\left(\mathbf{Z}_{i-1}^{(t)} + \mathcal{A}_{\mathrm{ch}}\left(\tilde{\mathbf{Z}}_i^{(t)}\right)\right) , \tag{22}$$

where $\mathcal{A}_{\mathrm{ch}}(\cdot)$ denotes the channel-attention operation.

For an input feature map $\mathbf{X} \in \mathbb{R}^{C\times N_{\mathrm{sc}}\times N_r N_t}$, it is given by:

$$\mathcal{A}_{\mathrm{ch}}(\mathbf{X}) = \mathbf{X} \odot \sigma\left(\mathbf{W}_{e2}\phi\left(\mathbf{W}_{e1}\mathrm{GAP}(\mathbf{X})\right)\right) , \tag{23}$$

where $\mathrm{GAP}(\cdot)$ denotes global average pooling, $\mathbf{W}_{e1}$ and $\mathbf{W}_{e2}$ are learnable excitation weights, $\sigma(\cdot)$ is the sigmoid function, and $\odot$ denotes element-wise multiplication.

The residual connection improves gradient flow and enables the encoder to learn informative corrections over the input features [4], [12].

The per-slot feature map produced by the shared encoder is denoted by:

$$\mathbf{F}^{(t)} = \mathbf{Z}_{N_{\mathrm{RB}}}^{(t)} \in \mathbb{R}^{C\times N_{\mathrm{sc}}\times N_r N_t} . \tag{24}$$

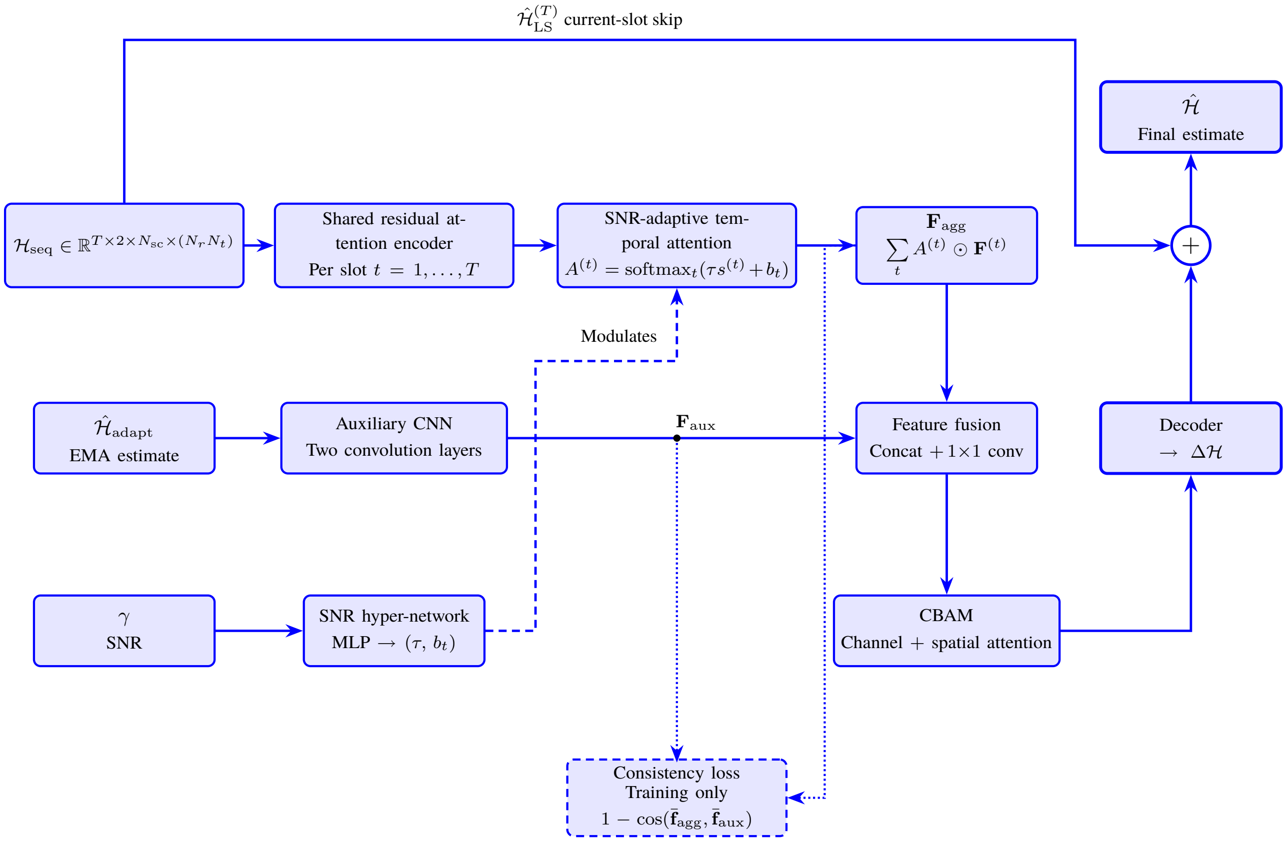

Fig. 1. Overall architecture of the proposed SMART-CE framework.

3.2. SNR-aware Temporal Attention

A key component of SMART-CE is the SNR-aware aggregation of the per-slot features. The reliability of past channel observations depends on both the noise level and the channel dynamics. At low SNR, exploiting several temporally correlated slots can reduce the impact of noise. At high SNR, however, the current LS estimate becomes more reliable and excessive reliance on past slots may introduce tracking bias due to residual channel variations.

To account for this tradeoff, a lightweight hyper-network $f_{\mathrm{hyp}}(\cdot)$ maps the normalized SNR to a temporal temperature τ and a set of per-slot bias terms $\{b_t\}_{t=1}^{T}$ [20]:

$$[\tau, b_1, \dots, b_T] = f_{\mathrm{hyp}}(\gamma/\gamma_0), \quad \tau = \mathrm{softplus}(\cdot) + \tau_0, \tag{25}$$

where γ_0 is a normalization constant and $\tau_0 > 0$ ensures a strictly positive temperature.

For the five-slot input window, the hyper-network is an MLP with layer dimensions $1 \to 32 \to 32 \to 6$ and ReLU after the first two linear layers. Its six outputs are one temperature parameter and five slot-dependent bias terms. The SNR input is expressed in dB and normalized by $\gamma_0 = 20$ dB, and the positive-temperature offset is $\tau_0 = 0.5$.

For each time slot, an element-wise temporal score is computed from the corresponding feature map using a 1×1 convolution:

$$\mathbf{s}^{(t)} = \mathbf{W}_s * \mathbf{F}^{(t)} \in \mathbb{R}^{1\times N_{\mathrm{sc}}\times N_r N_t}. \tag{26}$$

The temporal attention weights are then obtained by an SNR-modulated softmax over the temporal dimension:

$$\mathbf{A}^{(t)} = \frac{\exp\left(\tau\,\mathbf{s}^{(t)} + b_t\right)}{\sum_{t'=1}^{T} \exp\left(\tau\,\mathbf{s}^{(t')} + b_{t'}\right)}. \tag{27}$$

The aggregated spatio-temporal feature is given by:

$$\mathbf{F}_{\mathrm{agg}} = \sum_{t=1}^{T} \mathbf{A}^{(t)} \odot \mathbf{F}^{(t)}. \tag{28}$$

Through this mechanism, the network can learn to assign smoother temporal weights when noise dominates and to emphasize the current slot when the instantaneous LS observation becomes more reliable. This SNR-aware fusion allows SMART-CE to balance noise reduction and tracking-bias mitigation without using a fixed temporal averaging rule.

The SNR value provided to the hyper-network is assumed to be available from standard receiver-side noise variance estimation in pilot-aided systems.

3.3. EMA Auxiliary Branch and CBAM-based Feature Refinement

In parallel with the main temporal branch, the EMA-based auxiliary estimate $\hat{\mathcal{H}}_{\mathrm{adapt}}$ is processed by a lightweight convolutional branch $\psi(\cdot)$ to produce a complementary contextual feature:

$$\mathbf{F}_{\text{aux}} = \psi\left(\hat{\mathcal{H}}_{\text{adapt}}\right) \in \mathbb{R}^{C\times N_{\text{sc}}\times N_r N_t} . \tag{29}$$

The aggregated temporal feature and the auxiliary feature are concatenated along the channel dimension and fused by a 1×1 convolution:

$$\mathbf{F}_{\text{fus}} = \phi\big(\mathcal{B}\left(\mathbf{W}_f * [\mathbf{F}_{\text{agg}},\, \mathbf{F}_{\text{aux}}]\right)\big) . \tag{30}$$

The auxiliary branch uses $\mathrm{Conv}_{3\times3}(2, 64)$–BN–ReLU–$\mathrm{Conv}_{3\times3}(64, 64)$–BN–ReLU. After concatenation, a 1×1 convolution followed by batch normalization and ReLU projects the resulting 128-channel tensor to 64 channels.

The auxiliary estimate is therefore injected at the feature level rather than being directly added to the channel estimate. This design allows the network to exploit smoothed temporal context while limiting the risk of propagating a biased auxiliary estimate to the output.

All convolutional layers use unit stride. Padding is one for the 3×3 convolutions, three for the 7×7 CBAM spatial-attention convolution, and zero for the 1×1 projection and output convolutions. These settings preserve the spatial-frequency dimensions through feature extraction, fusion, attention, and decoding.

The fused feature is further refined using CBAM [15], which applies channel-wise and spatial attention sequentially. The channel-attention stage recalibrates the fused feature as:

$$\mathbf{F}_c = \mathcal{A}_{\text{ch}}\left(\mathbf{F}_{\text{fus}}\right) , \tag{31}$$

where $\mathcal{A}_{\text{ch}}(\cdot)$ denotes the channel-attention operation defined in the previous subsection.

The spatial-attention stage then emphasizes the most informative subcarriers and spatial-channel components:

$$\mathbf{F}_{\text{cbam}} = \mathbf{F}_c \odot \sigma\big(\mathbf{W}_{sp} * [\mathrm{AvgPool}_c(\mathbf{F}_c),\, \mathrm{MaxPool}_c(\mathbf{F}_c)]\big) , \tag{32}$$

where $\mathrm{AvgPool}_c(\cdot)$ and $\mathrm{MaxPool}_c(\cdot)$ denote average and maximum pooling along the channel dimension, respectively, and $\mathbf{W}_{sp}$ is the spatial-attention convolution kernel.

The channel-attention stage uses a reduction ratio of 8. The spatial-attention stage applies a 7×7 convolution to the concatenated channel-wise average and maximum maps.

3.4. Residual Output Stage

The refined feature is decoded into a residual correction by two convolutional layers. The decoder uses a 3×3 convolution from 64 to 32 channels, followed by batch normalization, ReLU, and a 1×1 convolution from 32 to 2 output channels:

$$\Delta\mathcal{H} = \mathbf{W}_{d2} * \phi\big(\mathcal{B}\left(\mathbf{W}_{d1} * \mathbf{F}_{\text{cbam}}\right)\big) . \tag{33}$$

The final channel estimate is then obtained by adding this correction to the current-slot LS estimate:

$$\hat{\mathcal{H}} = \hat{\mathcal{H}}_{\text{LS}}^{(T)} + \Delta\mathcal{H} . \tag{34}$$

The last convolutional layer is initialized with zero weights and zero bias, so that $\Delta\mathcal{H} = \mathbf{0}$ at the beginning of training and the initial output coincides with the current LS estimate. This initialization provides a stable starting point for optimization and encourages the network to learn only the residual estimation error rather than reconstructing the full channel response from scratch. Moreover, the residual connection preserves a direct path from the current pilot-based observation to the output, which helps retain fine channel details, especially when the LS estimate becomes more reliable at high SNR.

The temporal branch, the auxiliary EMA branch, and the residual output stage therefore play complementary roles. The temporal branch exploits inter-slot correlation, the auxiliary branch provides smoothed contextual information, and the residual path anchors the final estimate to the current-slot LS observation. This design improves robustness without directly adding a potentially biased smoothed estimate to the output.

3.5. Training Objective

The network is trained in a supervised manner by minimizing a composite loss function. The primary term is the normalized mean square error between the estimated channel and the true current-slot channel:

$$\mathcal{L}_{\text{NMSE}} = \frac{1}{N_b}\sum_{i=1}^{N_b} \frac{\left\|\mathcal{H}_i - \hat{\mathcal{H}}_i\right\|_F^2}{\left\|\mathcal{H}_i\right\|_F^2 + \varepsilon} , \tag{35}$$

where N_b is the batch size and ε is a small positive constant introduced for numerical stability.

To encourage consistency between the aggregated temporal representation and the auxiliary EMA-based representation, an additional feature-consistency term is used during training:

$$\mathcal{L}_{\text{cons}} = \frac{1}{N_b}\sum_{i=1}^{N_b}\left(1 - \frac{\left\langle \bar{\mathbf{f}}_{\text{agg},i},\, \bar{\mathbf{f}}_{\text{aux},i}\right\rangle}{\left\|\bar{\mathbf{f}}_{\text{agg},i}\right\| \left\|\bar{\mathbf{f}}_{\text{aux},i}\right\| + \varepsilon}\right) , \tag{36}$$

where $\bar{\mathbf{f}}_{\text{agg},i}$ and $\bar{\mathbf{f}}_{\text{aux},i}$ denote the globally averaged temporal and auxiliary feature vectors of the i-th sample, respectively, and $\langle\cdot,\cdot\rangle$ denotes the inner product.

The Adam optimizer [24] is used for training, together with a plateau-based learning-rate scheduler and validation-based checkpoint selection. The total training loss is defined as

$$\mathcal{L} = \mathcal{L}_{\text{NMSE}} + \lambda\,\mathcal{L}_{\text{cons}} , \tag{37}$$

where λ controls the strength of the consistency regularization. In the implemented configuration, this term is used only as an auxiliary training regularizer. It is not required during inference and does not introduce additional computational cost at test time.

The consistency term encourages alignment between the learned spatio-temporal representation and the EMA-derived smoothed context, while the NMSE term directly optimizes channel reconstruction accuracy. Its empirical contribution is scenario dependent, as investigated in Subsection 4.5.

The complete procedure is summarized in Algorithm 1.

During inference, SMART-CE requires only a single forward pass and does not involve any iterative optimization. Given the LS sequence, the EMA estimate, and the SNR value, the trained network directly produces the refined channel

Algorithm 1 SMART-CE channel estimation

Input: LS sequence $\mathcal{H}_{\text{seq}} = \{\hat{\mathcal{H}}_{\text{LS}}^{(t)}\}_{t=1}^{T}$, EMA estimate $\hat{\mathcal{H}}_{\text{adapt}}$, SNR value γ, and trained parameters θ
Output: Refined channel estimate $\hat{\mathcal{H}}$
Start

for $t = 1$ to T **do**
  Feature extraction: $\mathbf{F}^{(t)} = f_\theta\left(\hat{\mathcal{H}}_{\text{LS}}^{(t)}\right)$
  Compute temporal score: $\mathbf{s}^{(t)} = \mathbf{W}_s * \mathbf{F}^{(t)}$
end for
Compute SNR-dependent temporal parameters:
$$[\tau, b_1, \ldots, b_T] = f_{\text{hyp}}(\gamma/\gamma_0)$$
Compute temporal attention weights:
$$A^{(t)} = \text{softmax}_t\big(\tau s^{(t)} + b_t\big)$$
Aggregate temporal features:
$$F_{\text{agg}} = \sum_{t=1}^{T} A^{(t)} \odot F^{(t)}$$
Extract auxiliary EMA-based feature:
$$F_{\text{aux}} = \psi(\hat{H}_{\text{adapt}})$$
Fuse temporal and auxiliary features:
$$F_{\text{fus}} = \phi\big(B(W_f * [F_{\text{agg}}, F_{\text{aux}}])\big)$$
Refine by CBAM: $F_{\text{cbam}} = \text{CBAM}(F_{\text{fus}})$
Decode residual correction:
$$\Delta H = W_{d2} * \phi\big(B(W_{d1} * F_{\text{cbam}})\big)$$
Produce final estimate:
$$\hat{H} = \hat{H}_{\text{LS}}^{(T)} + \Delta H$$
return $\hat{H}$
End

estimate. This non-iterative inference procedure makes the proposed framework suitable for practical MIMO-OFDM channel estimation scenarios where improved estimation accuracy must be achieved under constrained processing latency.

4. Simulation Results

In this section, we evaluate the channel estimation performance of the proposed SMART-CE framework through numerical simulations. We first describe the simulation setup, the dataset generation procedure, the compared estimation schemes, and the adopted performance metrics. Then, the results are discussed by comparing SMART-CE with LS, adaptive filters, and learning-based baselines under two representative MIMO-OFDM propagation scenarios: time-varying Rician fading and spatially correlated MIMO channels.

4.1. Training and Testing of SMART-CE

The datasets are generated according to the MIMO-OFDM system model described in Sec. 2. For each channel realization, a temporal trajectory of T consecutive slots is generated in order to provide both the current pilot-based observation and the previous temporally correlated observations. The mobility level is varied by drawing the user velocity uniformly within the predefined range, which allows the dataset to cover low- and moderate-mobility conditions. For each slot, the received pilot observations are generated under a randomly selected SNR value and the corresponding LS estimate is computed using orthogonal DFT pilots.

The resulting temporal sequence of LS estimates $\mathcal{H}_{\text{seq}}$ and the EMA-based auxiliary estimate $\hat{\mathcal{H}}_{\text{adapt}}$ are used as inputs to SMART-CE, while the true channel of the current slot $\mathcal{H}$ is used as the supervised learning target. The reported evaluation focuses on two representative MIMO-OFDM propagation scenarios: time-varying Rician fading and spatially correlated MIMO channels.

The training configuration uses an initial learning rate of 10^{-3}, weight decay 10^{-4}, a ReduceLROnPlateau scheduler with reduction factor 0.5, gradient clipping at unit norm, and early stopping with patience 12.

Training is limited to 50 epochs with batch size 128. For each training trajectory, one SNR value is drawn independently from the continuous uniform distribution $\mathcal{U}[-10, 20]$ dB and retained across its five slots. At test time, the SNR is fixed at the value associated with each evaluated point. All learning-based models are trained under the same data generation settings and then evaluated over the considered SNR range on the retained time-varying Rician and spatially correlated MIMO scenarios.

The proposed SMART-CE model is compared with LS, complex-valued NLMS and RLS adaptive estimators, as well as CNN, Res-CNN, and RA-CNN learning-based baselines. NLMS and RLS are initialized using the first-slot LS estimate and process the pilot observations from slots $t = 2, \ldots, T$. The NLMS and RLS implementations, including their parameter configurations, follow the standard complex-valued adaptive-filtering update rules and settings reported in [25]. All estimators are evaluated using the same channel trajectories, pilot observations, noise realizations, SNR values, and current-slot channel targets. CNN, Res-CNN, and RA-CNN use only the current-slot LS estimate, whereas NLMS, RLS, and SMART-CE exploit observations from the five-slot temporal window.

The main simulation parameters are summarized in Tab. 1.

4.2. Evaluation Metrics

The performance is assessed using two complementary metrics. The first metric is NMSE, which directly quantifies the channel estimation accuracy and is defined as:

$$\text{NMSE} = \mathbb{E}\left[\frac{\left\|\mathcal{H} - \hat{\mathcal{H}}\right\|_F^2}{\|\mathcal{H}\|_F^2}\right], \tag{38}$$

where $\mathcal{H}$ and $\hat{\mathcal{H}}$ denote the true and estimated channel tensors, respectively. The tables and robustness figures report NMSE in dB as $10\log_{10}(\text{NMSE})$.

Tab. 1. Simulation parameters.

Parameter	Value
Transmit antennas	$N_t = 8$
Receive antennas	$N_r = 8$
OFDM subcarriers	$N_{\text{sc}} = 64$
Pilot type	Orthogonal DFT pilots
Channel models	Time-varying Rician, spatially correlated MIMO
Rician K-factor	5 dB
Mobile velocity	0 – 60 km/h
Carrier frequency	$f_c = 3.5$ GHz
Inter-slot time interval	$T_s = 1$ ms
Spatial correlation	$\rho_t = \rho_r = 0.7$
Temporal window length	$T = 5$ slots
Training samples	40 000
Validation samples	4000
Testing samples per SNR	3000
Evaluation seeds	$\{11, 22, 33, 44, 55\}$
SNR-robustness seeds	$\{11, 22, 33\}$
SNR-input error range	$[-3, 3]$ dB
Training epochs	50
Batch size	128
Optimizer	Adam
Learning rate	10^{-3}
Training SNR	$\gamma_{\text{train}} \sim \mathcal{U}[-10, 20]$ dB
SNR normalization	$\gamma_0 = 20$ dB
Temperature offset	$\tau_0 = 0.5$
Convolution stride	1
Convolution padding	1 for 3×3, 3 for 7×7, and 0 for 1×1
Base channels	$C = 64$
Residual attention blocks	$N_{\text{RB}} = 6$
EMA smoothing factor	$\beta = 0.6$
Consistency weight	$\lambda = 0.1$
Loss function	NMSE + consistency loss
Reported metrics	NMSE, BER
Hardware accelerator	NVIDIA T4 GPU

The second metric is BER, which evaluates the end-to-end impact of channel estimation errors on data detection. QPSK symbols are transmitted through the true channel, while the receiver performs detection using an MMSE equalizer constructed from the estimated channel. For each subcarrier, the equalized symbol vector is obtained as

$$\hat{\mathbf{x}} = \left(\hat{\mathbf{H}}^H \hat{\mathbf{H}} + \frac{1}{\rho} \mathbf{I}_{N_t} \right)^{-1} \hat{\mathbf{H}}^H \mathbf{y} \,, \tag{39}$$

where $\hat{\mathbf{H}}$ is the estimated channel matrix, $\mathbf{y}$ is the received signal vector generated through the true channel, ρ is the linear SNR, and $\mathbf{I}_{N_t}$ is the identity matrix of size N_t.

The BER is then computed by comparing the detected QPSK bits with the transmitted bits and averaging the error rate over all test realizations at each SNR value.

4.3. Multi-seed Protocol

To quantify variability across test realizations without changing the trained estimators, the selected checkpoint of each learning-based method is evaluated on five independently generated held-out test sets with seeds $\{11, 22, 33, 44, 55\}$. For each seed, all estimators receive the same channel trajectories, pilot observations, and noise realizations. The resulting variability therefore reflects changes in the test data rather than independent network retraining.

For a metric value z_m obtained with evaluation seed m, the reported mean and two-sided 95% Student-t confidence interval are computed as

$$\bar{z} = \frac{1}{M} \sum_{m=1}^{M} z_m \,, \qquad \bar{z} \pm t_{0.975, M-1} \frac{s_z}{\sqrt{M}} \,, \tag{40}$$

where $M = 5$, s_z is the sample standard deviation, and $t_{0.975,4} = 2.776$.

The original NMSE and BER curves are retained, while the five-seed results are reported in tables as the mean and confidence interval.

4.4. Performance Evaluation of the Proposed Model

The performance of SMART-CE is evaluated in terms of NMSE and BER over the considered SNR range. The reported results focus on two representative MIMO-OFDM propagation scenarios: time-varying Rician fading and spatially correlated MIMO channels. The proposed method is compared with LS, RLS, NLMS, CNN, ResCNN, and RA-CNN baselines.

Figures 2 and 3 report the point-estimate NMSE and BER curves, respectively, for the time-varying Rician channel over the complete SNR range. SMART-CE achieves the lowest NMSE and BER values across the evaluated SNR range, with the most pronounced gains over the learning-based baselines in the low-to-medium SNR regime. These curves represent the reference point-estimate evaluation and are considered separately from the five-seed statistics reported below.

Figures 4 and 5 compare the point-estimate NMSE and BER results for the spatially correlated MIMO channel. Each channel realization remains fixed over the five-slot observation window and is regenerated independently for every test sample, allowing NLMS and RLS to combine repeated observations without motion-induced tracking bias. SMART-CE generally attains lower NMSE than the CNN-based estimators and adaptive filters at low and medium SNR. As the SNR increases, the NMSE curves become closer, and the adaptive filters achieve slightly lower values at the highest evaluated SNR.

The BER results follow a different trend: SMART-CE maintains the lowest BER across the complete SNR range, with a wider separation at medium and high SNR. Under the evaluated spatially correlated conditions, NLMS and RLS are there-

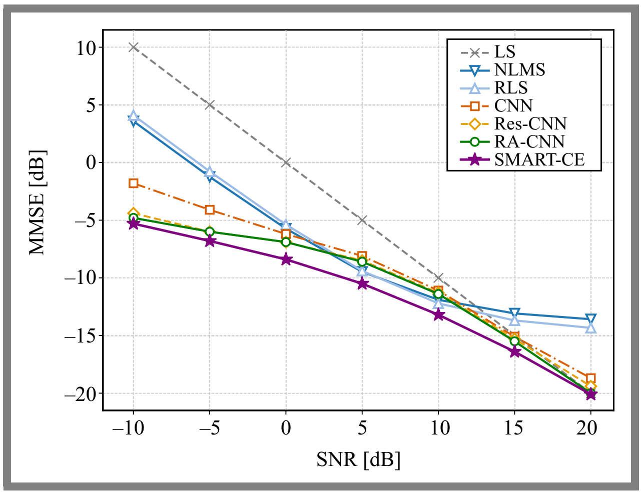

Fig. 2. NMSE performance over the time-varying Rician channel.

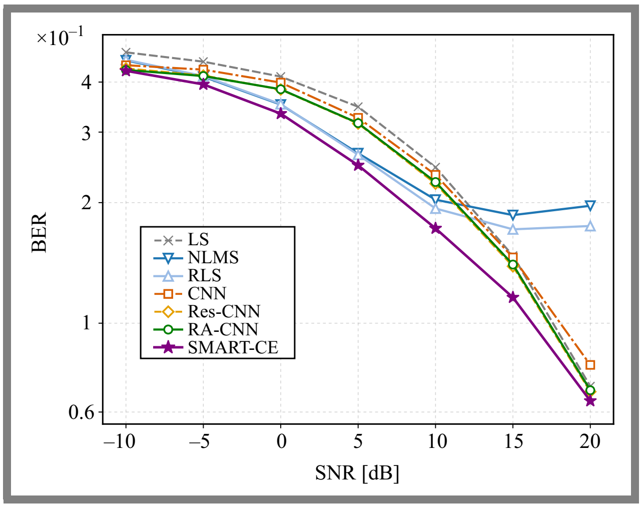

Fig. 3. BER performance over the time-varying Rician channel.

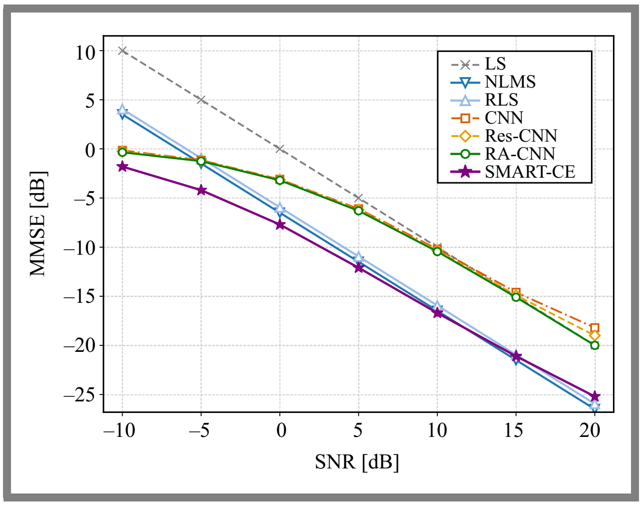

Fig. 4. NMSE performance over the spatially correlated MIMO channel.

fore effective for high-SNR channel reconstruction, whereas the SMART-CE estimates remain more favorable for MMSE-based data detection.

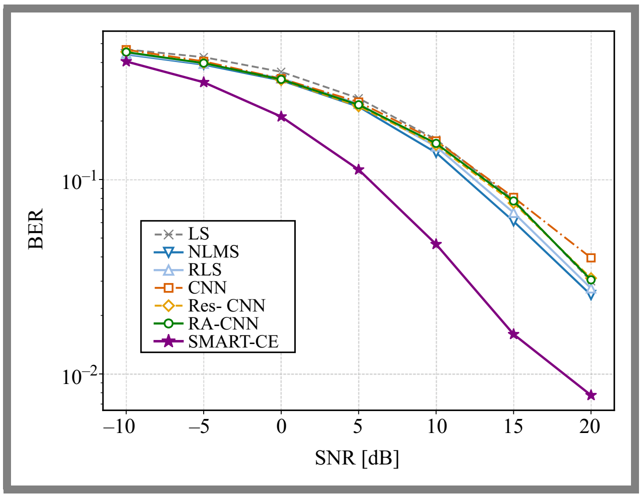

Fig. 5. BER performance over the spatially correlated MIMO channel.

Table 2 reports the mean NMSE and the corresponding 95% Student-t confidence interval for five independently generated held-out test sets over $\text{SNR} \in \{-10, -5, 0, 5, 10\}$ dB. For the time-varying Rician channel, SMART-CE gives the lowest mean NMSE at all five SNR values. At 0 dB, its mean NMSE is approximately 1.40 dB below that of RA-CNN, the closest learning-based baseline at this operating point. For the spatially correlated channel, SMART-CE gives the lowest mean NMSE at four of the five SNR values and remains close to the leading adaptive estimator at 10 dB.

Table 3 reports the corresponding mean BER and 95% Student-t confidence interval. SMART-CE gives the lowest mean BER at all five SNR values in the time-varying Rician channel. In the spatially correlated channel, it gives the lowest mean BER at four of the five SNR values and remains close to the leading adaptive estimator at 10 dB. The narrow intervals indicate limited variability across the five test sets; they do not imply uniform superiority under all propagation conditions.

4.5. Component Ablation Analysis

To isolate the contribution of each component, four ablated variants were retrained using the same data-generation procedure, optimization settings, and validation-based checkpoint selection as the complete SMART-CE model. The variants remove the SNR-conditioned hyper-network, CBAM refinement, EMA auxiliary branch, and feature-consistency loss, respectively. All variants use the same test seed (42), 3000 samples per SNR, and the SNR set $\{-10, -5, 0, 5, 10\}$ dB in both propagation scenarios.

Figure 6 reports the resulting NMSE and BER curves. As shown in Fig. 6a and Fig. 6c, removing the SNR hyper-network substantially increases both NMSE and BER over the time-varying Rician channel. The same trend is observed for the spatially correlated MIMO channel in Fig. 6b and Fig. 6d. This comparison is particularly important because the SNR-conditioned temporal weighting mechanism constitutes the main methodological contribution of SMART-CE.

Tab. 2. Mean NMSE and 95% Student-t confidence-interval half-width over five fixed-checkpoint evaluation seeds.

Time-varying Rician					
Method	**–10**	**–5**	**0**	**5**	**10**
LS	10.00 ±0.002	5.00 ±0.003	0.00 ±0.002	–5.00 ±0.002	–10.00 ±0.001
NLMS	3.59 ±0.002	–1.24 ±0.004	–5.74 ±0.001	–9.42 ±0.038	–11.84 ±0.045
RLS	4.10 ±0.003	–0.77 ±0.004	–5.39 ±0.001	–9.35 ±0.032	–12.15 ±0.041
CNN	–1.81 ±0.004	–4.05 ±0.006	–6.20 ±0.005	–8.11 ±0.002	–11.06 ±0.001
Res CNN	–4.38 ±0.009	–5.96 ±0.003	–6.91 ±0.001	–8.52 ±0.001	–11.39 ±0.001
RA-CNN	–4.84 ±0.005	–5.98 ±0.002	–6.93 ±0.002	–8.55 ±0.001	–11.44 ±0.001
SMART-CE	**–5.36 ±0.002**	**–6.79 ±0.002**	**–8.33 ±0.001**	**–10.51 ±0.034**	**–13.23 ±0.023**
Spatially correlated MIMO					
Method	**–10**	**–5**	**0**	**5**	**10**
LS	10.01 ±0.006	5.01 ±0.005	0.01 ±0.003	–4.99 ±0.002	–9.99 ±0.004
NLMS	3.51 ±0.006	–1.49 ±0.004	–6.49 ±0.003	–11.49 ±0.003	–16.49 ±0.004
RLS	4.05 ±0.005	–0.95 ±0.004	–5.95 ±0.003	–10.95 ±0.003	–15.95 ±0.004
CNN	–0.18 ±0.002	–1.44 ±0.003	–3.25 ±0.002	–5.72 ±0.003	–10.16 ±0.005
Res CNN	–0.80 ±0.002	–2.07 ±0.002	–4.06 ±0.002	–6.13 ±0.005	–10.23 ±0.005
RA-CNN	–0.63 ±0.001	–1.67 ±0.002	–3.58 ±0.002	–6.39 ±0.004	–10.14 ±0.006
SMART-CE	**–2.25 ±0.003**	**–4.59 ±0.004**	**–7.92 ±0.004**	**–12.12 ±0.003**	**–16.34 ±0.011**

Tab. 3. Mean BER and 95% Student-t confidence-interval half-width over five fixed-checkpoint evaluation seeds.

Time-varying Rician					
Method	**–10**	**–5**	**0**	**5**	**10**
LS	0.4745 ±0.0004	0.4499 ±0.0020	0.4099 ±0.0011	0.3441 ±0.0021	0.2519 ±0.0029
NLMS	0.4514 ±0.0011	0.4128 ±0.0016	0.3473 ±0.0021	0.2629 ±0.0040	0.2067 ±0.0051
RLS	0.4528 ±0.0004	0.4146 ±0.0015	0.3501 ±0.0024	0.2624 ±0.0027	0.1982 ±0.0047
CNN	0.4409 ±0.0005	0.4288 ±0.0021	0.3947 ±0.0019	0.3230 ±0.0020	0.2403 ±0.0031
Res CNN	0.4317 ±0.0006	0.4157 ±0.0006	0.3802 ±0.0017	0.3123 ±0.0022	0.2293 ±0.0026
RA-CNN	0.4282 ±0.0006	0.4138 ±0.0010	0.3795 ±0.0018	0.3130 ±0.0024	0.2313 ±0.0029
SMART-CE	**0.4264 ±0.0007**	**0.3938 ±0.0014**	**0.3303 ±0.0023**	**0.2455 ±0.0032**	**0.1757 ±0.0027**
Spatially correlated MIMO					
Method	**–10**	**–5**	**0**	**5**	**10**
LS	0.4729 ±0.0026	0.4446 ±0.0016	0.4018 ±0.0016	0.3478 ±0.0018	0.2750 ±0.0036
NLMS	0.4415 ±0.0008	0.3903 ±0.0015	0.3205 ±0.0021	0.2318 ±0.0017	0.1382 ±0.0018
RLS	0.4444 ±0.0007	0.3954 ±0.0013	0.3280 ±0.0016	0.2412 ±0.0006	0.1472 ±0.0021
CNN	0.4658 ±0.0022	0.4206 ±0.0016	0.3849 ±0.0018	0.3436 ±0.0020	0.2741 ±0.0039
Res CNN	0.4507 ±0.0034	0.4134 ±0.0016	0.3781 ±0.0018	0.3400 ±0.0016	0.2729 ±0.0032
RA-CNN	0.4527 ±0.0028	0.4152 ±0.0015	0.3808 ±0.0018	0.3378 ±0.0015	0.2737 ±0.0036
SMART-CE	**0.4207 ±0.0011**	**0.3720 ±0.0017**	**0.3090 ±0.0020**	**0.2249 ±0.0020**	**0.1400 ±0.0022**

For the time-varying Rician channel, the NMSE gap between the complete model and the variant without the SNR hyper-network decreases from 8.238 dB at –10 dB to 1.547 dB at 10 dB, with an average degradation of 4.170 dB across the five evaluated SNR values. For the spatially correlated channel, the NMSE gap ranges from 2.163 to 5.626 dB, with an average degradation of 3.372 dB. The corresponding average BER increases are 0.05359 and 0.04675, respectively.

The larger degradation at low SNR is consistent with the intended role of the hyper-network: historical observations are most useful when the current LS estimate is noise limited, whereas their influence should be reduced as the current observation becomes more reliable. These results support the proposed SNR-conditioned balance between temporal noise reduction and motion-induced tracking bias. As further illustrated in Fig. 6a–d, the EMA branch also contributes

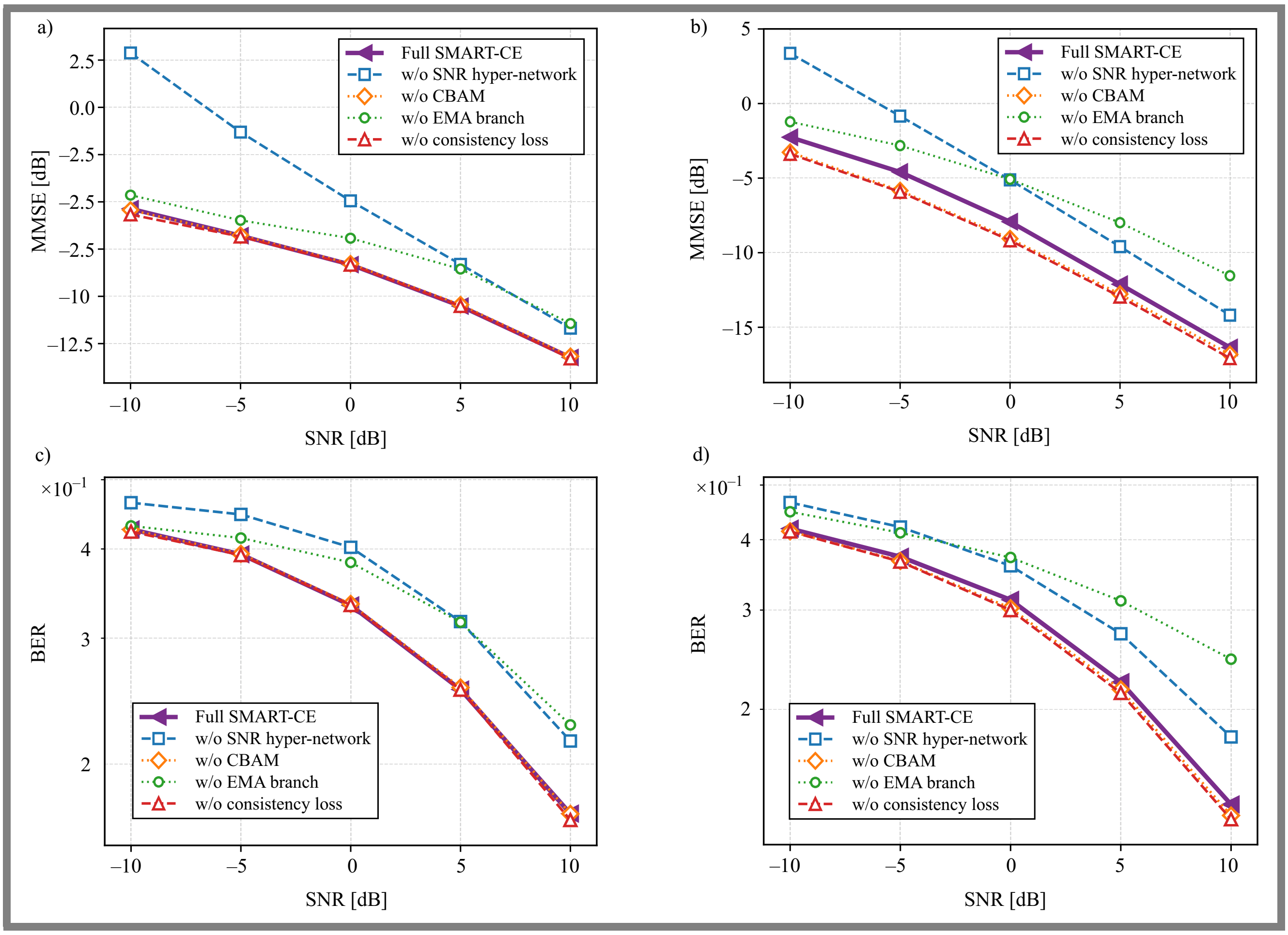

Fig. 6. Component-wise ablation of SMART-CE over the representative SNR range.

consistently, since its removal increases NMSE and BER at all five SNR values in both propagation scenarios. The corresponding average NMSE penalties are 1.336 dB for the Rician channel and 2.918 dB for the spatially correlated channel.

By contrast, removing CBAM or the consistency loss produces only small changes under Rician fading and lower errors under some spatially correlated conditions.

Therefore, the observed performance gain cannot be attributed equally to all architectural components. It is primarily associated with the SNR-conditioned temporal weighting mechanism, while the EMA branch provides complementary temporal context. CBAM and consistency regularization remain supporting components whose effects depend on the evaluated propagation scenario.

4.6. Robustness to Imperfect SNR Information

In practical receivers, the SNR supplied to the hyper-network may differ from the true operating SNR. This mismatch is modeled as $\hat{\gamma} = \gamma + \Delta\gamma$, where γ denotes the true operating SNR and $\Delta\gamma$ is the receiver-side SNR-estimation error. Robustness is evaluated for $\Delta\gamma \in \{-3, -2, -1, 0, 1, 2, 3\}$ dB at true SNR values $\{0, 5, 10, 15\}$ dB.

The trained SMART-CE checkpoint is kept fixed, and each condition is evaluated on three independently generated test sets using seeds $\{11, 22, 33\}$ and 3000 samples per seed. For each channel model, true SNR, and test seed, the same test data are used for all seven perturbations. Therefore, only the SNR value supplied to the hyper-network is modified, allowing the effect of the SNR-input error to be isolated.

Mean NMSE and two-sided 95% Student-t confidence intervals are computed using Eq. (40), with $M = 3$ and $t_{0.975,2} = 4.303$. These intervals quantify variability across independently generated test sets for a fixed SMART-CE checkpoint rather than variability across independent training runs.

Figures 7 and 8 show the NMSE sensitivity of SMART-CE to SNR-estimation errors under time-varying Rician fading and spatially correlated MIMO channels, respectively. The reference point $\Delta\gamma = 0$ corresponds to perfect SNR information. Across the evaluated true-SNR values, the maximum increase in mean NMSE relative to the perfect-SNR case remains below 0.05 dB. These results indicate that SMART-CE is robust to moderate SNR-input errors within the tested range.

The validation loss convergence curves of the learning-based estimators are shown in Fig. 9. From the first epoch, SMART-CE achieves a markedly lower validation loss, around 0.178,

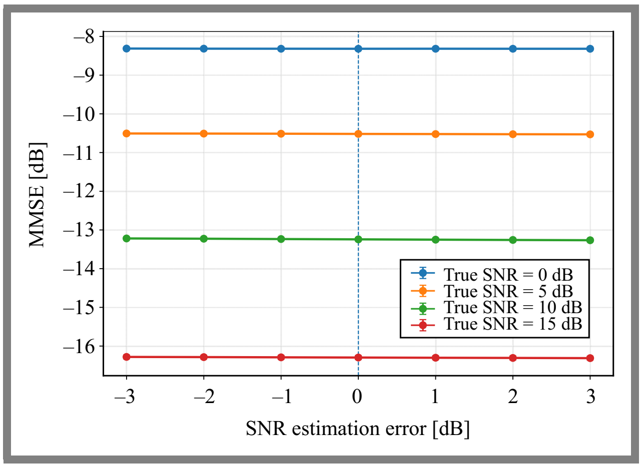

Fig. 7. SMART-CE robustness to SNR-estimation errors under time-varying Rician fading.

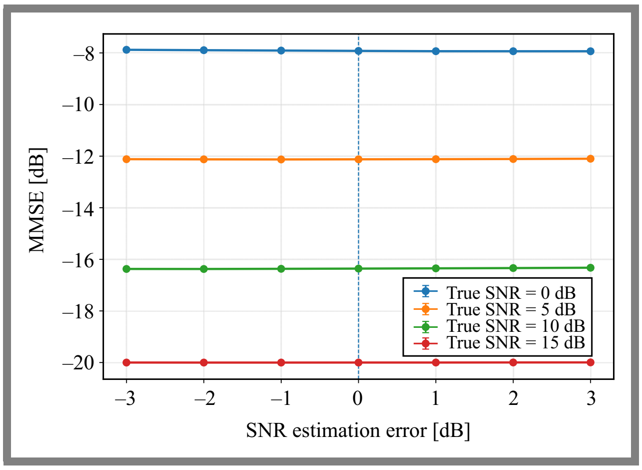

Fig. 8. SMART-CE robustness to SNR-estimation errors under spatially correlated MIMO channels.

compared with CNN, Res-CNN, and RA-CNN, which start at approximately 0.298, 0.240, and 0.260, respectively. This early advantage highlights the effectiveness of the residual initialization strategy, which guides the network output toward the current LS estimate from the initial training stage.

Although all models reach a stable plateau before epoch 35, SMART-CE attains the lowest final validation loss, approximately 0.110. This corresponds to a reduction of about 40% compared with RA-CNN, around 0.183, and 53% compared with CNN, around 0.235. It is also worth noting that Res-CNN and RA-CNN converge to almost the same final loss, indicating that spatial attention alone, in the absence of temporal information, provides only a marginal improvement over residual learning. Overall, these results demonstrate that the combined use of temporal correlation, SNR-adaptive fusion, and CBAM-based feature refinement enables SMART-CE to achieve faster, more stable, and more accurate convergence.

Table 4 compares the parameter count, forward-pass computational complexity, and batch-one inference latency of the learning-based estimators. SMART-CE has the highest computational cost among the evaluated models because it processes a sequence of five LS estimates and incorporates residual-attention blocks, SNR-adaptive temporal fusion,

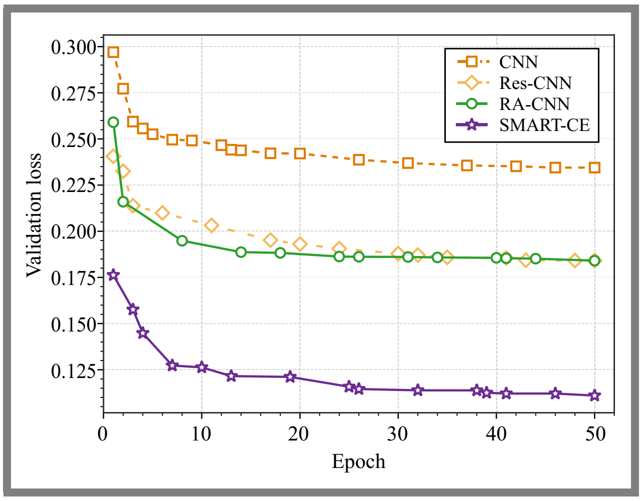

Fig. 9. Validation loss convergence of the learning-based estimators.

Tab. 4. Computational complexity and batch-one inference latency of the learning-based channel estimators on an NVIDIA T4 GPU.

Model	Parameters	GFLOPs	Runtime [ms/sample]
CNN	296 962	2.426	1.077 ±0.049
Res-CNN	463 746	3.785	2.455 ±0.619
RA-CNN	469 890	3.785	3.550 ±0.504
SMART-CE	518 987	18.700	16.130 ±4.097

CBAM refinement, and an auxiliary EMA branch. The inference runtime is reported as the mean and standard deviation over 100 batch-one forward passes performed after 30 warm-up iterations on an NVIDIA T4 GPU. FLOPs account for convolutional and linear-layer operations, with multiplications and additions counted separately, whereas element-wise operations and softmax computations are excluded.

5. Conclusions

This paper proposed SMART-CE, a spatio-temporal residual attention framework for MIMO-OFDM channel estimation. The proposed network combines a shared residual attention encoder, an SNR-adaptive temporal fusion mechanism, CBAM-based feature refinement, and an EMA-based auxiliary branch to exploit both temporal correlation and spatial-frequency channel structure.

The final estimate is obtained through a residual correction of the current LS estimate, which stabilizes learning and preserves a direct connection to the pilot-based observation. The ablation results identify the SNR-conditioned temporal weighting mechanism as the main source of the consistent performance gain, with the EMA branch providing complementary temporal context.

SMART-CE achieved lower NMSE and BER than the evaluated baselines over most of the tested conditions. Although SMART-CE requires a higher computational cost than simpler CNN-based baselines, it performs channel estimation through a single forward pass without iterative optimization.

References

[1] I. Almeida, J. Guerreiro, and R. Dinis, "On Deep Learning Hybrid Architectures for MIMO-OFDM Channel Estimation", *Electronics*, vol. 14, art. no. 4692, 2025 (https://doi.org/10.3390/electronics14234692).

[2] Q. Wu *et al.*, "A Convolutional-transformer Residual Network for Channel Estimation in Intelligent Reflective Surface Aided MIMO Systems", *Sensors*, vol. 25, art. no. 5959, 2025 (https://doi.org/10.3390/s25195959).

[3] B. Zheng, C. You, W. Mei, and R. Zhang, "A Survey on Channel Estimation and Practical Passive Beamforming Design for Intelligent Reflecting Surface Aided Wireless Communications", *IEEE Communications Surveys & Tutorials*, vol. 24, pp. 1035–1071, 2022 (https://doi.org/10.1109/COMST.2022.3155305).

[4] A. Melgar *et al.*, "Deep Neural Network: An Alternative to Traditional Channel Estimators in Massive MIMO Systems", *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, pp. 657–671, 2022 (https://doi.org/10.1109/TCCN.2022.3164888).

[5] H. Hirose, S. Yang, T. Ohtsuki, and M. Bouazizi, "Deep Learning-based Channel Estimation to Mitigate Channel Aging in Massive MIMO with Pilot Contamination", *IEEE Access*, vol. 13, pp. 1834–1845, 2025 (https://doi.org/10.1109/ACCESS.2024.3518763).

[6] J. Yuan, H.Q. Ngo, and M. Matthaiou, "Machine Learning-based Channel Prediction in Massive MIMO with Channel Aging", *IEEE Transactions on Wireless Communications*, vol. 19, pp. 2960–2974, 2020 (https://doi.org/10.1109/TWC.2020.2969627).

[7] R. Fateh, A. Darif, and S. Safi, "Performance Evaluation of MC-CDMA Systems with Single User Detection Technique Using Kernel and Linear Adaptive Method", *Journal of Telecommunications and Information Technology*, vol. 86, pp. 1–11, 2021 (https://doi.org/10.26636/jtit.2021.151621).

[8] R. Fateh *et al.* "Machine Learning Based System Identification with Binary Output Data Using Kernel Methods", *Journal of Telecommunications and Information Technology*, vol. 95, pp. 17–25, 2024 (https://doi.org/10.26636/jtit.2024.1.1430).

[9] R. Fateh *et al.*, "A Novel Kernel Algorithm for Finite Impulse Response Channel Identification", *Journal of Telecommunications and Information Technology*, vol. 92, pp. 84–93, 2023 (https://doi.org/10.26636/jtit.2023.169823).

[10] W. Shi *et al.*, "Adaptive Channel Estimation Based on Multidirectional Structure in Delay-Doppler Domain for Underwater Acoustic OTFS System", *Remote Sensing*, vol. 16, art. no. 3157, 2024 (https://doi.org/10.3390/rs16173157).

[11] B. Marinberg, A. Cohen, E. Ben-Dror, and H.H. Permuter, "A Study on MIMO Channel Estimation by 2D and 3D Convolutional Neural Networks", *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, New Delhi, India, 2020 (https://doi.org/10.1109/ANTS50601.2020.9342797).

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, 2016 (https://doi.org/10.1109/CVPR.2016.90).

[13] X. Ma, Z. Gao, F. Gao, and M. Di Renzo, "Model-driven Deep Learning Based Channel Estimation and Feedback for Millimeter-wave Massive Hybrid MIMO Systems", *IEEE Journal on Selected Areas in Communications*, vol. 39, pp. 2388–2406, 2021 (https://doi.org/10.1109/JSAC.2021.3087269).

[14] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation Networks", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, USA, 2018 (https://doi.org/10.1109/CVPR.2018.00745).

[15] S. Woo, J. Park, J.-Y. Lee, and I.S. Kweon, "CBAM: Convolutional Block Attention Module", *Proc. of European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018 (https://doi.org/10.1007/978-3-030-01234-2_1).

[16] 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, art. no. 2597866, 2025 (https://doi.org/10.1155/int/2597866).

[17] C. Eom and C. Lee, "Hybrid Neural Network-based Fading Channel Prediction for Link Adaptation", *IEEE Access*, vol. 9, pp. 117257–117266, 2021 (https://doi.org/10.1109/ACCESS.2021.3106739).

[18] S. Moon, H. Kim, and I. Hwang, "Deep Learning-based Channel Estimation and Tracking for Millimeter-wave Vehicular Communications", *Journal of Communications and Networks*, vol. 22, pp. 177–184, 2020 (https://doi.org/10.1109/JCN.2020.000012).

[19] T. Yassine and L. Le Magoarou, "mpNet: Variable Depth Unfolded Neural Network for Massive MIMO Channel Estimation", *IEEE Transactions on Wireless Communications*, vol. 21, pp. 5703–5714, 2022 (https://doi.org/10.1109/TWC.2022.3142737).

[20] R. Li, J. Sun, J. Xue, and C. Masouros, "Scenario-aware Learning Approaches to Adaptive Channel Estimation", *IEEE Transactions on Communications*, vol. 72, pp. 874–889, 2024 (https://doi.org/10.1109/TCOMM.2023.3330878).

[21] W.C. Jakes, *Microwave Mobile Communications*, New York, USA: Wiley, 656 p., 1974 (ISBN: 9780780310698).

[22] C.B. Peel and A.L. Swindlehurst, "Effective SNR for Space-time Modulation Over a Time-varying Rician Channel", *IEEE Transactions on Communications*, vol. 52, pp. 17–23, 2004 (https://doi.org/10.1109/TCOMM.2003.822148).

[23] R.H. Clarke, "A Statistical Theory of Mobile-radio Reception", *The Bell System Technical Journal*, vol. 47, pp. 957–1000, 1968 (https://doi.org/10.1002/j.1538-7305.1968.tb00069.x).

[24] D.P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization", *ArXiv*, 2015 (https://doi.org/10.48550/arXiv.1412.6980).

[25] R. Fateh, A. Darif, and S. Safi, "Kernel and Linear Adaptive Methods for the BRAN Channels Identification", *Advanced Intelligent Systems for Sustainable Development (AI2SD'2020)*, vol. 1418, 2022 (https://doi.org/10.1007/978-3-030-90639-9_47).

Ousama El-azizi, Ph.D.
LIMATI Laboratory
https://orcid.org/0009-0006-4299-0083
E-mail: elazizi.oussama@usms.ac.ma
Sultan Moulay Slimane University, Beni Mellal, Morocco
https://www.usms.ac.ma

Rachid Fateh, Professor
LIMATI Laboratory, IMIS Research Laboratory
https://orcid.org/0000-0002-0574-2105
E-mail: fateh.smi@gmail.com
Sultan Moulay Slimane University, Beni Mellal, Morocco
https://www.usms.ac.ma
Ibn Zohr University, Agadir, Morocco
https://www.uiz.ac.ma

Safi Said, Professor
LIMATI Laboratory
https://orcid.org/0000-0003-3390-9037
E-mail: safi.said@gmail.com
Sultan Moulay Slimane University, Beni Mellal, Morocco
https://www.usms.ac.ma