SEMARAK ILMU PUBLISHING

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Proposing LMU-GRUs Based Channel Estimation in Massive MIMO System

Hany Helmy[1,*], Sherif El Daysti[2], Hazem Shatila[3], Mohamed Aboul-Dahab[4]

[1] Cairo Airport Company (CAC), Cairo, Egypt
[2] Department of Electronics, Arab Academy for Science, Technology and Maritime Transport (AASTMT), Cairo, Egypt
[3] Virginia Tech, Artificial Intelligence & Markovdata, CEO, Cairo, Egypt
[4] Electronics and Communication Engineering, Arab Academy for Science, Technology and Maritime Transport (AASTMT), Cairo, Egypt

ARTICLE INFO

Article history:
Received 27 November 2025
Received in revised form 20 February 2026
Accepted 15 April 2026
Available online 16 June 2026

Keywords:
Massive MIMO; FDD; Compressed Sensing; Deep Learning; Conventional Neural Network

ABSTRACT

A Massive Multiple-Input Multiple-Output (massive MIMO) system relies on channel state information (CSI) feedback to perform precoding and achieve performance gain in frequency division duplex (FDD) networks. However, the transmission of a massive MIMO system is subject to excessive feedback overhead. In this paper, we propose a Deep Learning (DL) approach-based channel estimation technique to enhance recovery quality and improve the trade-off between compression ratio (CR) and complexity of a massive MIMO system. The first proposed technique uses the Channel State Information Network combined with the gated recurrent unit (CsiNet-GRU). The second proposed technique is based upon using CsiNet combined with Legendre Memory Unit in GRUs layers (CsiNet-LMU-GRUs) which improves a novel memory cell for recurrent neural networks that dynamically maintains along by information across long windows of time using relatively few resources. Moreover, the proposed techniques use the dropout method to reduce overfitting during the learning process. The simulation results demonstrate that the proposed (CsiNet-GRU) and (CsiNet-LMU-GRUs) techniques result in a significant performance improvement when compared with existing techniques used in conjunction with massive MIMO systems.

1. Introduction

The fifth generation (5G) wireless communications networks exhibit extended facilities over 4G ones in terms of much higher system capacity, wider frequency spectrum by utilizing millimeter wave (mmWave) bands, higher data rates, ultra-low latency, and lower energy consumption, all at low cost [1]. Massive Multiple-Input Multiple-Output (MIMO) antenna configurations were devised to improve the performance of wireless communication systems, which are characterized by a very large number of users. They are therefore found most appropriate to be used in 5G networks since they have the characteristics that match its requirements. In addition, they are more suitable to be developed in the mmWave bands. However, the prediction of the channel characteristics is considered a factor of prime importance in the development of 5G networks. It is therefore essential to acquire sufficient knowledge about these characteristics to design and evaluate such networks.

* *Corresponding author.*
E-mail address: hany.nabil@cairo-airport.com

https://doi.org/10.37934/araset.60.5.393413

Nevertheless, the utilization of a massive MIMO framework poses new challenges in channel modeling [2]. The very large number of communication paths offered by massive MIMO configurations led to the introduction of channel state information (CSI) feedback techniques, which allowed a more accurate and dynamic estimate of channel parameters, as compared with other channel models such as the geometry-based stochastic channel models [3]. However, one challenge facing the practical implementation of CSI feedback networks is the computational complexity. A review of the CSI structures based on the spatial domain, the time/Doppler domain, and the frequency domain is given in [4].

To reduce the number of operations during signal processing, several low-rank (or sparse) approaches for channel estimation were investigated. The authors in [5] gave an overview of such approaches. The potential approaches of artificial intelligence (AI) and its sub-categories such as machine learning and deep learning were investigated in 5G systems to facilitate processing operations [6]. As far as massive MIMO configurations are concerned, several deep learning methods were proposed to estimate the communication channel state information (CSI) with a variety of adopted algorithms. The authors in [7] investigated the performance of the conventional minimum mean square error (MMSE) algorithm when used in massive MIMO CSI and proposed a method to predict the required correlation matrix.

The authors in [8] investigated the performance of the Least Absolute Shrinkage and Selection Operator (LASSO) algorithm with the use of an elastic net to get an estimate of the CSI in sparse channels. An approximate message passing (AMP) algorithm was proposed in [9] based on a sparsity–under sampling tradeoff approach to reconstruct the compressed signal. In [10], the authors developed a massive MIMO CSI system based on a sensing and recovery network. The proposed network estimated the channel structure from training samples and was called the Channel State Information Network (CsiNet). In [11], the authors utilized the linear regression (LR) and support vector machine (SVM) models to reduce the overhead for the downlink CSI estimation and feedback in frequency division duplex (FDD) massive MIMO systems.

In MIMO FDD systems with a relatively small number of elements, feedback CSI is obtained through a pre-defined codebook at both the transmitter and the receiver. However, in massive MIMO, the number of codewords increases exponentially with the number of antennas [12]. In [13], the authors proposed a relatively simple codebook construction for both the transmitters and receivers, making use of temporal correlations and channel gain differences. They also investigated the performance of different CSI feedback algorithms for distributed antenna systems (DAS) as applied to multi-user MIMO (MU-MIMO) configurations. In [14], the authors investigated efficient techniques for quantizing high-dimensional channel vectors to obtain CSI feedback in FDD systems. They devised a non-coherent trellis-coded quantization (NTCQ) in which the encoding complexity varied linearly with the number of antennas.

The authors in [15] proposed a cooperative feedback scheme, in which the users can exchange their CSI with each other through device-to-device (D2D) communications. They make use of this information to compute the precoder and then feed it back to the base station (BS). In [16], the authors proposed a deep autoencoder (AE) based CSI feedback scheme capable of being tolerant to the impact of both feedback errors and feedback delays in the FDD massive MIMO system. The very large number of antennas utilized in massive MIMO results in too large feedback to acquire CSI. It is therefore necessary to compress CSI feedback signals before being processed. This was done through the compressed sensing (CS) approach, in which a sparse signal can be reconstructed from very few incoherent measurements [17]. In this respect, several CS techniques were devised. In [18], the authors used a compressive sensing (CS) technique under a significantly reduced feedback load to obtain CSI. In [19], the authors introduced a reconstruction algorithm for CS and suggested dimensionality reduction as a compression method. The proposed technique exhibited better performance in both single and multi-user scenarios. In [20], the authors analyzed the performance of different algorithms utilized in CS as being applied to band-limited signals. In a massive MIMO system, the number of radio frequency (RF) chains is limited although the number of antenna elements is large. This represents a challenging aspect in the design of CSI feedback. Deep learning (DL) techniques were devised to tackle this dilemma. In DL, the multilayered neural network (NN) is fed with a very large number of CSI data as learning inputs and outputs [21].

One of the proposed deep learning approaches was given in which the authors proposed a denoising convolutional neural network technique that could learn channel structure and then get channel information from many training data. CNN translates the extracted features of the channel into a codeword. At the decoder, this codeword is translated back into channel parameters. In [22-23], the authors introduced modified versions of the CsiNet by introducing a long short-term memory (LSTM) memory module to make use of time and frequency correlations. The authors in [24] proposed a CNN configuration that estimated the CSI, making use of both spatial and frequency correlations. The proposed configuration was applied to massive MIMO operating in a communication system that employed orthogonal frequency division multiplexing (OFDM). In [25], the authors developed a CSI scheme that combined CNN with the LSTM network and proposed an online-offline two-stage training mechanism that resulted in more stable information suitable to be applied to 5G systems. The authors in [26] proposed a multiple-moment CSI prediction scheme based on multi-hidden layer NN, in which two sample construction methods were devised to improve prediction performance and reduce complexity.

The CSI acquired at a massive MIMO base station was used by the authors in [27] to indicate the position and movement of user equipment (UE) within the cell coverage area. To reduce the complexity encountered by massive MIMO systems operating in the millimeter wave (mm-Wave) bands, a hybrid precoding scheme based on deep learning (DL) NN was introduced [28]. The algorithm was based on DL, in which the feedback network was trained offline by the massive MIMO channel data and was characterized by low feedback overhead and high feedback accuracy.

In this paper, we propose a Deep Learning-based technique for improving channel estimation accuracy and the efficiency of massive MIMO in 5G systems. The proposed technique is based on a channel state information network combined with a gated recurrent unit (CsiNet-GRU) to enhance recovery quality. Besides, it gives a reasonable trade-off between compression ratio and complexity by using time correlation in training samples and is based upon using CsiNet combined with Legendre memory unit in GRUs layers (CsiNet-LMU-GRUs). This structure can increase the input time step or memory capacity, design a Legendre Memory Unit (LMU), and combine it with a model based on the gated recurrent unit layers to expand each period's time step. The proposed techniques use the dropout method to reduce overfitting during the learning process.

The proposed technique is evaluated and compared to other similar methods that have been published in the literature. The remainder of the paper is arranged as follows: Section 2 describes the COST 2100 Data Set. Section 3 describes the utilized system model. Section 4 describes the channel state information combined with a gated recurrent unit (CsiNet-GRU) and applies the dropout method. Section 5, using the dropout method, describes the proposed channel state information gated recurrent unit combined with Legendre Memory Unit (CsiNet-LMU-GRUs). Section 6 gives the simulation and results analysis. Finally, Section 7 gives the conclusion of the paper.

2. Methodology

2.1 COST 2100 Data Set

The European Cooperation in Science and Technology (COST) 2100 channel model is a GSCM that can reproduce the stochastic properties of MIMO channels over time, frequency, and space. A multi-path component (MPC) is characterized in delay and angular domains by its delay, angle of departure (Azimuth of Departure (AoD), Elevation of Departure (EoD)), and angle of arrival (Azimuth of Arrival (AoA), Elevation of Arrival (EoA)). The MPCs with similar delays and angles are grouped into multi-path clusters.

The Matlab implementation of C2CM supports both single-link and multiple-link MIMO channel access indoor (285 MHz) and semi-urban (5.3 GHz) channel scenarios. An overview of the C2CM is presented in a detailed description of the channel model. Parameterization of the C2CM in indoor scenarios is detailed while discussing semi-urban scenarios. The C2CM is implemented in Matlab, while the semi-urban channel scenario is implemented in [29]. However, the data generated in these Matlab implementations are not presented as potential datasets to validate multi-path clustering methods and even in the well-known clustering approaches.

2.2 Utilized System Model

A single-cell FDD massive MIMO-OFDM is considered a framework, where the number of transmitting antennas N_t is very large ($\mathbf{N_t}$ >>1) at the base station (BS), and a single receiving antenna is used at the user equipment (UE).

The discrepancy between the title of the paper mentioning “MIMO” and the description focusing on a scenario with a single receiving antenna can be explained by considering that even though there is only one receiving antenna at the user equipment in this specific setup, the massive MIMO concept still applies due to the large number of transmitting antennas at the base station.

In this context, despite having only one receiving antenna at the user end, the massive MIMO configuration with numerous transmitting antennas at the base station enables spatial multiplexing and beamforming techniques that are characteristic of traditional MIMO systems. Therefore, while it may seem contradictory at first glance, the title “MIMO” in this case refers to the overarching technology employed at the base station rather than solely focusing on multiple antennas at both ends of the communication link. Consider that the OFDM incorporates several $\mathbf{N_c}$ subcarriers ($\mathbf{N_c} < \mathbf{N_t}$). The $\mathbf{n^{th}}$ subcarrier received signal as mentioned in Eq. (1) is given by:

$$y_n = \tilde{h}_n^H \, v_n \, x_n + \, z_n \tag{1}$$

where $\tilde{h}_n^H$ and $y_n \in \mathbb{C}\ \mathrm{N_t}\ \mathrm{x}\ 1$ is the channel frequency response vector and the pre-coding vector at n^{th} subcarrier, separately, x_n separately represents the image of the transmitted information, z_n is the noise or obstruction additive, and $(\cdot)^{\mathrm{H}}$ Speaks to transpose conjugate; Enhance the feedback link through the UE and BS in the FDD system, with a focus on the feedback scheme "allowing auto-encoder processing, assuming": This means that the UE should return to the BS through a feedback link in the CSI stacked in the spatial frequency domain; The total number parameter is $\mathrm{N_t}\,\tilde{\mathrm{N}}_{\mathrm{c}}$ in the feedback system. The encoder's design for dimension compression is critical. The encoders in [5] on the other hand, each used one convolutional layer and one fully linked layer. CsiNet Pro's encoder uses four convolutional layers for feature extraction and one fully connected layer for dimension compression, which is a major change. The four convolutional layers use 7 kernels to generate 16, 8, 4, and 2 feature maps, respectively as shown in Figure 2.

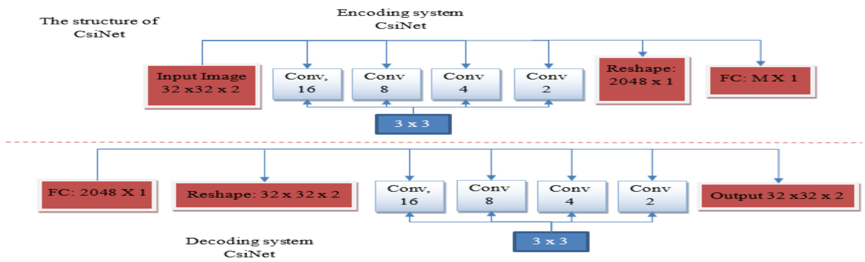

Fig. 1. Architecture of CsiNet [5]

2.3 The CsiNet-GRU Technique

Please pay attention to the feedback link included in both UE and BS in the FDD system. For this, pay attention to feedback schemes that allow autoencoder to handle. Use GRU as a gating mechanism, which is an advanced modeling of RNN, in which processing is performed to retain or forget information. The GRU (processing) operation is based on two gates, the reset gate and the update gate, being used in Figure 2. which displays the GRU block diagram. The update gate function is identical to the forget and input gate in LSTM, which determines what data is discarded and what new information is added. The reset gate is used to assess how much past knowledge is forgotten. utilized the GRU algorithm based on the Python-based deep learning library Keras configured with TensorFlow backend. The Xavier initialization was used to start all the weight matrices, while the bias matrices were set to zero. As demonstrated in Figure 16, the GRU layer's activation function is ReLU.

To reduce the loss function, adaptive moment estimation (Adam) was utilized as an optimizer. It calculates the update weights and bias based on the first-moment estimation and the second-moment estimation meanwhile (Dey and Salem), combining the advantages of AdaGrad and RMSProp. To avoid overfitting, the least squares (L2 norm) regularization and dropout approach were used. The parameters of the suggested GRU models are listed in Table 1. GRU is used for the time-dependent extraction and final reconstruction of CSI to extend the CsiNet decoder [30]. Each GRU has a memory unit that, for future prediction, can hold previously extracted information for a long time. A 3-3 convolutional layer, an M-unit dense layer for sensing, and a dense layer with $2N_c'N_t$ units should be considered to facilitate comparison with the results of the CsiNet structure given in [5] and two decoders from RefineNet for reconstruction and each RefineNet comprises four 3×3 convolutional layers with different channel sizes.

The gated recurrent unit structure allows it to adaptively capture dependencies from large data sequences without discarding information from earlier parts of the sequence. This is achieved through its gate units, like those in long short-term memory, which solve the vanishing/exploding gradient problem of traditional RNNs; These gates are responsible for regulating the information to be kept or discarded time step. Other than its internal gating mechanisms, the GRU functions like an RNN, where sequential input data is consumed by the GRU cell at each time step along with the memory, otherwise known as the hidden state. The hidden state is then re-fed into the RNN cell and the next input data is in the sequence.

Table 1
The parameters of the trained GRU model

Parameters	value
Number of neurons	32 for the GRU layer; and 1 for the dense layer
HW	14 days
Index of shuffle	100
Epoch	500
Batch size	200
Loss function	MAP
Optimizer	Adam
Activation function	ReLU
Regularization parameter of L2 norm (λ)	0.01
Dropout rate	0.1
Learning rate	0.001

The mathematical representation of the GRU structure as mentioned in Eq. (2), Eq. (3), and Eq. (4) is as follows: Fully gated unit Initially, for t = 0, the output vector is h_0 = 0.

$$z_t = \rho_g (W_z x_t + U_Z h_{t-1} + b_z) \quad (2)$$

$$r_t = \rho_g (W_r x_t + U_r h_{t-1} + b_r) \quad (3)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \phi_h (W_h x_t + U_h (r_t \odot h_{t-1}) + b_n) \quad (4)$$

where, x_t input vector, h_t output vector, z_t update gate vector, r_t reset gate vector, W, U and b denote matrices and vectors, respectively, Activation functions are ρ_g original sigmoid activation ϕ_h The initial hyperbolic tangent; Alternative activation functions can be used, provided the ρ_g (x) € [0,1]; It is possible to construct alternative forms by modifying z_t and r_t.

GRU's ability to hold on to long-term dependencies or memory stems from the gated recurrent unit cell's computations to produce the hidden state. At the same time, LSTMs have two different states passed between the cell state and hidden state, which carry the long and short-term memory, respectively GRUs only have one hidden state transferred between time steps.

This hidden state can hold both long-term and short-term dependencies at the same time due to the gating mechanisms and computations that the hidden state and input data go through.

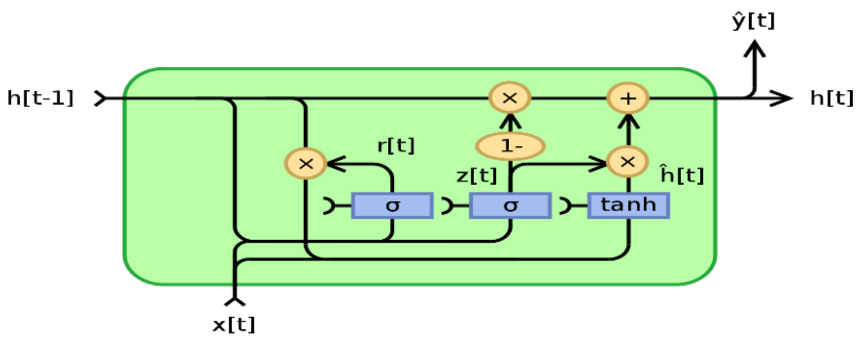

Fig. 2. The block diagram of (GRU) [30]

The channel group in an angular-interval domain as mentioned in Eq. (5) is given by:

$$\left\{H_t^{"}\right\}_{t=1}^{T} = \left\{H_1^{"}, \dots, H_t^{"}, \dots, H_T^{"}\right\} \quad (5)$$

2.4 The Proposed CsiNet-LMU-GRUs Technique Using Dropout Method

For tasks requiring lengthy time dependencies to be mastered, caption generation for machine translation, speech recognition, and several recurrent neural network (RNN) architectures have been used. LSTM and GRU are successful architectures that have an excellent example of a complex temporary relationship. mix memory cells and igniters that store and combine data non-linearly over time. Long-short-term memory is probably the most effective type of repetitive neural network (RNN) applied to many real-world problems, from captions, and generation to musical composition.

However, it breaks down when the task with learning time dependencies spans thousands of steps, making it very difficult to measure practically to improve long time windows. LMU is a new type of RNN with a memory cell with a particular weight structure mathematically derived with dynamic systems theory to preserve continuous time-scale unchanged memory.

Keras or TensorFlow expressed and trained by propagation over time, which puts a new state-of-the-art result on permuted sequential COST 2100 datasets. Difficult RNN surpasses GRU in memory capacity by several magnitudes and has been shown to scale to an input sequence spanning hundreds of millions of time steps, indicating this indoors and outdoors.

As shown in Figure 3. the key portion of the Legendre Memory Unit (LMU) [31]**.** A memory cell that orthogonalized as mentioned in Eq. (6) its input signal's continuous-time past, u(t) over a sliding length window, the cell is derived from the continuous-time delay linear transfer function, F(s) = $e^{-\theta s}$, which is best approximated by d-coupled ordinary differential equations (ODEs):-

$$\Theta\, \mathrm{m(t)} = \mathrm{A\, m(t)} + \mathrm{B\, m(t)} \quad (6)$$

where $\mathbf{m(t)} \in \mathbb{R}^{\boldsymbol{d}}$ is a d-dimensional state vector using the input vector, $\boldsymbol{x_t}$ the LMU layer structure produces the hidden state, $\boldsymbol{h_t} \in \mathbb{R}^{\boldsymbol{n}}$ Each layer retains its hidden state and memory vector. measure the non-linear function over time, state and memory communicate with each other, $\boldsymbol{m_t} \in \mathbb{R}^{\boldsymbol{d}}$ when dynamically writing it to the memory.

shall describe that the channel state information network combines with a gated recurrent unit (CsiNet-GRU) with a dropout technique; we proposed a new design, Legendre memory unit (LMU), combined with CsiNet in GRUs layers to improve the best performance in the modeling technique depending on the deep temporal hierarchy of LMUs.

The difference is the number of time steps in the temporal prediction of the memory unit and the number of the unit cells according to the Legendre memory unit model's structure. It applies the dropout technique randomly of the modeling after each learning processing modeling with the convolutional three layers of the network depending on GRUs layers.

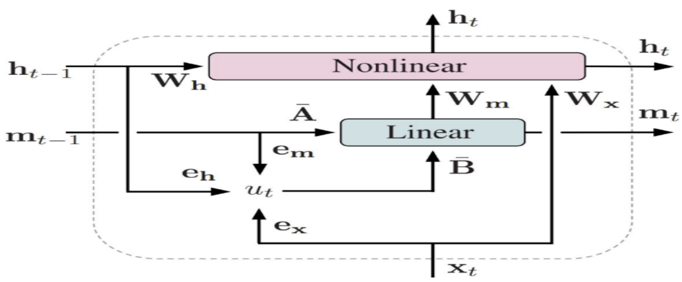

Fig. 3. Time-unrolled LMU layer [31]

The tasks were chosen to approve the LMU's determination while highlighting its key preferences: it can learn temporal dependencies over T = 100,000 time-steps and converge quickly due to the use of non-saturating memory units. LMU is a rare example of a first-principles RNN dynamics derivation with such desirable properties, meaning that nerve activity is consistent with such a derivative and shows peak deep-depth activity.

It serves as a reminder of the importance of addressing various aspects of neural architecture and integrating math, signal processing, and in-depth learning instruments, and applying it to CsiNet technology at GRU levels that enhance efficiency by achieving 97.15% LMU surpasses state-of-the-art technology. Although parameters are used, 15 percent test precision is used, making essential observations in this lesson on LMU performance:

The LMU model further addresses gradient loss and explosion, which is generally associated with learning RNNs using a cellular structure derived from the first principles for projecting a continuous signal into orthogonal dimensions **d**.

By mapping the Legendre polynomials into a linear memory cell, LMUs can efficiently handle long-term dependencies and converge quickly to learn complicated functions; few internal state variables are used and ensure that the gradient flows through the continuous history characteristics. LMU is a recent innovation that achieves peak memory capacity while ensuring energy efficiency, making it particularly suitable for the task of chaotic time-series predictions.

It learns quickly and surpasses modern art in 1000 epochs, it does more than remember input outside the baseline; it needs to use the hidden nonlinearities to perform some valuable calculations through memory. This method has a solution that reflects, by Legendre, the sliding windows of polynomials. It will be tested in COST 2100 data sets in many indoor and outdoor activities, depending on the increasing number of memory-only problems time-steps.

Experiments show promising results with Legendre Memory Units (LMUs) that achieve the best and most consistent performance in CsiNet-LMU-GRUs depending on indicators parameters: run time, NMSE, correlation coefficient, and accuracy compared to the other compressed sensing methods.

To reduce feedback overhead, exploit the following observations: - Observation 1 (angular-delay domain sparsity): $\boldsymbol{H_t}$ can be transformed into an approximately sparsified matrix $\boldsymbol{H_t'}$ in the angular-delay domain via 2D discrete Fourier transform (2D-DFT) by $\boldsymbol{H_t'} = \boldsymbol{F_d}\ \boldsymbol{H_t}\ \boldsymbol{F_a}$, where $\boldsymbol{F_d} \in \mathbb{C}^{N_c \times N_c}$ and $\boldsymbol{F_a} \in \mathbb{C}^{N_t \times N_t}$ are two DFT matrices.

First, due to limited multipath time delay, performing DFT on frequency domain channel vectors (i.e., column vectors of $\boldsymbol{H_t}$) can transform $\boldsymbol{H_t}$ into a sparse matrix in the delay domain, with only the first $\boldsymbol{N_c'}$ **(<** $\boldsymbol{N_c}$**)** rows have distinct non-zero values. Secondly, the channel matrix is sparse in a defined angle domain by performing DFT on spatial domain channel vectors (i.e., row vectors of $\boldsymbol{H_t}$) if the number of transmit antennas, $\boldsymbol{N_t} \rightarrow +\infty$, is very large.

Usually, $\boldsymbol{H_t'}$ is only approximately sparse for finite $\boldsymbol{N_t}$ which challenges conventional CS methods. Therefore, propose a DL-based feedback architecture without sparsity prior constraint. perform sparsity transformation to decrease parameter overhead and training complexity. retain the first $\boldsymbol{N_c'}$non-zero rows and truncate $\boldsymbol{H_t'}$ to a $\boldsymbol{N_c'} \times \boldsymbol{N_t}$ matrix, $\boldsymbol{H_t''}$ which reduces the total number of parameters for feedback to $\boldsymbol{N} = \boldsymbol{N_c'}\boldsymbol{N_t}$.

Observation 2 (correlation within coherence time): UE motion during communication results in a Doppler spread, i.e., time-varying characteristics of wireless channels with the maximum movement speed denoted as **v**, as mentioned in Eq. (7) coherence time can be calculated as: -

$$\Delta t = \frac{C}{2\, v\, fo} \tag{7}$$

where $\boldsymbol{f_0}$ is the carrier frequency, and **c** is the speed of light. The CSI within **Δt** is considered correlated with one other. Therefore, instead of independently recovering CSI, the BS can combine the feedback and previous channel information to enhance the subsequent reconstruction.

Designed a decoder with a memory that can extract time correlation from the previously recovered channel matrices, $\hat{\mathrm{H}}_{\boldsymbol{1}}^{''},\ldots\ldots,\hat{\mathrm{H}}_{\boldsymbol{t-1}}^{''}$ and combine them with the received $\boldsymbol{S_t}$ for the current reconstruction, $\hat{\mathrm{H}}_{\boldsymbol{t}}^{''} = \boldsymbol{f_{de}}\ (\boldsymbol{S_t};\ \hat{\mathrm{H}}_{\boldsymbol{1}}^{''}\ , \ldots,\ \hat{\mathrm{H}}_{\boldsymbol{t-1}}^{''})$, where $\mathbf{1 \leq t \leq T}$. Then, inverse 2D-DFT is performed to obtain the original spatial frequency channel matrix.

The CsiNet demonstrates remarkable performance in CSI sensing and reconstruction. However, the resolution degrades at low CR because it only focuses on angular-delay domain sparsity (Observation 1) and ignores the time correlation (Observation 2) of time-varying massive MIMO channels.

Motivated by RCNN, which excels in extracting spatial-temporal features for video representation, extend CsiNet-GRU with LMU to improve CR and recovery quality trade-off. introduce the multi-CR strategy to implement variable CRs on different channel matrices.

The proposed CsiNet-LMU-GRUs are illustrated in Figure 7. with CsiNet-GRU. Our model includes the following two steps: angular-delay domain feature extraction, correlation representation, and final reconstruction. Each LMU-GRU has an inherent memory unit that, for future prediction, can hold previously extracted information for a long time.

A 3 × 3 convolutional layer an M-unit dense layer for sensing, and a dense layer with $\boldsymbol{2N_c'N_t}$ units should be considered to facilitate comparison with the results of the CsiNet structure given in (C. Wen, W. Shih and S. Jin) and two decoders from RefineNet for reconstruction. As shown in Figure 4, each RefineNet comprises four 3 × 3 convolutional layers with different channel sizes. The CsiNet decoder's output generates a sequence, and the length of every sequence is **T**, which is then fed into a three-layer LMU-GRU.

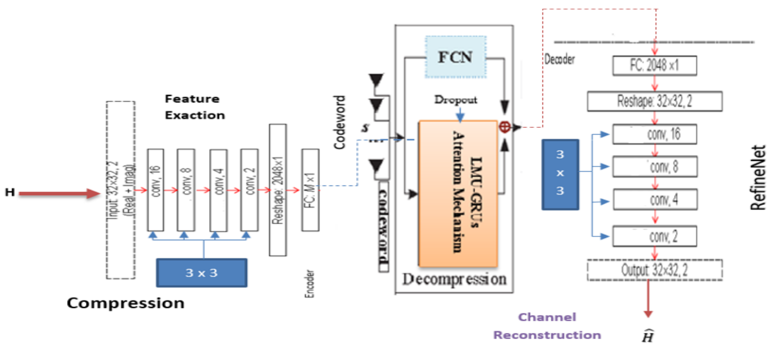

Fig. 4. The structure of CsiNet-LMU-GRUs based neural network which is used to be auto encoder-decoder

Combine LMU-GRUs and CsiNet in a typical neural network for MIMO channel feedback in this section, which contains two steps: encoding and decoding, as illustrated in Figure 4. A Reduced CsiNet-LMU-GRUs is also used to reduce training parameters and network complexity. This subsection proposes CsiNet-LMU-GRUs, a CSI feedback structure based on LMU-GRUs, to reduce the problem of CSI feedback overhead in FDD massive MIMO systems. The length, width, and number of feature maps are $\boldsymbol{N_1}$ **x** $\boldsymbol{N_2}$ **x** $\boldsymbol{N_3}$ in this structure. The real and imaginary components of the truncated $\boldsymbol{H \in \mathbb{C}^{N_c \times N_t}}$ are separated and concatenated into $\boldsymbol{H'} \in \mathbb{R}^{\boldsymbol{N_c \times N_t\, x\, 2}}$. at UE to reduce computing complexity and the processed $\boldsymbol{H'}$ can then be passed into an encoder, which will convert it to codewords. Feature extraction and feature compression are the two elements of a single encoder. A convolutional layer, batch normalization layer, and leaky rectified linear unit are used to extract features (Leaky Relu). Two three-dimensional kernels are used to perform the convolutional process. Then, in the feature compression, rearrange the two N_t x N_c feature matrices into one feature vector, $\mathfrak{l} \in R^{Nx1}$, where $N = 2N_tN_c$ and then $\mathfrak{l} \in R^{Nx1}$ is fed into two paths: A FCN that is linear: The FCN is a jump connection that converts the vector $\mathfrak{l}$ into the vector $s_1 \in R^{Mx1}$ and this network can speed up convergence and solve the problem of vanishing gradients [5].

Legendre Memory Unit (LMU): - The LMU is specifically designed to enhance memory capacity by orthogonalizing its continuous-time history using Legendre polynomials. It achieves this by solving a set of coupled ordinary differential equations (ODEs), allowing it to maintain a more structured representation of temporal information.

Extended Memory: The LMU can effectively manage longer sequences due to its mathematical formulation, which allows it to capture and utilize information from much earlier time steps without degradation.

Orthogonalization: By employing orthogonal polynomial basis functions, the LMU minimizes interference between different memory states, enhancing clarity and retention of relevant information.

Performance Metrics: - Research indicates that LMUs outperform traditional RNN architectures in tasks requiring long-term memory retention, demonstrating superior performance in scenarios where maintaining historical context is critical.

When comparing the memory capacities of LMUs and GRUs: Long-Term Dependencies: LMUs excel in scenarios requiring retention over extended periods due to their mathematical foundation that supports longer temporal spans.

Efficiency vs. Capacity: While GRUs are efficient and effective for many applications, they may face challenges with very long sequences where LMUs demonstrate clear advantages.

In summary, while both architectures provide significant improvements over traditional RNNs, the LMU offers enhanced capabilities for managing long-term dependencies through its unique design principles rooted in polynomial mathematics.

2.5 The Dropout Technique

Dropout is a regularization method when training the network, as shown in Figure 5., it may not exclude the input and loop connections to the GRU unit from activation and weight updates as shown in Figure 6. The framework adopts the dropout regularization approach used in the neural network to minimize overfitting and boost the CsiNet-GRU structure's efficiency.

Indicated that the effect on the activation of "downstream neurons" during the forward process would be temporarily removed, and no "backward propagation to neurons" weight update would be applied [32]. Dropout is a technique where randomly selected neurons are ignored during training, “dropped-out” randomly. This means that their contribution to downstream neurons’ activation is temporally removed on the forward pass, and no weight updates are applied to the neuron on the backward pass. Dropout may be implemented on any hidden layers in the network. The visible or input layer and the term “dropout” refer to dropping out units (hidden and visible) in a neural network; Training Phase: For each hidden layer, for each training sample, for each iteration, ignore (zero out) a random fraction, p, of nodes (and corresponding activations) and **Testing Phase:** Use all activations, but reduce them by a factor p (to account for the missing activations during training).

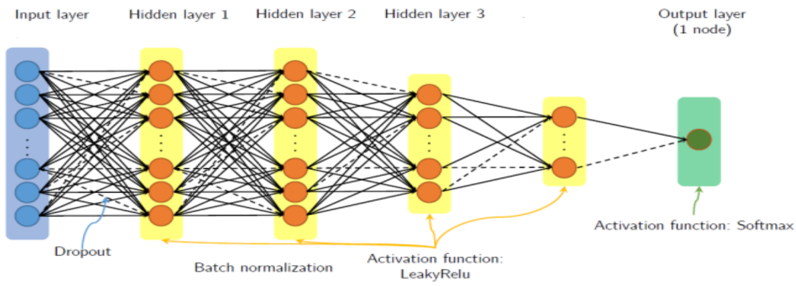

Fig. 5. Neural Network with Dropout Architecture [32]

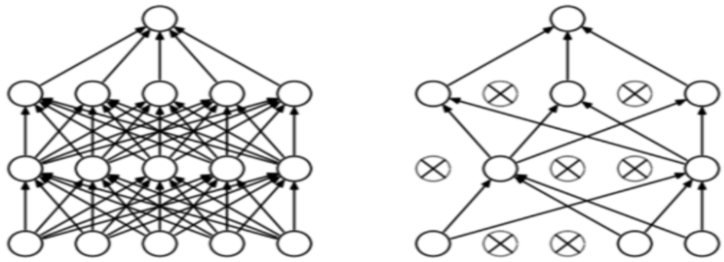

Fig. 6: a) Standard neural network b) neural network after applying dropout

Dropout in some observations forces a neural network to learn more robust features useful in conjunction with many other neurons' random subsets, Dropout roughly doubles the number of iterations required to converge; however, each epoch's training time is less and in a testing phase, the entire network is considered, and a factor of p reduces each activation.

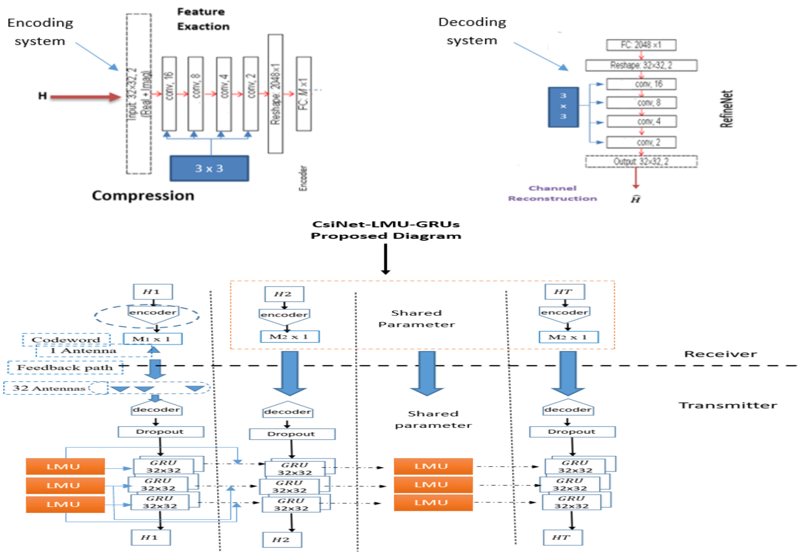

Fig. 7. The design of the proposed modeling Legendre Memory Unit combined with CsiNet in GRUs layers (CsiNet-LMU-GRUs)

3. Simulation and Results Analysis

The COST 2100 channel model is a stochastic channel model based on geometry (GSCM) that can replicate stochastic properties over time, frequency, and space of multi-link channels with multiple linkages (MIMO), as was illustrated in [29]. The extension of the model also threatens parameterization and validation attempts.

It is also clear that the efficient implementation and sustainable use of the COST 2100 model requires more sophisticated channel evaluation methods and enough parameterization and validation model measurement campaigns. Multiple Input Multiple Output (MIMO) is an enabling technology that can meet the increasing demand for faster and more efficient wireless channel transmission.

MIMO technology research has covered the entire field in recent years, from theory to implementation. The technology is now used as a key component in standards such as Third Generation Partnership Project Long Term Evolution (3GPP LTE) WiMAX 802.16e Inside.

simulate MIMO channels and produce training samples; the COST 2100 model is used using a uniform linear array, setting the MIMO-OFDM system to operate on a 20 MHz bandwidth (ULA). Table 2. lists the parameters used in both indoor and outdoor channel scenarios. Data sets are generated by creating different start locations for indoor and outdoor scenarios at random.

Perform simulations with CR values such that less than 1/4 is compressed to the first channel $\boldsymbol{H}_{\boldsymbol{1}}^{"}$ The training, validation, and testing sets given in Table 2. are provided. Some startup parameters (epochs 500 - 1000, learning rate 0.001, and batch size 200) have been preloaded by CsiNet, as shown in Table 2.

explore the effects of varying GRU depth on network performance. For all experiments, use the Adam optimizer with a learning rate of $\mathbf{10^{-3}}$. For the GRU-only networks in the ablation study, train for 500 epochs. For CsiNet-GRU, pertain CsiNet at each compression ratio for 600 epochs, and then initialize the CsiNet at each timeslot of CsiNet-GRU with the pertained weights before training for 500 epochs.

In order also to have a fair comparison with the CsiNet, adopting the exact Adaptive Moment Estimation (ADAM) as the optimization algorithm and using mean squared error (MSE) as the loss function, as mentioned in Eq. (8) which is defined as,

$$L(\theta) = \frac{1}{M}\sum_{m=1}^{M}\sum_{t=1}^{T}\|f(H_t^{"};\theta) - H_t^{"}\|_2^2 \tag{8}$$

where $\boldsymbol{M}$ is the total number of samples in the training set and $\|\cdot\|_2$ Present the Euclidean norm.

Multiple CR CsiNet encoders are deployed at UE, whereas the CsiNet decoders and GRUs are deployed at the BS. In the beginning, $\boldsymbol{H}_{\boldsymbol{1}}^{"}$ is compressed with high CR at the UE recovered by a high-CR CsiNet decoder and initialized by the BS's GRUs. In the subsequent time step $\mathbf{t}$ $(\boldsymbol{2} \leq \boldsymbol{t} \leq \boldsymbol{T})$, $\boldsymbol{H}_{\boldsymbol{t}}^{"}$ is transformed into a lower-dimensional codeword S_t at the user equipment UE, which is expected to contain the learned correlation information. The lower-dimensional codeword, S_t is then concatenated with the first one S_1 and inversely transformed by the GRUs at the base station (BS).

As for all optimization problems, particular algorithms' performance is highly dependent on the problem details and hyper-parameters, while having not attempted fine-tuning the hyper-parameters of individual optimization methods, kept as many hyper-parameters as possible constant to better allow for comparison.

Simulation is carried out through m-files implemented on a Python 3.7 platform. Comparing the proposed architecture's performance with the previous similar modeling approaches of CSI with different deep learning approaches, namely Conv-LSTM CsiNet, LASSO, TVAL3, and CsiNet.

Utilizing the default setup in the open-source codes of the previously mentioned techniques for reproduction, note that in the CsiNet, the slight distinction between datasets is considered.

LASSO achieves outstanding results despite using simple sparsity priors. TVAL3 is a minimum total variation system with high computing efficiency and excellent recovery quality. In the function extraction and recovery modules, Convolutional-LSTM CsiNet uses RNN and depth-wise separable convolution, respectively.

Run the proposed technique CsiNet-LMU-GRUs on Colaboratory (python) according to zero configuration required, free access to GPUs, and easy sharing training and testing of the CsiNet, Conv-LSTM CsiNet, CsiNet-GRU, and CsiNet-LMU-GRUs on python colab editor.

Comparisons are investigated based on the normalized mean square error, the cosine similarity, accuracy, and the best run-time in the indoor and outdoor channel and the complexity incorporated.

The Normalized Mean Square Error (NMSE) as mentioned in Eq. (9) utilized for comparisons quantifies the difference between the input $\{H_t\}_{t=1}^{T}$ and the output $\{\hat{\mathrm{H}}_t\}_{t=1}^{T}$ in both proposed techniques CsiNet-GRU and CsiNet-LMU-GRUs and is given by:

$$NMSE = \mathbb{E}\left\{\frac{1}{T}\sum_{t=1}^{T}\left\|H_t^{"} - \widehat{H}_t^{"}\right\|_2^2 / \left\|H_t^{"}\right\|_2^2\right\} \tag{9}$$

To measure the degree of similarity between the actual channel $h_{n.t}$ and the estimated channel value $\hat{h}_{n.t}$ of the n^{th}subcarrier at time t., used the cosine similarity coefficient ρ, as mentioned in Eq. (10) in CsiNet-GRU, which is given as:

$$\rho = \mathbb{E}\left\{\frac{1}{T}\frac{1}{N_c}\sum_{t=1}^{T}\sum_{n=1}^{N_c}\frac{\left|\hat{h}_{n.t}^{H}\ h_{n.t}\right|}{\left\|\hat{h}_{n.t}\right\|_2\ \left\|h_{n.t}\right\|_2}\right\} \tag{10}$$

To measure the degree of similarity as mentioned in Eq. (11) between the actual channel $\boldsymbol{h}_{\boldsymbol{n},\boldsymbol{t}}$ and the estimated channel value $\widehat{\widetilde{\boldsymbol{h}}}_{\boldsymbol{n.t}}$ of the $\boldsymbol{n^{th}}$subcarrier at time t., using the cosine similarity coefficient $\boldsymbol{\rho}$, in CsiNet-LMU-GRUs which is given as:

$$\rho = \mathbb{E}\left\{\frac{1}{T}\frac{1}{\widetilde{N}_c}\sum_{t=1}^{T}\sum_{n=1}^{\widetilde{N}_c}\frac{\left|\hat{\tilde{h}}_{n.t}^{H}\ \tilde{h}_{n.t}\right|}{\left\|\hat{\tilde{h}}_{n.t}\right\|_2\ \left\|\tilde{h}_{n.t}\right\|_2}\right\} \tag{11}$$

introduce a new parameter for comparison, which is called accuracy. Defining it as the ratio of several correct predictions to the total number of input samples, means accuracy is the ratio of the recovered channel vector to the original channel vector, as mentioned in Eq. (12) so the accuracy in CsiNet-LMU-GRUs is defined as:

$$Accuracy = \mathbb{E}\left\{\frac{1}{T}\frac{1}{\widetilde{N}_c}\sum_{t=1}^{T}\sum_{n=1}^{\widetilde{N}_c}\frac{\left|\hat{\tilde{h}}_{n.t}^{H}\ \right|}{\left\|h_{n.t}\right\|_2}\right\} \tag{12}$$

Table 2
COST 2100 Model Data – Sets and System, Parameters

MIMO OFDM	20 MHz bandwidth
$\boldsymbol{H}$	32 × 32
$\boldsymbol{N_t}$	32 Antennas
$\boldsymbol{N_C}$	1024 Subcarriers
Training Samples	100,000
Validation Samples	30,000
Testing Samples	20,000
Epochs	500 - 1000
Learning Rate	0.001
Batch size	200

∂t	0.04 s	
T	10 s	
CR	4, 16, 32, 64	
Channel Model	**Indoor Scenario**	**Outdoor Scenario**
Frequency, f **Speed, v** Δt	Pico, 5.3 GHz, 0.0036 0.0036 km/h 30 s	Rural, 300 MHz 4.24 km/h 0.56 s

3.1 Comparison of results in different techniques

compare the channel state information network CsiNet, CsiNet-GRU, and CsiNet-LMU-GRUs in indoor and outdoor scenarios as shown in the following figures to prove that the proposed techniques improve best results in different indicators parameter systems with the dropout method. run GRU with LMU techniques in different time scales and compare the performance and run-time in each epoch during the iteration of the model when running the GRU without the LMU technique.

Figure 8. and Figure 9. illustrate the relation between CR and NMSE in cases of indoor and outdoor scenarios, respectively, for all structures. It is clear from Figure 9. that the proposed CsiNet-GRU and CsiNet-LMU-GRUs have the lowest NMSE, whereas, in Figure 10, it has the lowest NMSE among others except that of Conv-LSTM CsiNet at values of CR > 20.

Figure 10. and Figure 11. illustrate the relation between the CR and accuracy in indoor and outdoor scenarios, respectively, for all structures. The CsiNet-GRU and CsiNet-LMU-GRUs outperform the other designs with higher accuracy values observed at lower CR values.

Figure 12. and Figure 13. illustrate the relation between the cosine similarity (ρ) and CR in indoor and outdoor scenarios, respectively, for all structures. Again, the proposed CsiNet-GRU and CsiNet-LMU-GRUs outperform the other systems, and besides, exhibit a near-linear performance with the lowest slope.

Figure 14. shows the reconstruction results of the proposed technique compared to the other modeling techniques, namely LASSO, TVAL3, CsiNet, Conv-LSTM CsiNet, CsiNet-GRU, CsiNet-LSTM and CsiNet-LMU-GRUs in indoor Picocellular scenario as an example. The figure represents the average performance at different CRs, reflecting on the reconstructed images to utilize the other techniques.

Figure 15. Pseudo-gray plots of an original channel generated by the COST 2100 model in the outdoor scenario, showing real part, imagined part and absolute values, and Absolute values of reconstructed images, which are performed by different methods on the original channel at different CRs.

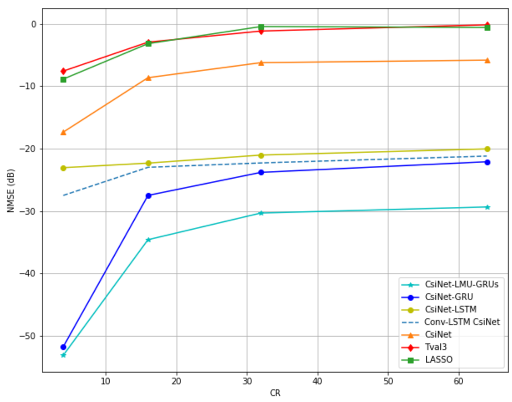

Fig. 8. NMSE (dB) performance comparison INDOOR scenario

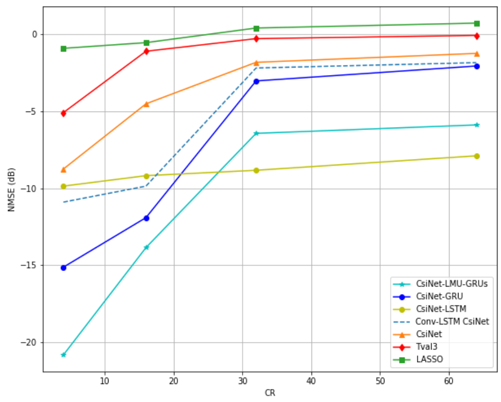

Fig. 9. NMSE (dB) performance comparison OUTDOOR scenario

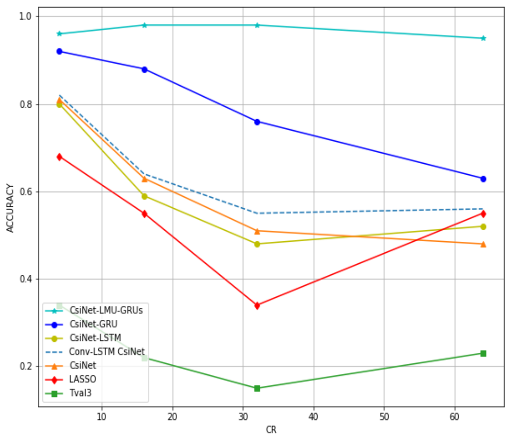

Fig. 10. Accuracy performance comparison INDOOR scenario

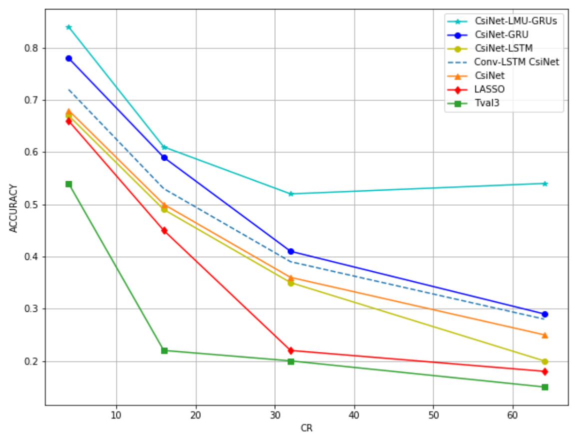

Fig. 11. Accuracy performance comparison OUTDOOR scenario

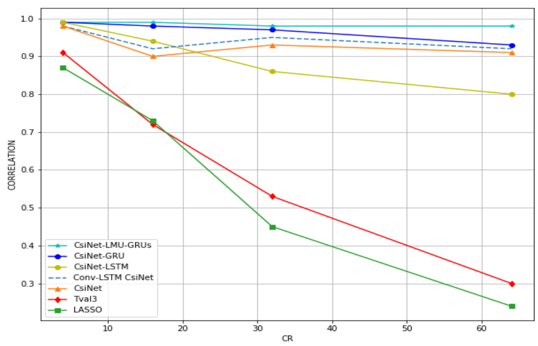

Fig. 12. ρ performance comparison INDOOR scenario

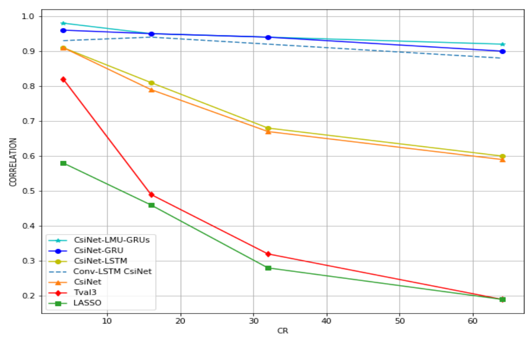

Fig. 13. ρ performance comparison OUTDOOR scenario

Proposed a real-time and end-to-end CSI feedback framework by extending the DL-based CsiNet with LSTM. CsiNet-LSTM achieves a remarkable trade-off among CR, recovery quality, and complexity by utilizing the time correlation and structure properties of time-varying massive MIMO channels.

Thus, CsiNet-LSTM outperforms CsiNet and CS-based methods. believe that this framework has the potential for practical deployment on real systems.

Explore the effects of varying LSTM depth on network performance. For all experiments, use the Adam optimizer with a learning rate of 10^{-3}. For the LSTM-only networks in the ablation study, train for 500 epochs. For CsiNet-LSTM, pertain CsiNet at each compression ratio for 600 epochs, and then initialize the CsiNet at each timeslot of CsiNet-LSTM with the pertained weights before training for 500 epochs.

Finally, comparing the final results between CsiNet-LSTM, CsiNet-GRU, and CsiNet-LMU-GRUs using the Dropout method technique, the final results showed that all the indicators parameter performance proves that the channel state information network with the Dropout technique in a complex neural network is the best and it showed in all the values of the accuracy, normalized mean square error and correlation coefficients, Table 3 shows that all the results compared with all the previous techniques used in massive MIMO channel state information network (CsiNet) using the Dropout method to reduce the overfitting of the modeling system.

CsiNet, Conv-LSTM CsiNet, CsiNet-GRU, and CsiNet-LMU-GRUs continue to offer acceptable correlation coefficients (ρ) at low CRs, where CS-based methods fail to work. It is quite remarkable that the proposed CsiNet-GRU and CsiNet-LMU-GRUs techniques outperform the others. The performance comparison in the proposed CsiNet-LMU-GRUs to other available methods is given in Table 3. where corresponding values of NMSE, ρ, and accuracy are calculated for different values of γ for indoor and outdoor scenarios. All techniques have better performances in the indoor scenario than the outdoor one. From Table 3, It is notable that the CsiNet strategies considerably outperform the other CS-based methods (LASSO and TVAL3) [33-35]. CsiNet, CsiNet-GRU, and CsiNet-LMU-GRUs continue to offer the highest cosine similarity values at low CRs, where CS-based methods fail to work. However, the proposed CsiNet-LMU-GRUs outperform the channel state information network (CsiNet).

As far as the NMSE is concerned, the CsiNet-GRU and CsiNet-LMU-GRUs achieve the lowest values of all compressed ratios (CRs), especially when CR is low. Notably, CsiNet-GRU and CsiNet-LMU-GRUs are characterized by very short-run periods compared to LASSO and TVAL3 techniques. However, compared with the other CsiNet method, the proposed CsiNet-LMU-GRUs slightly loses time efficiency.

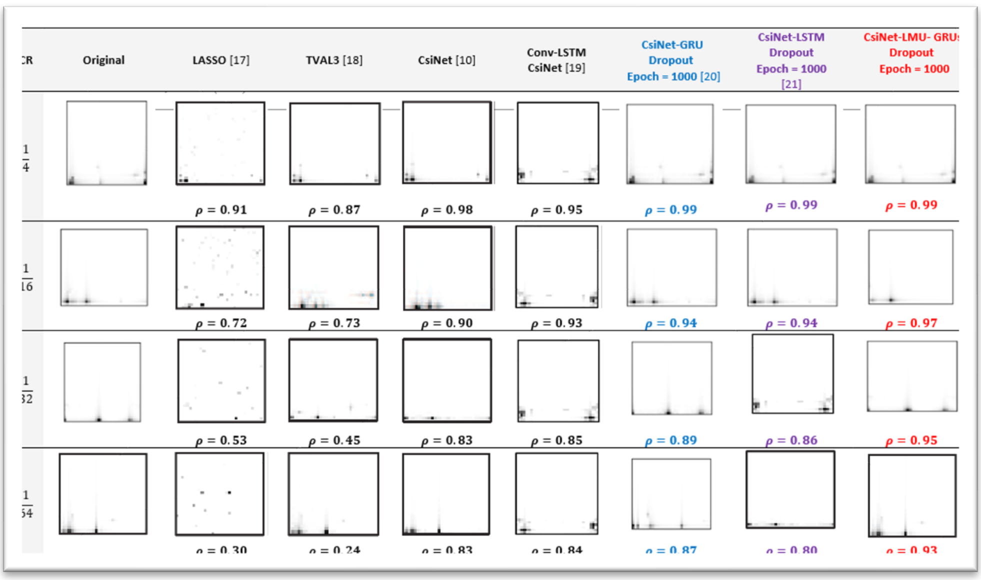

Fig. 14. Reconstruction images for CR in CS algorithms in an indoor scenario

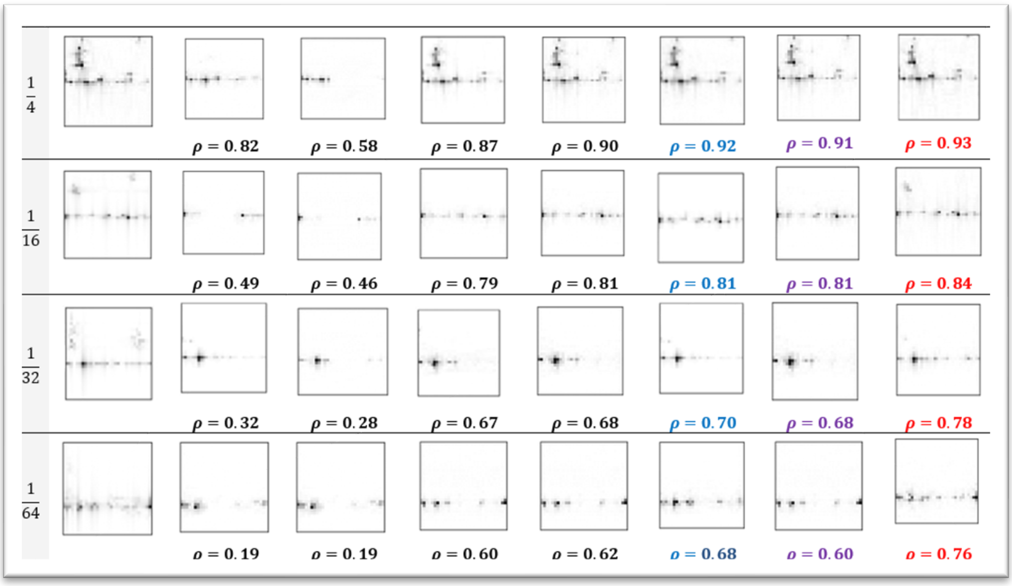

Fig. 15. Reconstruction images for CR in CS algorithms in an outdoor scenario

Indicate the complexity that results from adding the GRU units; It is quite notable that although there is a considerable added complexity due to the introduction of the GRU layers, the run time is still comparable to that of the CsiNet. applied the dropout method, which has compensated for the expected excessive run time. On the other COST 2100 datasets, demonstrate a consistent advantage of the LMU over GRUs in performance and parameter counts. While these are not hitting state-of-the-art accuracies, mainly because both models are much smaller than networks that do, these experiments demonstrate significantly superior performance with 60-1000x fewer parameters, suggesting a general trend of parameter efficiency.

Finally, one of this approach's main contributions is that it provides a means to train networks that are recurrent during inference in a Legendre memory unit manner and provides a much better mapping of the training problem to current commodity hardware. It also provides a practical inference advantage. The final model can be run recurrently, thus providing a natural means of processing data online with minimal latency and efficient memory use.

CsiNet-GRU and CsiNet-LMU-GRUs outperform CsiNet, especially when compression ratios are low. Furthermore, CsiNet-GRU and CsiNet-LMU-GRUs can recover the CSI with better accuracy and maintain some features that may be lost during the CsiNet feedback procedure. CsiNet-LMU-GRUs outperforms previous approaches in terms of recovery accuracy and reconstruction quality, according to experimental data. propose that our architecture has the potential to be used in real-world FDD MIMO systems.

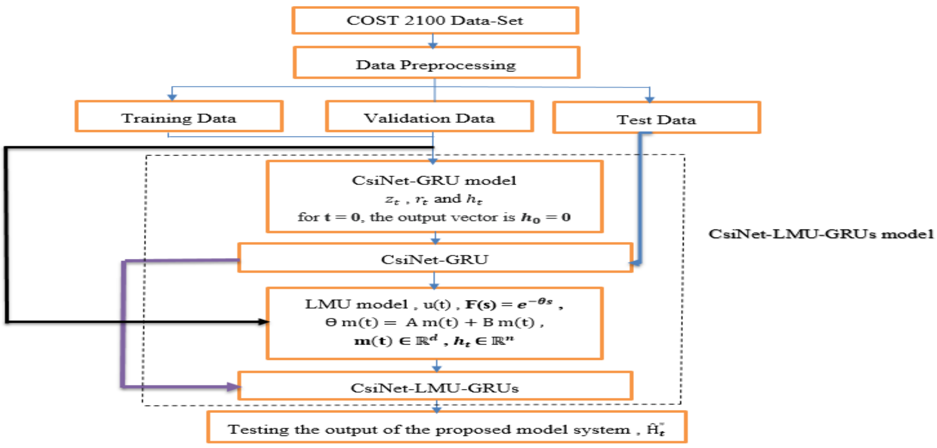

Fig. 16. The overall workflow of the proposed CsiNet-LMU-GRUs model

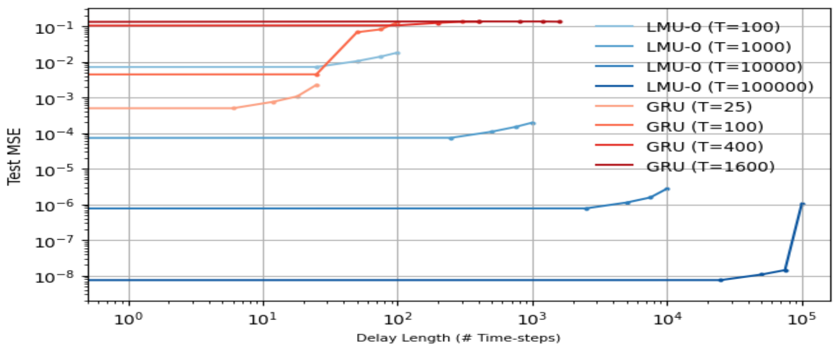

Fig. 17. Memory Capacity of Recurrent Architectures

From Figure 17, the comparison of GRUs and LMUs is shown in a scale between the input and output levels. A model trained at a different window length corresponds to T and evaluates through the window at five different delay lengths with only 105 internal state variables.

Achieved this by maintaining a compressed representation with a time level of Δt = 1/T of the 10 Hz band minimal input signal. In Figure 16. Gated Recurrent Unit (GRU) took different time per second in every epoch iteration during the modeling processing on the training data as follows: GRU took 0.046356 sec/epoch on T=25, GRU took 0.054484 sec/epoch on T=100, GRU took 0.119503 sec/epoch on T=400 and GRU took 0.373979 sec/epoch on T=1600.

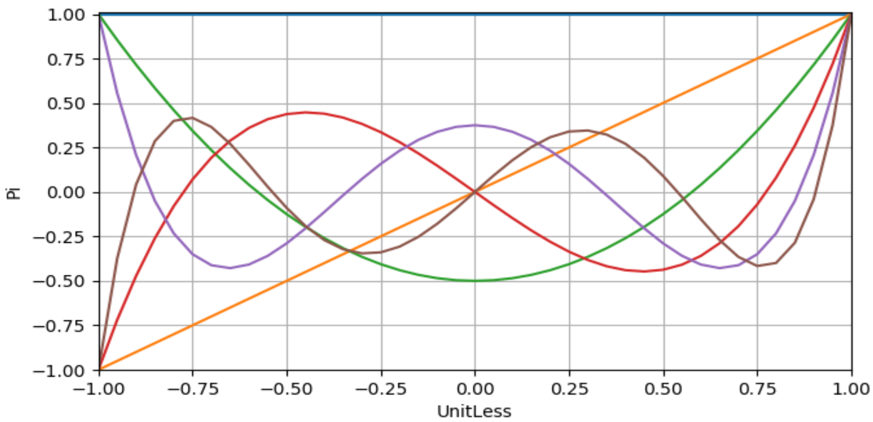

Fig. 18. Shifted Legendre polynomials

Figure 18. The mapping of Legendre polynomials (d= 12) shows the mapping, as a linear combination of these invariant polynomials of size, the memory of the LMU reflects the whole sliding window of the input background, and an increasing number of the dimensions supports the memory against the time scale of high-frequency inputs.

Table3
Comparison of results between the proposed framework and other available Ones

		γ	LASSO [33]	TVAL3 [34]	CsiNet [5]	Conv-LSTM CsiNet [35]	CsiNet-GRU Dropout / 1000 Epoch [30]	CsiNet-LSTM Dropout /1000 Epoch [22]	CsiNet-LMU-GRUs Dropout / 1000 Epoch
Indoor	NMSE Epoch= 1000	1/4	-7.59	-8.87	-17.36	-27.5	-51.73	-23.06	-53.10
		1/16	-2.96	-3.2	-8.65	-23	-27.3	-22.33	-34.59
		1/32	-1.18	-0.46	-6.24	-22.3	-23.81	-21.04	-30.32
		1/64	-0.18	-0.6	-5.84	-21.2	-22.11	-20.05	-29.35
	ρ Epoch= 1000	1/4	0.91	0.87	0.98	0.95	0.99	0.99	0.99
		1/16	0.72	0.73	0.90	0.93	0.94	0.94	0.97
		1/32	0.53	0.45	0.83	0.85	0.89	0.86	0.95
		1/64	0.30	0.24	0.83	0.84	0.87	0.80	0.93
	Accuracy Epoch= 1000	1/4	0.68	0.34	0.81	0.82	0.84	0.80	0.90
		1/16	0.55	0.22	0.60	0.60	0.62	0.59	0.68
		1/32	0.34	0.15	0.51	0.51	0.52	0.48	0.58
		1/64	0.55	0.23	0.48	0.53	0.54	0.52	0.60
	Run time Epoch= 1000	1/4	0.345	0.314	0.0001	0.0001	0.0003	0.0003	0.000096
		1/16	0.555	0.314	0.0001	0.0001	0.0003	0.0003	0.0000103
		1/32	0.604	0.286	0.0001	0.0001	0.0003	0.0003	0.0000120
		1/64	0.708	0.285	0.0001	0.0001	0.0003	0.0003	0.0000114
Outdoor	NMSE Epoch= 1000	1/4	-5.08	-0.9	-8.75	-10.9	-15.13	-9.86	-20.84
		1/16	-1.09	-0.53	-4.51	-9.86	-11.91	-9.18	-13.86
		1/32	-0.27	0.42	-2.81	-9.18	-3.02	-8.83	-6.43
		1/64	-0.06	0.74	-1.93	-8.83	-2.05	-7.89	-5.88
	ρ Epoch= 1000	1/4	0.82	0.58	0.87	0.90	0.92	0.91	0.93
		1/16	0.49	0.46	0.79	0.81	0.81	0.81	0.84
		1/32	0.32	0.28	0.67	0.68	0.70	0.68	0.78
		1/64	0.19	0.19	0.60	0.62	0.68	0.60	0.76
	Accuracy Epoch= 1000	1/4	0.66	0.54	0.68	0.70	0.71	0.67	0.76
		1/16	0.45	0.22	0.49	0.49	0.51	0.49	0.55
		1/32	0.20	0.20	0.36	0.36	0.38	0.35	0.45
		1/64	0.18	0.15	0.26	0.26	0.27	0.20	0.40

4. Conclusion

This paper proposed a channel state information (CSI) feedback network by extending the DL-based CsiNet technique to incorporate GRUs and LMU over GRU layers using the dropout method. The proposed CsiNet-LMU-GRUs technique achieved the lowest NMSE, the best cosine similarity coefficient, and the best accuracy compared to other CS-based and CSI-based methods. The work is motivated by the active use of the recurrent convolutional neural network (RCNN) model in image reconstruction using a channel state information network. The basic concept is to extract spatial features and inter-frame correlation using the COST 2100 model to simulate time-varying MIMO channels and acquire training samples.

References

[1] Almamori, Aqiel, and Seshadri Mohan. "Improved MMSE Channel Estimation in Massive MIMO System with a Method for the Prediction of Channel Correlation Matrix." In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2018. https://doi.org/10.1109/CCWC.2018.8301699.

[2] Alrabeiah, Muhammad, and Ahmed Alkhateeb. "Deep Learning for TDD and FDD Massive MIMO: Mapping Channels in Space and Frequency." In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2019. https://doi.org/10.1109/IEEECONF44664.2019.9048929.

[3] Boas, B. V., N. Fonseca, A. Klautau, and N. González-Prelcic. "Low Complexity Transform Coding for Millimeter Wave MIMO CSI Compression." In *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, 1582–1586. Pacific Grove, CA: IEEE, 2018. https://doi.org/10.1109/ACSSC.2018.8645142.

[4] Luo, Changqing, Jiayi Ji, Qian Wang, Xuhui Chen, and Peng Li. "Channel State Information Prediction for 5G Wireless Communications: A Deep Learning Approach." *IEEE Transactions on Network Science and Engineering* 7, no. 1 (2018): 227–236. https://doi.org/10.1109/TNSE.2018.2848960.

[5] Wen, Chao-Kai, Wan-Ting Shih, and Shi Jin. "Deep Learning for Massive MIMO CSI Feedback." *IEEE Wireless Communications Letters* 7, no. 5 (2018): 748–751. https://doi.org/10.1109/LWC.2018.2818160.

[6] Morocho Cayamcela, M. E., and W. Lim. "Artificial Intelligence in 5G Technology: A Survey." In *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, 860–865. Jeju, Korea (South): IEEE, 2018. https://doi.org/10.1109/ICTC.2018.8539642.

[7] Donoho, D. L., A. Maleki, and A. Montanari. "Message Passing Algorithms for Compressed Sensing: I. Motivation and Construction." In *2010 IEEE Information Theory Workshop on Information Theory (ITW 2010, Cairo)*, 1–5. Cairo, Egypt: IEEE, 2010. https://doi.org/10.1109/ITWKSPS.2010.5503193.

[8] Dey, Rahul, and Fathi M. Salem. "Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks." In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2017. https://doi.org/10.1109/MWSCAS.2017.8053243.

[9] Fan, Cheng, Jiayuan Wang, Wenjie Gang, and Shenghan Li. "Assessment of Deep Recurrent Neural Network-Based Strategies for Short-Term Building Energy Predictions." *Applied Energy* 236 (2019): 700–710. https://doi.org/10.1016/j.apenergy.2018.12.004.

[10] Gerstoft, P., C. F. Mecklenbräuker, W. Seong, and M. Bianco. "Introduction to Special Issue on Compressive Sensing in Acoustics." *Journal of the Acoustical Society of America* 143, no. 6 (2018): 3731. https://doi.org/10.1121/1.5043089.

[11] He, H., C.-K. Wen, S. Jin, and G. Y. Li. "Deep Learning-Based Channel Estimation for Beamspace mmWave Massive MIMO Systems." *IEEE Wireless Communications Letters* 7, no. 5 (2018): 852–855. https://doi.org/10.1109/LWC.2018.2832128.

[12] Xie, H., F. Gao, and S. Jin. "An Overview of Low-Rank Channel Estimation for Massive MIMO Systems." *IEEE Access* 4 (2016): 7313–7321. https://doi.org/10.1109/ACCESS.2016.2623772.

[13] Schwarz, Stefan, Robert W. Heath, and Markus Rupp. "Single-User MIMO versus Multi-User MIMO in Distributed Antenna Systems with Limited Feedback." *EURASIP Journal on Advances in Signal Processing* 2013 (2013): 54. https://doi.org/10.1186/1687-6180-2013-54.

[14] Choi, Junil, Z. Chance, D. J. Love, and U. Madhow. "Noncoherent Trellis Coded Quantization: A Practical Limited Feedback Technique for Massive MIMO Systems." *IEEE Transactions on Communications* 61, no. 12 (2013): 5016–5029. https://doi.org/10.1109/TCOMM.2013.111413.130379.

[15] Chen, J., H. Yin, L. Cottatellucci, and D. Gesbert. "Feedback Mechanisms for FDD Massive MIMO With D2D-Based Limited CSI Sharing." *IEEE Transactions on Wireless Communications* 16, no. 8 (2017): 5162–5175. https://doi.org/10.1109/TWC.2017.2706279.

[16] Jang, Youngrok, G. Kong, M. Jung, S. Choi, and I. Kim. "Deep Autoencoder Based CSI Feedback with Feedback Errors and Feedback Delay in FDD Massive MIMO Systems." *IEEE Wireless Communications Letters* 8, no. 3 (2019): 833–836. https://doi.org/10.1109/LWC.2019.2895039.

[17] Orović, Irena, Vladan Papić, Cornel Ioana, Xiumei Li, and Srdjan Stanković. "Compressive Sensing in Signal Processing: Algorithms and Transform Domain Formulations." *Mathematical Problems in Engineering* 2016 (2016): 1–16. https://doi.org/10.1155/2016/7616393.

[18] Kuo, P.-H., H. T. Kung, and P.-A. Ting. "Compressive Sensing Based Channel Feedback Protocols for Spatially-Correlated Massive Antenna Arrays." In *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, 492–497. Paris, France: IEEE, 2012. https://doi.org/10.1109/WCNC.2012.6214417.

[19] Sim, M. S., J. Park, C.-B. Chae, and R. W. Heath. "Compressed Channel Feedback for Correlated Massive MIMO Systems." *Journal of Communications and Networks* 18, no. 1 (2016): 95–104. https://doi.org/10.1109/JCN.2016.000012.

[20] Mihajlović, R., M. Scekic, A. Draganic, and S. Stankovic. "An Analysis of CS Algorithms Efficiency for Sparse Communication Signals Reconstruction." In *2014 3rd Mediterranean Conference on Embedded Computing (MECO)*, 221–224. Budva, Montenegro: IEEE, 2014. https://doi.org/10.1109/MECO.2014.6862700.

[21] Jiang, Z., S. Chen, A. F. Molisch, R. Vannithamby, S. Zhou, and Z. Niu. "Exploiting Wireless Channel State Information Structures Beyond Linear Correlations: A Deep Learning Approach." *IEEE Communications Magazine* 57, no. 3 (2019): 28–34. https://doi.org/10.1109/MCOM.2019.1800581.

[22] Wang, T., C.-K. Wen, S. Jin, and G. Y. Li. "Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels." *IEEE Wireless Communications Letters* 8, no. 2 (2019): 416–419. https://doi.org/10.1109/LWC.2018.2874264.

[23] Lu, Chao, W. Xu, H. Shen, J. Zhu, and K. Wang. "MIMO Channel Information Feedback Using Deep Recurrent Network." *IEEE Communications Letters* 23, no. 1 (2018): 188–191. https://doi.org/10.1109/LCOMM.2018.2882829.

[24] Dong, P., H. Zhang, G. Y. Li, N. NaderiAlizadeh, and I. S. Gaspar. "Deep CNN for Wideband Mmwave Massive Mimo Channel Estimation Using Frequency Correlation." In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4529–4533. Brighton, UK: IEEE, 2019. https://doi.org/10.1109/ICASSP.2019.8682819.

[25] Luo, Changqing, Jiayi Ji, Qian Wang, Xuhui Chen, and Peng Li. "Channel State Information Prediction for 5G Wireless Communications: A Deep Learning Approach." *IEEE Transactions on Network Science and Engineering* 7, no. 1 (2018): 227–236. https://doi.org/10.1109/TNSE.2018.2848960.

[26] Liao, Run-Fa, Hong Wen, Jinsong Wu, Huanhuan Song, Fei Pan, and Lian Dong. "The Rayleigh Fading Channel Prediction via Deep Learning." *Wireless Communications and Mobile Computing* 2018, no. 1 (2018): 6497340. https://doi.org/10.1155/2018/6497340.

[27] Studer, C., S. Medjkouh, E. Gönültaş, T. Goldstein, and O. Tirkkonen. "Channel Charting: Locating Users Within the Radio Environment Using Channel State Information." *IEEE Access* 6 (2018): 47682–47698. https://doi.org/10.1109/ACCESS.2018.2866979.

[28] Huang, Hongji, Y. Song, J. Yang, G. Gui, and F. Adachi. "Deep-Learning-Based Millimeter-Wave Massive MIMO for Hybrid Precoding." *IEEE Transactions on Vehicular Technology* 68, no. 3 (2019): 3027–3032. https://doi.org/10.1109/TVT.2019.2893928.

[29] Liu, L., C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, and P. De Doncker. "The COST 2100 MIMO Channel Model." *IEEE Wireless Communications* 19, no. 6 (2012): 92–99. https://doi.org/10.1109/MWC.2012.6393523.

[30] Helmy, H. M. N., A. M. Ghaleb, T. H. El-Fishawy, and E. S. M. El-Rabaie. "Performance Enhancement of Massive MIMO Using Deep Learning-Based Channel Estimation." *IOP Conference Series: Materials Science and Engineering* 1051, no. 1 (2021): 012029. https://doi.org/10.1088/1757-899X/1051/1/012029.

[31] Diógenes, Gabriela Kaori, V. S. B. Silva, L. B. S. Rezende, G. L. S. Silva, A. S. B. Oliveira, and R. F. S. Silva. "Live Births Prediction Using Legendre Memory Unit: A Case Study for the Health Regions of Goiás." In *2023 IEEE 36th International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE, 2023. https://doi.org/10.1109/CBMS58004.2023.00239.

[32] Liang, Jianglin, and Ruifang Liu. "Stacked Denoising Autoencoder and Dropout Together to Prevent Overfitting in Deep Neural Network." In *2015 8th International Congress on Image and Signal Processing (CISP)*. IEEE, 2015. https://doi.org/10.1109/CISP.2015.7407967.

[33] Ranstam, Jonas, and Jonathan A. Cook. "LASSO Regression." *Journal of British Surgery* 105, no. 10 (2018): 1348. https://doi.org/10.1002/bjs.10895.

[34] Yin, Fei, X. Liu, Y. Zhang, and J. Zhang. "High Precision Reconstruction for Compressed Femtosecond Dynamics Images Based on the TVAL3 Algorithm." *Optical Materials Express* 12, no. 11 (2022): 4435–4443. https://doi.org/10.1364/OME.468475.

[35] Li, Xiangyi, and Huaming Wu. "Spatio-Temporal Representation with Deep Neural Recurrent Network in MIMO CSI Feedback." *IEEE Wireless Communications Letters* 9, no. 5 (2020): 653–657. https://doi.org/10.1109/LWC.2020.2964550.