Spatio-Temporal Representation with Deep Neural Recurrent Network in MIMO CSI Feedback

Xiangyi Li and Huaming Wu, *Member, IEEE*

***Abstract*—In multiple-input multiple-output (MIMO) systems, it is crucial of utilizing the available channel state information (CSI) at the transmitter for precoding to improve the performance of frequency division duplex (FDD) networks. One of the main challenges is to compress a large amount of CSI in CSI feedback transmission in massive MIMO systems. In this paper, we propose a deep learning (DL)-based approach that uses a deep recurrent neural network (RNN) to learn temporal correlation and adopts depthwise separable convolution to shrink the model. The feature extraction module is also elaborately devised by studying decoupled spatio-temporal feature representations in different structures. Experimental results demonstrate that the proposed approach outperforms existing DL-based methods in terms of recovery quality and accuracy, which can also achieve remarkable robustness at low compression ratio (CR).**

***Index Terms*—MIMO, CSI Feedback, FDD, Recurrent Neural Network, Spatio-Temporal Feature.**

I. Introduction

THE technology of massive multiple-input multiple-output (MIMO), which was first pointed out in the early twentieth century, has become increasingly crucial in new generation mobile wireless communications (5G or B5G). The system uses multiple antennas as multiple transmitters at the base station (BS) and receivers at user equipment (UE) to realize the multipath transmitting, which can double the channel capacity without increasing spectrum resources or antenna transmit power. A growing number of studies [1]–[4] have shown the significance of utilizing the channel state information (CSI) feedback at the transmitter to gain the improvement of MIMO systems. In a frequency division duplex (FDD) network [5], UE can estimate the downlink CSI, which is then fed back to the BS to perform precoding for the next signal.

In fact, the uplink CSI feedback process is not an easy task in massive MIMO systems [6], due to a large number of antennas at the BS, resulting in high CSI feedback and huge computational complexity. In order to reduce the CSI feedback overhead, many methods and technologies have been proposed. Some compressive sensing (CS)-based approaches may not fit in real world CSI feedback systems and perform poorly in CSI compression due to the harsh preconditions. Recent studies have shown that applying DL to address the nonlinear problems or challenges in wireless communications can boost the quality of CSI feedback compression [4]. Wen *et al.* [7] proposed an autoencoder network called CsiNet, which used several neural network (NN) layers as an encoder instead of the CS model to compress CSI as well as a decoder to recover the original CSI. Furthermore, they also put forward another network called CsiNet-LSTM [2], which extended CsiNet with three RNN layers to show the benefits of exploring temporal channel correlation. Another paralleled work, called RecCsiNet [8], applied RNN in both the encoder and decoder to reduce errors in CSI compression and decompression. Both of them can improve the performance of the CsiNet network to some extent and outperform state-of-the-art CS methods.

In this paper, we design a new architecture of deep NN in CSI feedback compression, which also takes advantage of RNN. Based on the RecCsiNet architecture, we retain its structure of feature compression and decompression modules, and further improve the feature extraction by applying RNN and separating feature extraction in the spatial and temporal domains. In addition, motivated by MobileNet [9] that used depthwise separable convolutions to build lightweight deep NN, which we substitute them for standard convolutions to enhance the quality of RefineNet [7]. The main contributions are summarized as follows:

- We propose a novel and effective CSI sensing and recovery mechanism in the FDD MIMO system, referred to as ConvlstmCsiNet, which takes advantage of the memory characteristic of RNN in modules of feature extraction, compression and decompression, respectively. Moreover, we adopt depthwise separable convolutions in feature recovery to reduce the size of the model and interact information between channels.
- We further refine ConvlstmCsiNet in the feature extraction module by exploring the spatial-temporal feature representation that decouples a convolution in the spatial and temporal domains. Experimental results demonstrate that the improved ConvlstmCsiNet achieves the highest recovery quality at different compression ratios (CRs) compared to the state-of-the-art DL-based models.

II. CSI Feedback System

We consider an FDD massive MIMO downlink system with N_t transmitting antennas at the BS and a single receiving antenna at each UE, which is operated in OFDM with $\tilde{N}_c$ subcarriers. The received signal carried by the n^{th} ($n = 1, 2, \cdots, \tilde{N}_c$) subcarrier can be given as:

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

Y. Li and H. Wu are with the Center for Applied Mathematics, Tianjin University, Tianjin 300072, China (e-mail: xiangyi_li@tju.edu.cn; whming@tju.edu.cn).

This work was supported by the National Key R & D Program of China (2018YFC0809800), the National Natural Science Foundation of China (61801325), and the Huawei Innovation Research Program (HO2018085138). (Corresponding author: Huaming Wu.)

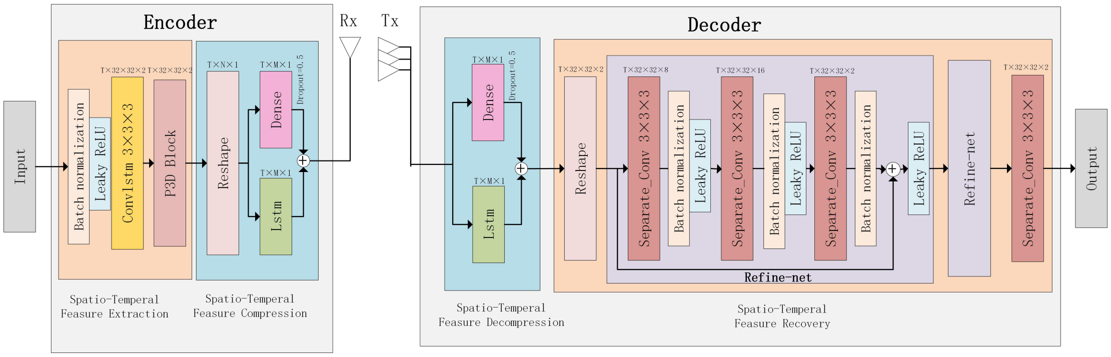


Fig. 1: The architecture of ConvlstmCsiNet with P3D block

where $\tilde{\mathbf{h}}_n \in \mathbb{C}^{N_t}$, $\mathbf{v}_n \in \mathbb{C}^{N_t}$, $x_n \in \mathbb{C}$ and $z_n \in \mathbb{C}$ denote the instantaneous channel vector, the precoding vector, the modulated transmit symbol and the additional noise at the n^{th} subcarrier, respectively. Then the CSI matrix can be denoted as:

$$\tilde{\mathbf{H}} = [\tilde{\mathbf{h}}_1, \tilde{\mathbf{h}}_2, \cdots, \tilde{\mathbf{h}}_{\tilde{N}_c}] \in \mathbb{C}^{N_t \times \tilde{N}_c} \tag{2}$$

We assume that each UE can acquire the estimation of channel response $\tilde{\mathbf{H}}$, which is then fed back to the BS to help the BS to generate the precoding vector $\mathbf{v}_n$. The process of CSI feedback from the UE to the BS, involving the actual required CSI compression, is the main goal of our research. Before being transmitted to the BS, the CSI matrix requires two pretreatments:

- $\tilde{\mathbf{H}}$ is supposed to be sparse in the angular-delay domain after undergoing a 2D discrete Fourier transform (DFT) operation.
- In the delay domain, most of the elements in $\tilde{\mathbf{H}}$ are zeros except for the first few non-zero columns, because the time delay between multipath arrivals around the straight path lies within a small finite time period. Therefore, the first N_c ($N_c < \tilde{N}_c$) nonzero columns can be retained, while the rest are removed, and the new $N_t \times N_c$ sized CSI matrix is represented as $\mathbf{H}$.

According to [10], we assume that the channel matrix $\mathbf{H}$ remains fixed for a given OFDM symbol and its associated precoding vector, however, it varies from time to time based on a state-space model. Denote that $\mathbf{H}_t = [\mathbf{h}_{1,t}, \mathbf{h}_{2,t}, \cdots, \mathbf{h}_{N_c,t}] \in \mathbb{C}^{N_t \times N_c}$ is the instantaneous CSI at t^{th} time step, and then $\mathbf{H}_{t+1}$ at next time step can be expressed as:

$$\mathbf{H}_{t+1} = \mathbf{F} \cdot \mathbf{H}_t + \mathbf{G} \cdot \mathbf{u}_t \tag{3}$$

where $\mathbf{u}_t \in \mathbb{C}^{N_t \times N_c}$ is the additive noise that each element $u_t^{(i,j)} \sim N(0, \sigma_u^2)$, and $\mathbf{F}, \mathbf{G} \in \mathbb{C}^{N_t \times N_t}$ are the weight square matrices, which are assumed to be available to the receiver. For convenience, we set $\mathbf{F} = (1-\alpha^2)\mathbf{I}$ and $\mathbf{G} = \alpha^2\mathbf{I}$ by introducing a new parameter α, which depicts the correlation between adjacent CSI matrices. So this sequence of time-varying channel matrix is defined as: $\{\mathbf{H}_t\}_{t=1}^T = \{\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_T\}$.

During transmission, $\{\mathbf{H}_t\}_{t=1}^T$ is separated into a real part and an imaginary part to reduce the computational complexity, where all elements in the matrix are turned into real numbers and normalized within $[0, 1]$. With the help of DFT and truncation operations, the number of feedback parameters should be reduced from $\tilde{N} = 2 \times \tilde{N}_c \times N_t$ to $N = 2 \times N_c \times N_t$, which still remains a large number of parameters in massive MIMO systems and information compression is required during the transmission procedure. The model consists of an encoder at the UE to convert a CSI matrix $\mathbf{H}_t$ of size N into a compressed M-dimensional ($M < N$) codeword $\mathbf{s}_t$, as well as a decoder at the BS to make the compressed vector $\mathbf{s}_t$ transform back to the original CSI matrix. The data compression ratio is $\gamma = M/N$. Once the BS completes the recovery of $\mathbf{H}_t$, i.e., $\hat{\mathbf{H}}_t$, it outputs the final matrix $\hat{\tilde{\mathbf{H}}}_t$ by adding zero columns and performing inverse DFT.

III. Proposed ConvlstmCsiNet with P3D blocks

The proposed ConvlstmCsiNet is illustrated in Fig. 1. It includes an encoder at the UE and a decoder at the BS. The encoder is divided into two modules, i.e., *feature extraction* and *feature compression*; and the decoder consists of *feature decompression* and *feature recovery* modules, where RefineNet unit is employed in the feature recovery module.

Different types of network layers are colored and each layer has the output shape on the top, marked by $T \times H \times W \times C$ or $T \times L \times C$, where T, H, W, C and L denote the time step of RNN, height, width, channel numbers of feature maps, and codeword length, respectively. After the DFT and truncation operations, the CSI matrix $\mathbf{H}$ is then fed into this CSI feedback autoencoder with the input shape of $T \times 32 \times 32 \times 2$ ($H = N_t = 32$, $W = N_c = 32$), where two channels represent the real and imaginary parts of $\mathbf{H}$. The output remains the same shape as the input.

A. ConvlstmCsiNet

1) RNN in Feature Extraction: On the basis of CsiNet [7], we refine the feature extraction module by adding a convolutional long short-term memory (ConvLSTM) [11] layer before the convolution, and adopt the memory function of RNN to learn the temporal correlation from the inputs of previous time steps as well as compress the temporal redundancy. Therefore, it can help the convolution to capture more useful temporal information in feature extraction.

ConvLSTM is a variant of LSTM, which is proposed in RNN to solve the problem of time sequence gradient disappearing with the increase of calculation time. The main change is that the weight calculation is switched from linear operation to convolution operation, which helps it not only inherit the ability of LSTM and capture the temporal correlation, but also depict the detailed local information in image features like CNN, simultaneously.

The main structure of ConvLSTM is shown in Fig. 2. It has the ability to remove or add information to the cell state through three well-designed gates, i.e., forget gate, input gate and output gate, including a sigmoid activation layer and a convolutional operation. Both the input x_n at n^{th} time step and the $(n-1)^{th}$ time step output h_{n-1} are undergone with this operation group four times, i.e., once in the forget and the output gates and twice in the input gate, respectively. Since convolution operations require far fewer parameters than linear operations, ConvLSTM can help reduce the size of the model, especially for large input sizes. Suppose that the input amount of ConvLSTM and LSTM are the same, i.e., ConvLSTM takes in features with a shape of (H, W, C) and LSTM takes in vectors with a length of N, i.e., $H \times W \times C$, and both their outputs remain the same shape with their inputs, and the kernel size is $k \times k$ in ConvLSTM. Then the LSTM operation requires $4TN(2N+1)$ parameters while the ConvLSTM operation only requires $4T(k^2 \times C \times C + 1)$ ones, which are much fewer than LSTM's.

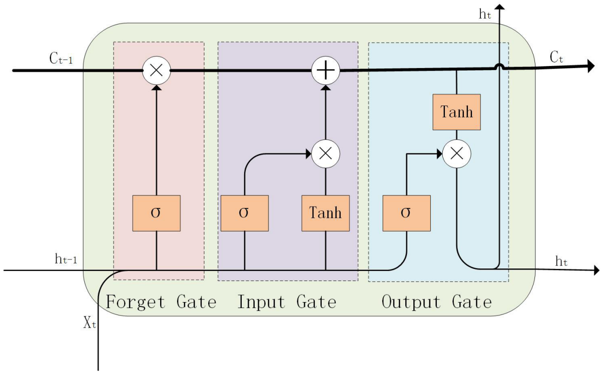
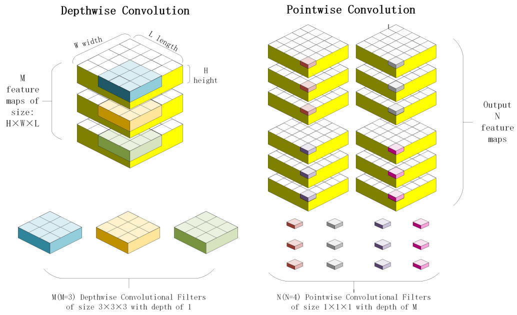


Fig. 2: The structure of three gates in ConvLSTM [11]

The symmetric feature compression and feature decompression modules refer to RecCsiNet [8], which has achieved higher accuracy than PR-RecCsiNet [8] or CsiNet-LSTM [2]. It uses two parallel row structures, i.e., the fully-connected (FC) layer and the LSTM layer, to compress the reshaped N-length vector into a M-length codeword, simultaneously. Then we put the merged codeword as the output of the encoder, and decompress it back to N-length with the symmetric structure, which will be reshaped into two 32×32 sized features, serving as a rough estimation of the real and imaginary parts of $\mathbf{H}$. During the feedback transmission, the feedback channel is assumed to be perfect enough to transmit the compressed codeword without any damage or loss.

Although ConvLSTM has so many advantages, we retain LSTM instead of completely replacing it with ConvLSTM since LSTM can perform better in terms of overall information interaction due to its FC operation in weight calculations, thus is more suitable for feature compression, while ConvLSTM is more adaptable for depicting local detailed information.

2) Depthwise Separable Convolution in Feature Recovery: RefineNet in CsiNet [7] is adopted as the basic structure. Each RefineNet block has three $3 \times 3 \times 3$ Conv3D layers, which are cascaded together one by one, outputting 8, 16 and 2 feature maps, respectively. The feature recovery module helps to refine the primary rough estimation of $\mathbf{H}$ with two RefineNet blocks and the results in CsiNet have testified that two blocks are sufficient to recover the CSI matrix and more blocks will lead to parameter redundancy. After two RefineNet blocks follow a $3 \times 3 \times 3$ Conv3D layer and a sigmoid activation layer, which outputs the final result of the recovered $\mathbf{H}$, including its real and imaginary parts.



Depthwise Convolution
Pointwise Convolution
W width
L length
H height
M feature maps of size: H×W×L
Output N feature maps
M(M=3) Depthwise Convolutional Filters of size 3×3×3 with depth of 1
N(N=4) Pointwise Convolutional Filters of size 1×1×1 with depth of M

Fig. 3: Filters of depthwise separable convolution

While in this module, all standard convolutions in the feature recovery module are replaced by a new type of convolutional layer, i.e., depthwise separable convolution [9], referred to as DS-Conv. This substitution not only reduces the number of parameters, but also helps the RefineNet achieve better performance and higher recovery accuracy. According to MobileNet, it can be divided into two steps: *depthwise convolution* and *pointwise convolution*, the kernels of which are shown in Fig. 3.

It is assumed that the original $3 \times 3 \times 3$ Conv3D accepts M input feature maps and outputs N feature maps. Depthwise convolution is a set of convolutions, each of which is responsible for one feature map separately, so there are M $3 \times 3 \times 3$ 1-depth depthwise convolution filters to output M feature maps. While pointwise convolution is a M-depth $1 \times 1 \times 1$ convolution to deal with M feature maps obtained from depthwise convolution and outputs N feature maps. The first step is mainly responsible for capturing features in each channel, while the second step is for the dimensions of ascending and descending channels, as well as for information integration and interaction across channels, which helps the convolution to better understanding the correlation between different channels. The parameter number of DS-Conv3D is $(M \times 3^3 + M \times N)/M \times 3^3 \times N$ times of the Conv3D, so that DS-Conv3D can also help to reduce the parameter size of the feature recovery module to a certain extent. In addition, due to the large use of pointwise convolution, highly optimized matrix multiplications, such as GEMM, can be used directly to complete them without the pre-processing operation of im2col, which greatly improves the operational efficiency.

B. Decoupled Spatial-Temporal Feature Extraction in ConvlstmCsiNet

For further refinement of ConvlstmCsiNet, we focus on the spatial-temporal feature representation in the feature extraction module. Notice that ConvLSTM first extracts the spacial features in the cell and then cycles the cell to form a time series, indicating a certain degree of independence between extracting spatial features and temporal features in a sense. To better cooperate with ConvLSTM, we rise study on this independence in a 3D convolution to show how this representation can affect the NN's performance.

Inspired by [12], we replace the convolutional layer with Pseudo-3D (P3D) in ConvlstmCsiNet to perform feature extraction. The key idea of P3D is to capture features in the temporal and spatial domains, respectively. Suppose we have 3D convolutional filters of size $T_d \times S_d \times S_d$ (T_d and S_d denote temporal depth and spatial depth, respectively), which can be naturally decoupled into $1 \times S_d \times S_d$ convolutional filters equivalent to 2D convolutions in spatial domain and $T_d \times 1 \times 1$ convolutional filters equivalent to 1D convolutions on temporal domain. This block replaces the standard convolutional layer with two filters in a cascaded or paralleled manner. In this way, both the number of parameters and computational complexity can be reduced. Moreover, separating spatial and temporal feature extraction in P3D blocks acquires higher efficiency than a standard 3D convolution especially when combining with the ConvLSTM, which can maintain the relative independence of spatial-temporal features, thereby eliminating the redundancy more efficiently and accurately in spatial and temporal domain, respectively.

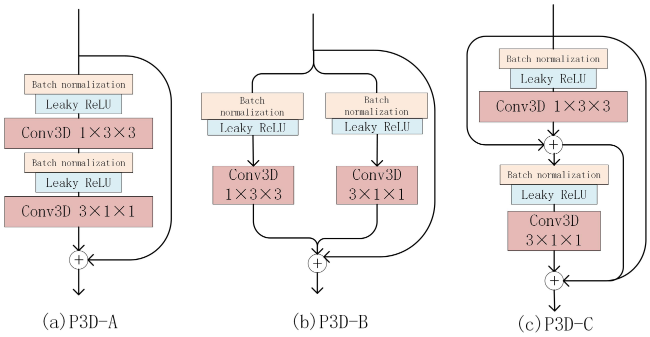


Fig. 4: Three designs of P3D block

Considering whether the temporal and spatial filters should directly or indirectly influence each other or the final output, three designs of P3D blocks are proposed, which are shown in Fig. 4. The skip connection structure of ResNet [13] is also used here, which can directly pass the data flow to subsequent layers and lead the model degenerating into a shallow network, thus helping to ease the optimization and improve the robustness of the NN by skipping those unnecessary layers. For regularization, we adopt the idea of pre-activation structure [14] that the batch normalization (BN) layer followed with an activation layer of leaky ReLU is placed before all weighted layers (e.g., convolutional layer), which has the impacts that the optimization is further eased and the regularization of the model is improved.

The model complexity analysis is depicted in Table I, where the number of parameters and MACCs[1] stand for space and time complexity, respectively.

TABLE I: The number of parameters and MACCs

	CR	1/4	1/8	1/16	1/32
Params	CsiNet	2,103,904	1,055,072	530,656	268,448
	RecCsiNet	28,331,104	22,300,512	19,481,824	18,121,632
	ConvlstmCsiNet	28,326,904	22,296,312	19,477,624	18,117,432
	ConvlstmCsiNet_A/B/C	28,326,854	22,296,262	19,477,574	18,117,382
MACCs	CsiNet	21,659,648	5,668,864	3,571,712	2,523,136
	RecCsiNet	153,059,328	128,942,080	117,669,888	112,230,400
	ConvlstmCsiNet	121,708,544	97,591,296	86,319,104	80,879,616
	ConvlstmCsiNet_A/B/C	121,462,784	97,345,536	86,073,344	80,633,856

In Table I, CsiNet has the lowest number of parameters and MACCs, at the cost of low recovery quality. RecCsiNet improves the NN's performance by modeling a more complicated structure, and the strong augment in space and time complexity is primarily caused by LSTM, where four dense layers are laid in each LSTM cell. Our approaches, ConvlstmCsiNet or ConvlstmCsiNet with P3D block, can achieve much higher accuracy and robustness than RecCsiNet without increasing the NN's complexity. Although the proposed methods devise a dedicated structure based on the structure of RecCsiNet, e.g., ConvLSTM layer, in order to achieve more improvement in model capacity, the number of parameters and MACCs are reduced due to the alleviative effects on the NN's complexity of depthwise separable convolutions as well as P3D blocks. Especially in MACCs, the operation size of ConvlstmCsiNet can be reduced to 121 M, which is 21% lower than RecCsiNet at 1/4 CR. As the number of antennas in BS grows, this shrinking effect can be enlarged exponentially.

Based on ConvlstmCsiNet, we refer the three newly proposed models as ConvlstmCsiNet-A, ConvlstmCsiNet-B and ConvlstmCsiNet-C, where P3D-A, P3D-B or P3D-C blocks replace the convolution in feature compression module, respectively. Then ConvlstmCsiNet is used to highlight the effect of P3D in feature extraction by comparing to those with P3D blocks. We assume that the models are trained on a fully differentiable channel model and interpret the whole CSI feedback communication system as an auto-encoder [15], [16]. Define the network as an autoencoder function f of the input $\mathbf{H}_t$, then the output can be expressed as:

$$\begin{aligned}\hat{\mathbf{H}}_t &:= f(\{\mathbf{H}_k\}_{k=1}^{t}; \Theta) \\ &= f_{dec}(f_{enc}(\{\mathbf{H}_k\}_{k=1}^{t}; \Theta_{enc}); \Theta_{dec}) \qquad (4)\end{aligned}$$

where Θ is the whole parameters and $f(\cdot)$ represents the function of the network. f_{dec}, f_{enc}, Θ_{dec} and Θ_{enc} denote the maps and parameters of the decoder and encoder, respectively.

All networks are trained end-to-end by updating parameters in the procedure of minimizing the mean squared error (MSE) loss function using the ADAM algorithm, which can be given as follows:

$$\begin{aligned}L(\Theta) &= \frac{1}{MT} \sum_{m=1}^{M} \sum_{t=1}^{T} \|f(\mathbf{H}_{m,t}; \Theta) - \mathbf{H}_{m,t}\|_F^2 \\ &= \frac{1}{MT} \sum_{m=1}^{M} \sum_{t=1}^{T} \sum_{i=1}^{N_t} \sum_{j=1}^{N_c} \left| f(\mathbf{H}_{m,t}^{(i,j)}; \Theta) - \mathbf{H}_{m,t}^{(i,j)} \right|^2 \qquad (5)\end{aligned}$$

where $\| \cdot \|_F$ is the Frobenius norm, T and M denote the number of recurrent steps and the total number of examples in the training data, respectively.

[1] MACC: multiply-accumulate operations. A multiplication operation and an additive operation count for one MACC operation.

IV. Experiments and Numeral Results

In this section, we illustrate the training process in details and discuss the experimental results compared with several other methods of CSI feedback compression networks.

Two metrics are introduced to evaluate the models:

- **Normalized Mean Square Error (NMSE)**: it quantifies the difference between the input $\{\mathbf{H}_t\}_{t=1}^{T}$ and the output $\{\hat{\mathbf{H}}_t\}_{t=1}^{T}$, which can be defined as:

$$\mathrm{NMSE} = \mathbb{E}\left\{ \frac{1}{T} \sum_{t=1}^{T} \frac{\|\mathbf{H}_t - \hat{\mathbf{H}}_t\|_F^2}{\|\mathbf{H}_t\|_F^2} \right\} \tag{6}$$

- **Cosine Similarity** ρ: it depicts the similarity between the original CSI matrix $\tilde{\mathbf{H}}$ and the recovered $\hat{\tilde{\mathbf{H}}}$ by calculating cosine similarity within the channel response $\tilde{\mathbf{h}}_{n,t}$ ($n = 1, \cdots, \tilde{N}_c$) of each subcarrier, which is given as:

$$\rho = \mathbb{E}\left\{ \frac{1}{T} \frac{1}{\tilde{N}_c} \sum_{t=1}^{T} \sum_{n=1}^{\tilde{N}_c} \frac{|\hat{\tilde{\mathbf{h}}}_{n,t}^H \cdot \tilde{\mathbf{h}}_{n,t}|}{\|\hat{\tilde{\mathbf{h}}}_{n,t}\|_2 \|\tilde{\mathbf{h}}_{n,t}\|_2} \right\} \tag{7}$$

The MIMO-OFDM feedback system is set to work with $\tilde{N}_c = 1,024$ subcarriers and uniform linear array (ULA) with $N_t = 32$ antennas at the BS. After the DFT and truncation operations, only the first $N_c = 32$ columns in CSI feedback matrix $\mathbf{H}$ are nonzero and remain unchanged, which turns $\mathbf{H}$ from a $1,024 \times 32$ shape to a new 32×32 shape. According to Eq. 3, we add tiny white Gauss noise ($\sigma_u = 10^{-3}$) and coloration index α between each time step, and the 2D CSI feedback matrix can be extended to a T-time sequence of time-varying CSI matrix, where T is the recurrent time steps and is set to four for convenience.

All examples of $\mathbf{H}$ are generated based on the COST 2100 [17] channel model. We use the indoor picocellular scenario at the 5.3 GHz band, and the outdoor rural scenario at the 300 MHz band, respectively. The BS is fixed in the center of a square area of length 20 m for the indoor scene and 400 m for the outdoor scene, while UEs are randomly placed within the square area of each sample. All parameters follow the default setting in [17]. During the training process of each model, we use 100,000 examples for training, 30,000 for validation and 20,000 for testing. The learning rate is set to 10^{-3} for the first 1,000 epochs, 5×10^{-4} for the middle $1,000 - 1,200$ epochs and 10^{-4} for the last $1,200 - 1,500$ epochs.

TABLE II: The NMSE and ρ at different CRs when $\alpha = 0.1$

Scenario		Indoor				Outdoor			
CR		1/4	1/8	1/16	1/32	1/4	1/8	1/16	1/32
NMSE	CsiNet	-17.5	-12.3	-9.93	-6.98	-10.2	-7.40	-5.07	-3.43
	RecCsiNet	-21.5	-18.8	-16.8	-13.4	-15.1	-13.8	-12.9	-9.36
	ConvlstmCsiNet-A	**-28.4**	**-23.5**	**-20.7**	**-15.0**	**-20.8**	**-18.5**	**-16.5**	**-13.9**
	ConvlstmCsiNet-B	-25.9	-20.7	-18.3	-14.0	-19.0	-14.3	-13.3	-10.5
	ConvlstmCsiNet-C	-26.5	-22.0	-19.0	-14.4	-19.9	-17.0	-14.3	-12.7
	ConvlstmCsiNet	-24.9	-23.0	-18.7	-13.5	-15.5	-15.1	-14.5	-11.1
ρ	CsiNet	95.1%	93.1%	90.4%	87.4%	89.8%	85.1%	77.1%	66.8%
	RecCsiNet	95.7%	95.0%	94.7%	93.3%	93.2%	92.4%	91.9%	89.3%
	ConvlstmCsiNet-A	**95.8**%	**95.7**%	**95.5**%	**94.2**%	**94.3**%	**94.0**%	**93.5**%	**92.7**%
	ConvlstmCsiNet-B	**95.8**%	95.4%	95.0%	93.8%	93.8%	92.7%	92.2%	89.8%
	ConvlstmCsiNet-C	**95.8**%	95.6%	95.3%	93.7%	94.1%	93.6%	92.7%	91.8%
	ConvlstmCsiNet	95.7%	**95.7**%	95.2%	93.5%	93.2%	93.0%	92.7%	90.5%

Since the DL-based approaches are superior to the traditional CS-based methods, we only compare our methods with the DL-based approaches, such as CsiNet [7] and RecCsiNet [8]). The corresponding NMSE and ρ of each network are given in Table II, where the best results are marked in bold. The value of NMSE is too small that we use log(NMSE) to represent it. Obviously, our proposed model ConvlstmCsiNet-A can achieve the best performance on both NMSE and ρ.

TABLE III: The improvement percentage of proposed networks compared with CsiNet & RecCsiNet

Scenario			Indoor				Outdoor			
CR			1/4	1/8	1/16	1/32	1/4	1/8	1/16	1/32
Compare to CsiNet	NMSE	ConvlstmCsiNet-A	64.0%	91.1%	108.5%	114.9%	103.9%	150.0%	225.4%	305.2%
		ConvlstmCsiNet-B	48.0%	68.3%	84.3%	100.6%	86.3%	93.2%	162.3%	206.1%
		ConvlstmCsiNet-C	51.4%	78.9%	91.3%	106.3%	95.1%	129.7%	182.1%	270.3%
		ConvlstmCsiNet	42.3%	87.0%	88.3%	93.4%	52.0%	104.1%	186.0%	223.6%
	ρ	ConvlstmCsiNet-A	0.73%	2.80%	5.64%	7.78%	5.01%	10.5%	21.3%	38.8%
		ConvlstmCsiNet-B	0.73%	2.47%	5.09%	7.32%	4.45%	8.93%	19.6%	34.4%
		ConvlstmCsiNet-C	0.73%	2.69%	5.42%	7.21%	4.79%	10.0%	20.2%	37.4%
		ConvlstmCsiNet	0.63%	2.79%	5.31%	7.00%	3.79%	9.28%	20.2%	35.5%
Compare to RecCsiNet	NMSE	ConvlstmCsiNet-A	32.1%	25.0%	23.2%	11.9%	37.7%	34.1%	27.9%	48.5%
		ConvlstmCsiNet-B	20.5%	10.1%	8.93%	4.48%	25.8%	3.62%	3.10%	12.2%
		ConvlstmCsiNet-C	23.3%	17.0%	13.1%	7.46%	31.8%	23.2%	10.9%	35.7%
		ConvlstmCsiNet	15.8%	22.3%	11.3%	0.75%	2.65%	9.42%	12.4%	18.6%
	ρ	ConvlstmCsiNet-A	0.10%	0.74%	0.84%	0.96%	1.18%	1.73%	1.74%	3.81%
		ConvlstmCsiNet-B	0.10%	0.42%	0.32%	0.54%	0.64%	0.32%	0.33%	0.56%
		ConvlstmCsiNet-C	0.10%	0.63%	0.63%	0.43%	0.97%	1.30%	0.87%	2.80%
		ConvlstmCsiNet	0.00%	0.74%	0.53%	0.21%	0.00%	0.65%	0.87%	1.34%

To show the contrast more intuitively, we give percentage improvements of the proposed network compared with CsiNet and RecCsiNet in Tabel III. It demonstrates that all four purposed models outperform CsiNet and RecCsiNet. In the networks with P3D blocks, ConvlstmCsiNet-A achieves the best performance while ConvlstmCsiNet-B achieves the worst, indicating that the cascaded manner of temporal and spatial filter performs better than the parallel fashion, which can also be proved by the result that the performance of the combined structure ConvlstmCsiNet-C is between ConvlstmCsiNet-A and ConvlstmCsiNet-B.

When analyzing the functions of P3D blocks, all ConvlstmCsiNet-A, ConvlstmCsiNet-B and ConvlstmCsiNet-C have obtained much lower NMSE and higher cosine similarity ρ than ConvlstmCsiNet, especially at high CRs, indicating that the decoupling convolution structure (P3D block) does have a positive impact on capturing features and improving the performance of the network.

In Table III, we can find that in the first part (compared with CsiNet) that the improvements of all four networks are increasing as CR decreases due to a better and more complicated devised architecture. However, the increase in improvement becomes slower when compared with RecCsiNet, which indicates that the advantage of the ConvLSTM layer in feature extraction module in our models becomes less noticeable compared with RecCsiNet at low CRs. This is because the CR value only affects the performance of feature compression and decompression, where LSTM begins to play a major role in accelerating the convergence of models, emphasizing the benefits of LSTM and shrinking the advantage effects of feature extraction part.

Moreover, all the four models can achieve higher improvements in the outdoor scenario than in the indoor scenario, indicating the high robustness of our proposed methods, especially when applied to those difficult situations. Compared with ConvlstmCsiNet, the models using P3D block achieve much better improvements, which demonstrates that the P3D block has a positive effect on enhancing the robustness of the models.

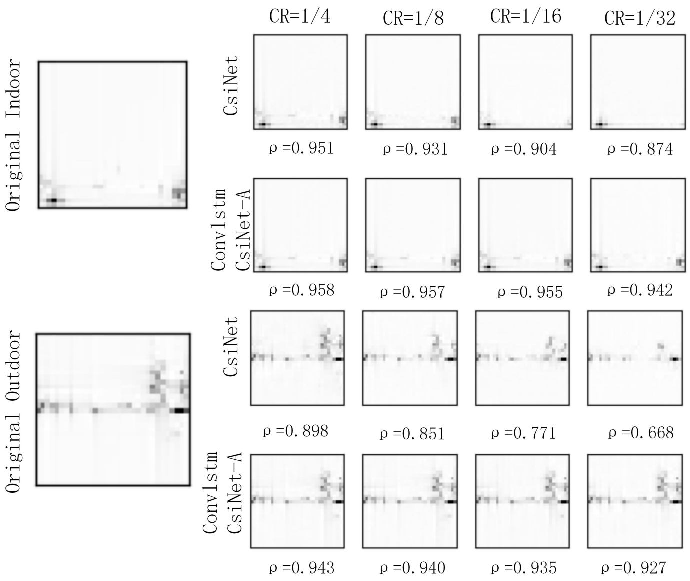


Fig. 5: The absolute value of original ($\alpha = 0.1$) and reconstructed CSI images at different CRs

Figure 5 plots the reconstructed CSI images by CsiNet and ConvlstmCsiNet-A (the best model we proposed) in Pseudo-gray. Obviously, ConvlstmCsiNet-A outperforms CsiNet, especially at low CRs. Furthermore, CsiNet may lose some feature information in both indoor and outdoor scenarios, while ConvlstmCsiNet-A does not. Particularly in the outdoor scenario, the cosine similarity of CsiNet decreases to a low 66.8%, while ConvlstmCsiNet-A always performs above 90%.

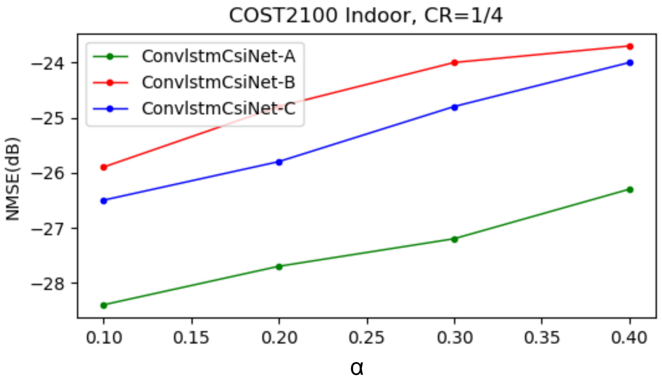


Fig. 6: The NMSE of the proposed NN at CR=1/4 in different correlation parameter α

Figure 6 demonstrates that the rise of α leads to a growth of corresponding NMSE, indicating that a decrease in temporal correlation may prevent the proposed networks from achieving high performance in CSI recovery.

V. Conclusion

We proposed a novel network architecture of CSI feedback by adopting RNN and depthwise separable convolution in feature extraction and recovery modules, respectively. Furthermore, we also devised the feature extraction part by studying the decoupled temporal-spatial convolutional representations, which proved to be better than standard Conv3D convolutions. Experimental results demonstrate that our method can improve the performance of RecCsiNet in terms of recovery robustness, accuracy and quality. This architecture has the potential for practical deployment on real MIMO systems.

References

[1] T. J. O'Shea, T. Erpek, and T. C. Clancy, "Deep learning based MIMO communications," *arXiv preprint arXiv:1707.07980*, Jul 2017.

[2] T. Wang, 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*, vol. 8, pp. 416–419, Oct 2018.

[3] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. Ten Brink, "Ofdm-autoencoder for end-to-end learning of communications systems," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, 2018.

[4] T. Wang, C. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Communications*, vol. 14, pp. 92–111, Nov 2017.

[5] J. Choi, D. J. Love, and P. Bidigare, "Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 802–814, Mar 2014.

[6] J. Zhang, C. Wen, S. Jin, X. Gao, and K. Wong, "On capacity of large-scale MIMO multiple access channels with distributed sets of correlated antennas," *IEEE Journal on Selected Areas in Communications*, vol. 31, pp. 133–148, Feb 2013.

[7] C. Wen, W. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Communications Letters*, vol. 7, pp. 748–751, Oct 2018.

[8] C. Lu, W. Xu, H. Shen, J. Zhu, and K. Wang, "MIMO channel information feedback using deep recurrent network," *IEEE Communications Letters*, vol. 23, pp. 188–191, Nov 2018.

[9] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, Jun 2018.

[10] T. Y. Al-Naffouri, "An EM-based forward-backward kalman filter for the estimation of time-variant channels in OFDM," *IEEE Transactions on Signal Processing*, vol. 55, pp. 3924–3930, Jun 2007.

[11] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, pp. 802–810, Jun 2015.

[12] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *proceedings of the IEEE International Conference on Computer Vision*, pp. 5533–5541, Oct 2017.

[13] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 4278–4284, AAAI Press, 2017.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*, pp. 630–645, Springer, Sep 2016.

[15] F. A. Aoudia and J. Hoydis, "End-to-end learning of communications systems without a channel model," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pp. 298–303, IEEE, 2018.

[16] F. A. Aoudia and J. Hoydis, "Model-free training of end-to-end communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2503–2516, 2019.

[17] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. De Doncker, "The COST 2100 MIMO channel model," *IEEE Wireless Communications*, vol. 19, pp. 92–99, Dec 2012.