A Spatially Separable Attention Mechanism for massive MIMO CSI Feedback

Sharan Mourya SaiDhiraj Amuru Kiran Kumar Kuchi

***Abstract*—Channel State Information (CSI) Feedback plays a crucial role in achieving higher gains through beamforming. However, for a massive MIMO system, this feedback overhead is huge and grows linearly with the number of antennas. To reduce the feedback overhead several compressive sensing (CS) techniques were implemented in recent years but these techniques are often iterative and are computationally complex to realize in power-constrained user equipment (UE). Hence, a data-based deep learning approach took over in these recent years introducing a variety of neural networks for CSI compression. Specifically, transformer-based networks have been shown to achieve state-of-the-art performance. However, the multi-head attention operation, which is at the core of transformers, is computationally complex making transformers difficult to implement on a UE. In this work, we present a lightweight transformer named STNet which uses a spatially separable attention mechanism that is significantly less complex than the traditional full-attention. Equipped with this, STNet outperformed state-of-the-art models in some scenarios with approximately $1/10^{th}$ of the resources.**

***Index Terms*—STNet, CSI Feedback, Transformers, Self-Attention, Massive MIMO, TransNet, CSIFormer, CLNet.**

I. Introduction

A massive MIMO system is equipped with hundreds of antennas that facilitate increased throughput with reduced BLER (Block Error Rate). Real-time channel state information (CSI) at the base station (gNB) plays an important role to meet the promises offered by massive MIMO. CSI at gNB allows the base station to perform beamforming and serve multiple users at once with minimal interference. The difficulty in this is that the channel experienced by the user equipment (UE) in the downlink has to be measured by UE and send it back to gNB in real-time which is not very convenient given the limited amount of resources at the UE end and the huge amount of overhead caused by CSI on the uplink. In order to deal with this problem, CSI compression methods were introduced where we compress the channel matrix at UE to reduce the feedback overhead and power consumption.

A Compressive Sensing (CS) based CSI feedback in FDD systems was studied in [1] where it was achieved by using 2-D Discrete Cosine Transform (DCT) or Karhunen-Loeve Transform (KCT). By exploiting sparsity, CS facilitates efficient data sampling at much lower rates than determined by the Nyquist theorem. However, this method assumes the channel matrices to be sparse which is not always the case. For efficient compression, the spatial correlation characteristics of the channel matrix have to be exploited which was proposed in [2] by using a principal component analysis (PCA). Even in this method, sparsity of the channel matrix in some basis is assumed for efficient compression but channels don't always have an interpretable structure.

In order to overcome this, a data-driven approach is chosen over an algorithmic-driven one and the usage of deep learning in CSI compression has taken over in recent years. CSINet [3] introduced a Convolutional Neural Network (CNN) based Variational Auto-Encoder (VAE) to the compression problem. This tremendously outperformed all the traditional CS-based methods. Inspired by this, another model was developed [4] with a larger receptor size, i.e., kernel size, of the CNN to better capture the spatial correlation in the angular-delay domain. However, the variability of the channel sparsity with the scenario means that a fixed receptor size is not sufficient to capture the correlation. Hence, a multiple-resolution CNN with varying receptor sizes was introduced by CRNet [5]. In order to focus the resources more on highly correlated areas and less on less correlated areas, it is useful to employ an attention mechanism on the CNNs which was first introduced by Attention-CSINet [6] that performed better in outdoor scenarios where the variability of correlation is more dominant. In all these methods, real and imaginary parts of the channel matrices are treated separately which is not efficient in capturing the correlation in the angular domain as the complex number as a whole contains the phase information. To overcome this, a simple approach to combining real and imaginary values of a channel matrix was introduced by CLNet [7] which outperformed several models. CLNet is also computationally less complex compared to other methods.

So far, CNNs are used for feature extraction in all the models. A transformer [8]-based architecture with a full attention mechanism was first studied in [9] that was not very competitive compared to the state-of-the-art models. A two-layer transformer architecture named TransNet [10] was introduced that outperformed several models by a significant amount but the computational complexity of TransNet was very high and not practically affordable. Another transformer-based model with locally grouped (windowed) self-attention was studied by CSIFormer in [11]. Although this has low complexity compared to TransNet, the performance was sacrificed.

In this work, we introduce a spatially separable attention mechanism [12] that can achieve state-of-the-art performance with very less computational complexity. We also introduce a hybrid two-stem approach in the decoder that combines CNN with a transformer for better channel reconstruction [13]. We then validate the performance of our model on the COST2100 dataset [14].

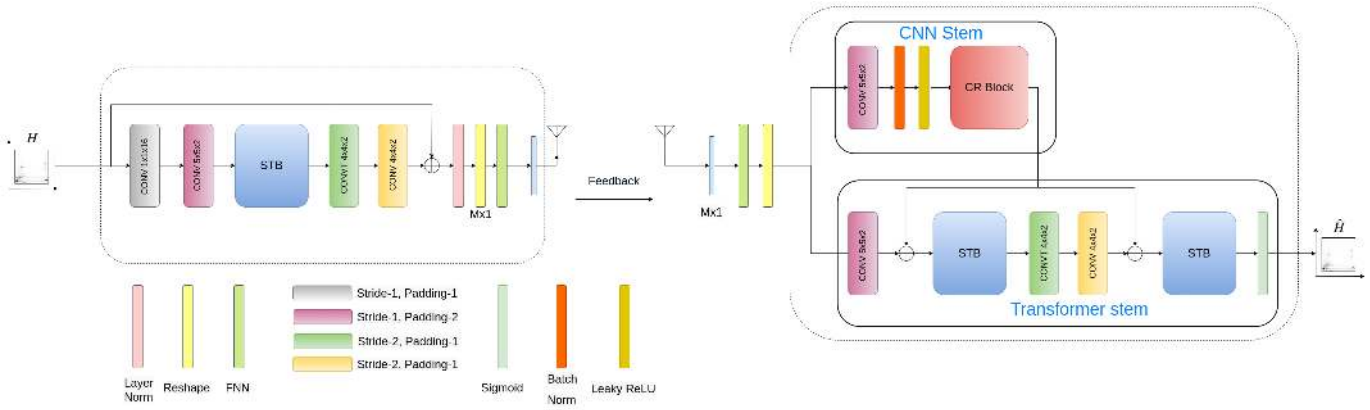

Fig. 1: Proposed encoder-decoder architecture for CSI feedback aka STNet. "CONV" represents a convolutional layer and "CONVT" represents a transposed convolutional layer. The encoder consists of a few CONV and CONVT layers with a spatially separable attention transformer block "STB". The decoder consists of two stems: the CNN stem and the transformer stem. CR Block is a multi-resolution CNN block proposed in CRNet as shown in Fig. 2

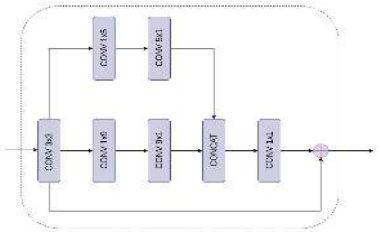

Fig. 2: CR Block as proposed in CRNet architecture. It consists of two paths with different kernel sizes that are concatenated (represented by "CONCAT" block) at the end and combined using a 1x1 convolutional layer.

II. System Model

In this work, we consider a Frequency Division Duplex (FDD) system with N_t antennas at the base station (gNB) and 1 antenna at the user equipment (UE) such that $N_t \gg 1$. This employs Orthogonal Frequency Division Multiplexing (OFDM) with $\tilde{N}_c$ sub-carriers. The received signal at UE on the n^{th} sub-carrier can be expressed as

$$y_n = \tilde{\boldsymbol{h}}_{\boldsymbol{n}}^H \tilde{\boldsymbol{v}}_{\boldsymbol{n}} x_n + w_n, \tag{1}$$

where, $\tilde{\boldsymbol{h}}_{\boldsymbol{n}} \in \mathbb{C}^{N_t \times 1}$, $\tilde{\boldsymbol{v}}_{\boldsymbol{n}} \in \mathbb{C}^{N_t \times 1}$, $x_n \in \mathbb{C}$ and $w_n \in \mathbb{C}$ represent channel vector, precoding vector, symbol transmitted and additive noise on the n^{th} sub-carrier. The overall channel matrix is of the dimension $\tilde{N}_c \times N_t$ and is expressed as

$$\tilde{\boldsymbol{H}} = \left[\tilde{\boldsymbol{h}}_{\boldsymbol{1}}, \tilde{\boldsymbol{h}}_{\boldsymbol{2}}, \cdots, \tilde{\boldsymbol{h}}_{\tilde{\boldsymbol{N}}_{\boldsymbol{c}}}\right]^H. \tag{2}$$

The total number of feedback elements is $2N_t\tilde{N}_C$ (for both real and imaginary parts of the channel) which is huge and impractical in real scenarios considering $N_t = 32, 64, \ldots$ and $\tilde{N}_C = 1024, 2048, \ldots$ for a massive MIMO system. So, we reduce the overhead by making the channel matrix sparse. Specifically, we achieve this by transforming it into angular-delay domain [3] as follows

$$\bar{\boldsymbol{H}} = \boldsymbol{F_d} \tilde{\boldsymbol{H}} \boldsymbol{F_a^H}, \tag{3}$$

where, $\boldsymbol{F_d}$ and $\boldsymbol{F_a}$ are 2-D DFT matrices of dimensions $\tilde{N}_c \times \tilde{N}_c$ and $N_t \times N_t$ respectively. In the delay domain, the time delay between multipath arrivals lies within a limited period. Using this, we can truncate the matrix $\bar{\boldsymbol{H}}$ by only keeping the first N_c rows where N_c is chosen such that remaining entries of $\bar{\boldsymbol{H}}$ are close to zero [3]. We define this truncated matrix as $\boldsymbol{H}$ that has dimensions $N_c \times N_t$. Also, we split this matrix into real and imaginary parts and combine them as a third dimension similar to RGB channels of an image. With this, the overall feedback overhead becomes $2N_cN_t$ which is significantly smaller than earlier as N_c will only be a fraction of $\tilde{N}_c$ (total number of sub-carriers).

Now that we have the sparsified channel matrix, $\boldsymbol{H}$, it is sent into the encoder-decoder architecture as shown in Fig. 1 where $\boldsymbol{H}$ is compressed into a 1-D vector of dimension $M \times 1$. Here, we define compression ratio as $\gamma = \frac{M}{2N_cN_t}$. This compressed channel matrix is sent back to gNB from UE on the uplink. gNB then decodes this fedback signal as $\hat{\boldsymbol{H}}$. This encoding and decoding process is defined as follows

$$s = f_e(\boldsymbol{H}) \quad \& \quad \hat{\boldsymbol{H}} = f_d(s),$$

where f_e, f_d denote the functions of the encoder and decoder, respectively. s is the compressed code word and $\hat{\boldsymbol{H}}$ is the estimated channel matrix by the model.

III. Architecture

In this section, we describe a high-level overview of how f_e and f_d of our proposed model STNet[1] are designed. Transformers are traditionally designed to capture global context using a global self-attention mechanism which makes them highly efficient in modeling high-level semantics that may be sufficient for a classification task. For example, in our model Spatially Separable Attention Transformer Block

[1]Source code of this paper: https://github.com/sharanmourya/Pytorch_STNet

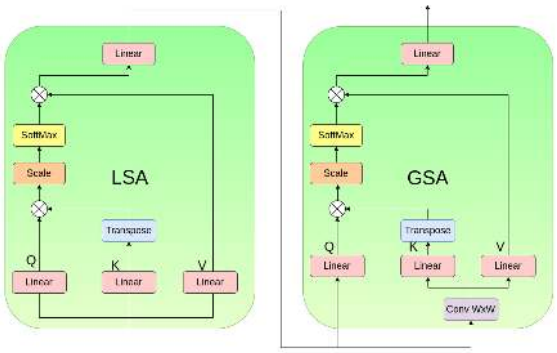

Fig. 3: Spatially Separable Attention Mechanism (Locally grouped self-attention (LSA) followed by Global sub-sampled attention (GSA)).

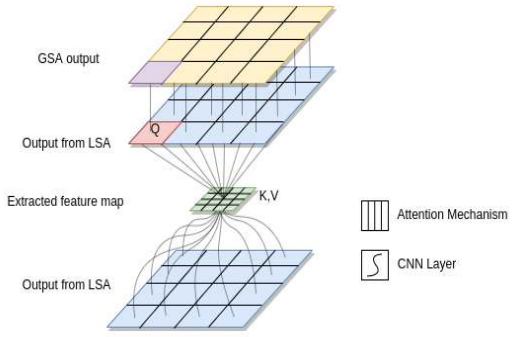

Fig. 4: Global Sub-Sampled Attention (GSA) with sampling performed by a convolutional layer (shown in curved lines) followed by LSA (shown in vertical lines).

(STB) captures the long-range correlation between antennas. But, channel reconstruction also requires low-level details in order to minimize the reconstruction error. These low-level details are better captured by CNNs which also provide better generalization due to their spatial invariance. So in order to get the best of both worlds, we use a hybrid approach in our decoder design with two stems [13], one consisting of a transformer and the other consisting of CNNs (Fig. 1).

A. Spatially Separable Attention Mechanism

First, let's summarize the self-attention mechanism of a transformer. Every channel matrix that enters the attention block is fed to three independent linear layers as shown in Fig. 3. The outputs of these three branches are queries (Q), keys (K), and values (V) respectively. If the input is X, these values are calculated as follows

$$Q_n = XW_n^Q, \quad K_n = XW_n^K, \quad V_n = XW_n^V,$$

where W_n^Q, W_n^K, W_n^V are the weights of the respective linear layers on the n^{th} head of a P headed multi-head attention block. With this, the attention is calculated as,

$$A_n = Softmax\left(\frac{Q_n K_n^T}{\sqrt{d}}\right), \tag{4}$$

where A_n is the attention on the n^{th} head and d is the output dimension of Q_n and K_n. This attention is then multiplied with the values across all the heads

$$Y_n = A_n V_n, \tag{5}$$

which are then concatenated to get the final output.

$$Y = [Y_1, Y_2, \cdots, Y_T]. \tag{6}$$

This is also called full attention or global attention as the receptor region of the attention block is the full channel matrix. This global attention mechanism has a complexity of $\mathbb{O}(L^4 d)$ when operating on a channel of dimension $L \times L$ with encoded dimension d [12]. One way to reduce the complexity is to reduce the receptor region of the attention by using windowed attention, where each channel matrix is sub-divided into $m \times m$ smaller matrices with dimensions $W \times W$, where $W = \frac{L}{m}$ and attention is calculated for each window separately. This is called Locally Grouped Self-Attention (LSA) and this reduces the complexity to $\mathbb{O}(\frac{L^4}{m^4} d)$.

As the windows are fixed and do not communicate with one another, the antenna correlation across windows is now lost and can't be utilized in compressing the channel matrix. To solve this, we can introduce a global attention layer after LSA, but that would only increase the complexity further. So, we introduce another layer of locally grouped attention after LSA that can capture the antenna correlations between windows. This can be achieved by a Global Sub-Sampled Attention (GSA) layer which is shown in Fig. 4.

In GSA, LSA's output is first followed by a CNN layer (with $stride = W$). The output of this layer is a $m \times m$ feature map in which each element represents a window from which it is extracted. This now becomes keys and values (K and V) for another layer of windowed attention whose queries (Q) are the same output of LSA that we used to obtain the feature map from as shown in Fig. 4. Suppose X is the output from LSA, we apply a CNN layer to X to get a feature map of dimensions $m \times m$. This feature map becomes K and V for X, which becomes the query, Q. We are constructing the global attention from the feature map which is a sub-sampled version of X, hence the name global sub-sampled attention. This GSA layer has a complexity of $\mathbb{O}(m^2 L^2 d)$ [12] and with this, the total complexity of the attention mechanism becomes $\mathbb{O}(\frac{L^4}{m^4} d) + \mathbb{O}(m^2 L^2 d)$. The entire attention mechanism (LSA+GSA) is shown in Fig. 3. Note that this approach of breaking down a complex operation into two simpler operations is similar to separable convolutions (point-wise + depth-wise) [15]. Hence, the name spatially separable attention.

B. Spatially Separable Attention Transformer Block (STB)

STB consists of four different types of blocks which are LSA, GSA, LayerNorm, and Multi-Layer Perceptron (MLP) as shown in Fig. 5. MLP block has a linear layer followed by a Gaussian Error Linear Unit (GELU) non-linearity and a linear layer again as shown in Fig. 5. The entire architecture of STNet with STBs is shown in Fig. 1. It can be seen that its encoder is slightly more complex than that of CLNet [7],

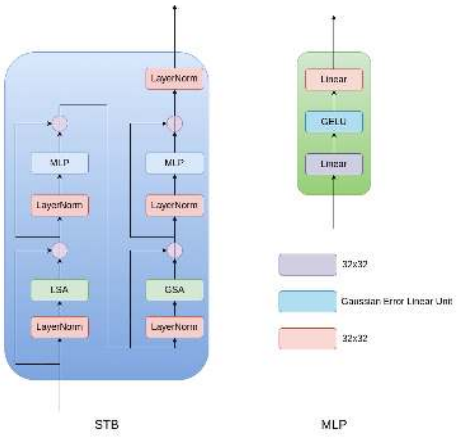

Fig. 5: STB consists of both LSA and GSA each followed by an add & normalize layer and a Multi-Layer perceptron (MLP). MLP has a linear layer of dimension 32×32 followed by a Gaussian Error Linear Unit (GELU).

CRNet [5], or CSInet [3] with more CNN layers and an STB block. The reason for which is explained in Section IV.

IV. Analysis

A. Model Performance

We consider a system with 32×1 antennas (i.e., 32 antennas at BS and 1 antenna at UE). For evaluation purposes, we choose the COST2100 dataset with two scenarios: the indoor picocellular scenario at 5.3GHz and the outdoor rural scenario at 300MHz. We choose $N_c = 32$, window size: $W = 8$ and number of heads of multi-head attention: $P = 4$. The training, validation, and test datasets consist of 100,000, 30,000, and 20,000 matrices, respectively. Batch size is set to 200 and epochs to 1000. The learning rate is 0.001 and the loss function is the Mean Squared Error (MSE) with an Adam optimizer.

$$MSE = \frac{1}{B}\sum_{i=1}^{B} ||\boldsymbol{H} - \hat{\boldsymbol{H}}||^2, \tag{7}$$

where $\boldsymbol{H}$ is the input channel matrix, $\hat{\boldsymbol{H}}$ is the reconstructed channel matrix and B is the batch size. We use Normalised Mean Square Error (NMSE) as the performance metric which is defined as follows

$$NMSE = \mathbb{E}\left\{\frac{||\boldsymbol{H} - \hat{\boldsymbol{H}}||^2}{||\boldsymbol{H}||^2}\right\}. \tag{8}$$

The number of floating-point operations per second (FLOPS) and runtime delay of the model is other important factors when comparing the models as deployment is also done in UEs which are power and memory-constrained devices. So we tabulated the NMSE results of our model over the COST2100 dataset and its FLOPs and runtimes compared with various other models in Table I.

STNet is compared with the recently proposed transformer-based models CSIFormer [11] and TransNet [10]. TransNet may have performed well in most cases but it takes significantly more FLOPs to achieve that. For the indoor case with $CR = 1/16$, STNet achieved 102.86% of the performance of TransNet with just 11.6% of FLOPs. Similarly, for $CR = 1/64$, STNet achieved 128.45% of the performance of TransNet with just 10.8% of its FLOPs. Also, notice that STNet performs better than CSIFormer in every case while consuming less number of FLOPs. For instance consider outdoor environment with $CR = 1/64$. STNet achieved 116.44% of the performance of CSIFormer with 65.88% of its FLOPs. Also from Table I, we can see that the runtimes of STNet are comparable to other models. It is evident from these results that STNet exploits the trade-off between performance and complexity perfectly.

Although the runtimes of CSIFormer and TransNet are not available, the runtimes of a similar full attention mechanism based model called CSITransformer [9] are available which can be used for comparison with STNet. CSITransformer is evaluated on a different dataset so its NMSE results are not listed. However, the size of the channel matrices used by it is 32×32 which is the same as all the other models so the runtimes of it can be used in a fair comparison with STNet. Thus, the runtimes of CSITransformer for $CR = 1/16$ and $CR = 1/32$ are listed in Table I and we can see that STNet is faster than CSITransformer in both scenarios.

B. Communication System Performance

A Massive MIMO system can achieve high capacities by using transmit precoding. Therefore, we use the widely common linear Zero-Forcing (ZF) transmit precoding to evaluate the overall performance improvement of the communication system due to different CSI feedback methods [11]. For evaluation, the spectral efficiency of each method is plotted against SNR for various compression ratios as shown in Fig. 6. At 10dB SNR, spectral efficiency values of all the methods are labeled and from Fig. 6, we can conclude that STNet performs better than every model in every scenario and the difference is more profound for $CR = 1/16$ as STNet achieves lowest NMSE for this scenario. This improvement in system performance is achieved by STNet's ability to capture antenna and sub-carrier correlations at the encoder side. This operation is so critical that the encoder consumes more than 40% of the entire STNet's resources. For example in an indoor scenario with $CR = 1/4$, STNet's encoder has 2.09 Million FLOPs which is 40.03% of the total FLOPs which is 5.22 Million. On the other hand, CLNet's encoder under similar conditions has 1.11 Million FLOPs which is 25.11% of the total FLOPs which is 4.42 Million. Although STNet has a slightly higher encoder complexity than CSINet, CRNet, or CLNet, its encoder complexity would still be better than other transformer-based models making it a promising choice for storage and computational limited applications because of its higher spectral efficiency and lower or similar runtime as other models.

V. Conclusion

In this work, a lightweight transformer architecture with spatially separable attention is introduced for CSI feedback.

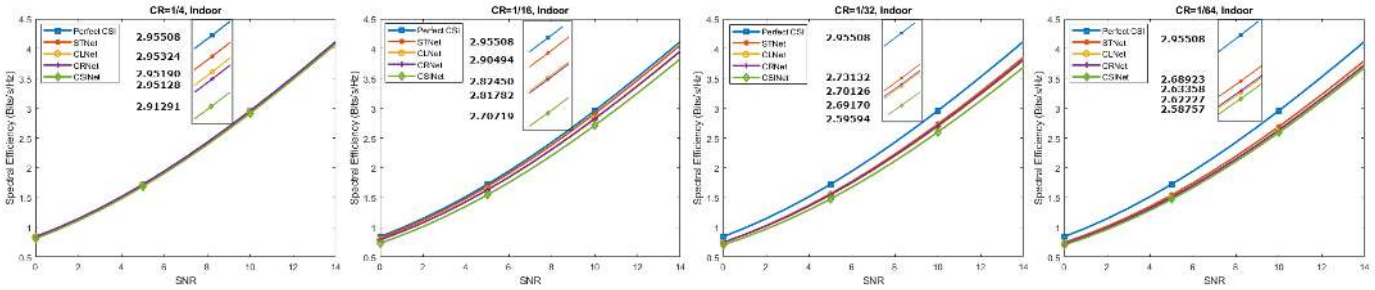

Fig. 6: Spectral Efficiency vs SNR plots for different CSI feedback methods along with perfect CSI. Spectral efficiency values at SNR=10dB are zoomed in and labeled for clarity.

TABLE I: Performance over Cost 2100 dataset

NMSE										
Compression Ratio (γ)	1/4		1/8		1/16		1/32		1/64	
Scenario	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor
CSINet	-17.36	-8.75	/	/	-8.65	-4.51	-6.24	-2.81	-5.84	-1.93
CRNet	-26.99	-12.71	-16.01	-8.04	-11.35	-5.44	-8.93	-3.51	-6.49	-2.22
CLNet	-29.16	-12.88	-15.60	-8.29	-11.15	-5.56	-8.95	-3.49	-6.34	-2.19
CSIFormer	/	/	/	/	/	/	-9.32	-3.51	-6.85	-2.25
TransNet	**-32.38**	**-14.86**	**-22.91**	**-9.99**	-15.00	**-7.82**	**-10.49**	**-4.13**	-6.08	**-2.62**
STNet	-31.81*	-12.91*	-21.28*	-8.53*	**-15.43**	-5.72*	-9.42*	-3.51*	**-7.81**	-2.46*

FLOPS and RUNTIME (in seconds)										
Compression Ratio (γ)	1/4		1/8		1/16		1/32		1/64	
Scenario	FLOPS	Runtime	FLOPS	Runtime	FLOPS	Runtime	FLOPS	Runtime	FLOPS	Runtime
CSINet	5.41M	0.0001	4.37M	0.0001	3.84M	0.0001	3.58M	0.0001	3.45M	0.0001
CRNet	5.12M	0.0003	4.07M	0.0003	3.55M	0.0003	3.28M	0.0003	3.16M	0.0003
CLNet	4.42M	0.0002	3.37M	0.0002	2.85M	0.0002	2.58M	0.0002	2.45M	0.0002
CSIFormer	/	-	/	-	/	-	5.41M	-	5.54M	-
TransNet	35.72M	-	34.70M	-	34.14M	-	33.88M	-	33.75M	-
STNet	5.22M	0.0004	4.38M	0.0003	3.96M	0.0003	3.75M	0.0003	3.65M	0.0003
CSITransformer	/	/	/	/	/	0.003	/	0.002	/	/

/ indicates that the performance is not reported in the original paper

* indicates the second-best value in that column and - indicates that the code is not made public in order to generate the results

Along with this, a hybrid approach to channel reconstruction is also introduced where we use a two stems approach (CNN and Transformer) that improves the channel reconstruction quality. We evaluated the performance and runtime of STNet along with other models on the COST2100 dataset. Combining both techniques, STNet produced the best performance per floating-point operation among various other models.

REFERENCES

[1] P.-H. Kuo and H. T. Kung et al., "Compressive sensing based channel feedback protocols for spatially-correlated massive antenna arrays," in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 492–497, 2012.

[2] A. Ge and T. Zhang et al., "Principal component analysis based limited feedback scheme for massive mimo systems," in *2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 326–331, 2015.

[3] C.-K. Wen and W.-T. Shih et al., "Deep learning for massive mimo csi feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, 2018.

[4] J. Guo and C.-K. Wen et al., "Convolutional neural network-based multiple-rate compressive sensing for massive mimo csi feedback: Design, simulation, and analysis," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2827–2840, 2020.

[5] Z. Lu and J. Wang et al., "Multi-resolution csi feedback with deep learning in massive mimo system," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2020.

[6] Q. Cai and C. Dong et al., "Attention model for massive mimo csi compression feedback and recovery," in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–5, 2019.

[7] S. Ji and M. Li, "Clnet: Complex input lightweight neural network designed for massive mimo csi feedback," *IEEE Wireless Communications Letters*, vol. 10, no. 10, pp. 2318–2322, 2021.

[8] A. Vaswani and N. Shazeer et al., "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, (Red Hook, NY, USA), p. 6000–6010, Curran Associates Inc., 2017.

[9] Y. Xu and M. Yuan et al., "Transformer empowered csi feedback for massive mimo systems," in *2021 30th Wireless and Optical Communications Conference (WOCC)*, pp. 157–161, 2021.

[10] Y. Cui, A. Guo, and C. Song, "Transnet: Full attention network for csi feedback in fdd massive mimo system," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 903–907, 2022.

[11] X. Bi, S. Li, C. Yu, and Y. Zhang, "A novel approach using convolutional transformer for massive mimo csi feedback," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 1017–1021, 2022.

[12] X. Chu and Z. Tian et al., "Twins: Revisiting the design of spatial attention in vision transformers," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 9355–9366, Curran Associates, Inc., 2021.

[13] D. Ye and Z. Ni et al., "Csformer: Bridging convolution and transformer for compressive sensing," 2021.

[14] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. D. Doncker, "The cost 2100 mimo channel model," *IEEE Wireless Communications*, vol. 19, no. 6, pp. 92–99, 2012.

[15] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, 2017.