Efficient Deep Learning for Massive MIMO Channel State Estimation

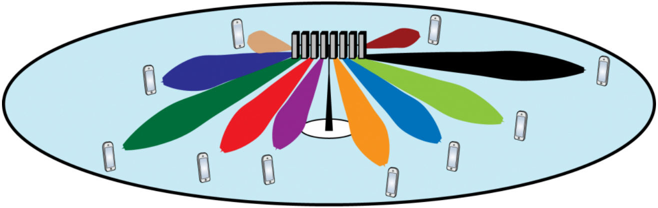
Mason del Rosario

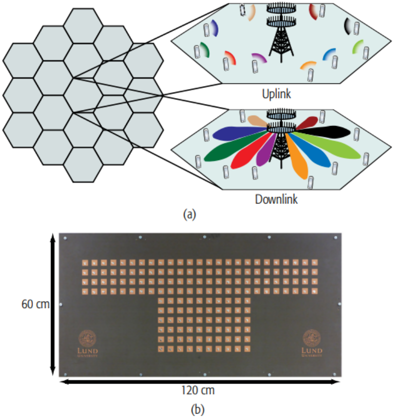
Doctoral Qualifying Examination

June 2021

Background

Feedback-based estimation of channel state information in MIMO networks.

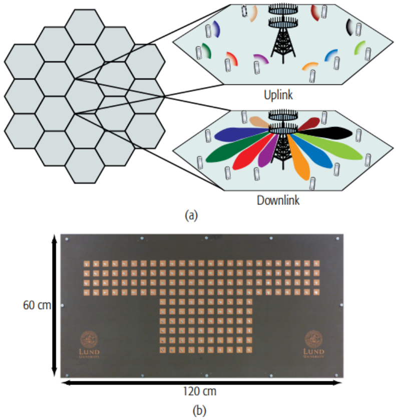




- MIMO = Multiple input multiple output

E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: Ten Myths and One Critical Question," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 114–123, 2016

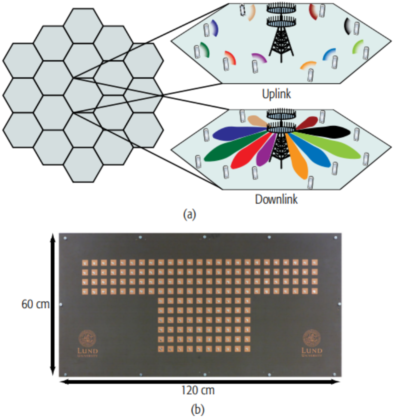
MIMO and CSI



- MIMO = Multiple input multiple output
- Massive w.r.t. antenna count, not physical size.

E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: Ten Myths and One Critical Question," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 114–123, 2016

MIMO and CSI



- MIMO = Multiple input multiple output
- Massive w.r.t. antenna count, not physical size.
- Spatial diversity → **high throughput**.

E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: Ten Myths and One Critical Question," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 114–123, 2016

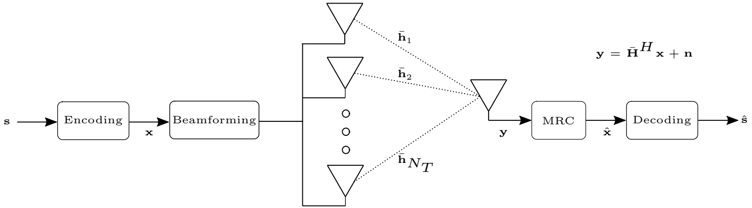


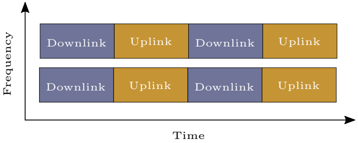
Figure: Multi-antenna transmitter (BS, gNB) and single-antenna user equipment (UE) with relevant system values.

In OFDM, the fading coefficients between Tx/Rx = **Channel State Information (CSI)**, $\bar{\mathbf{H}}$.

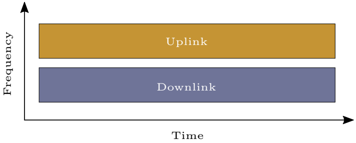
$$\bar{\mathbf{H}} = \begin{bmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,N_T} \\ h_{2,1} & h_{2,2} & \dots & h_{2,N_T} \\ \vdots & \vdots & \vdots & \vdots \\ h_{N_f,1} & h_{N_f,2} & \dots & h_{N_f,N_T} \end{bmatrix} \in \mathbb{C}^{N_f \times N_T}$$

For N_T transmit antennas and N_f subcarriers.

Downlink-uplink reciprocity in TDD, but not in FDD.

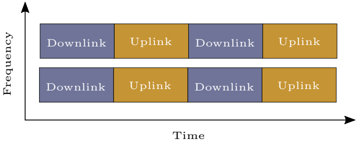


a) Time division duplex (TDD)

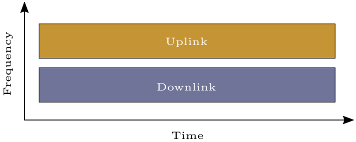


b) Frequency division duplex (FDD)

Downlink-uplink reciprocity in TDD, but not in FDD.



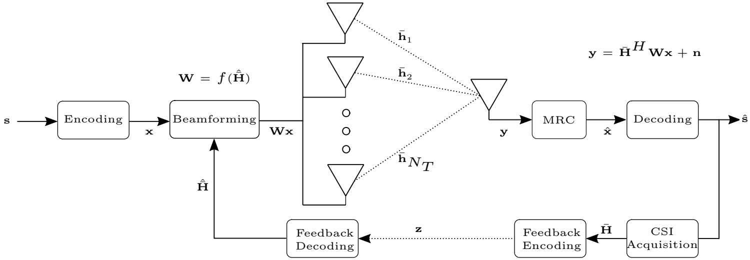
a) Time division duplex (TDD)



b) Frequency division duplex (FDD)

FDD requires feedback for downlink CSI estimation.

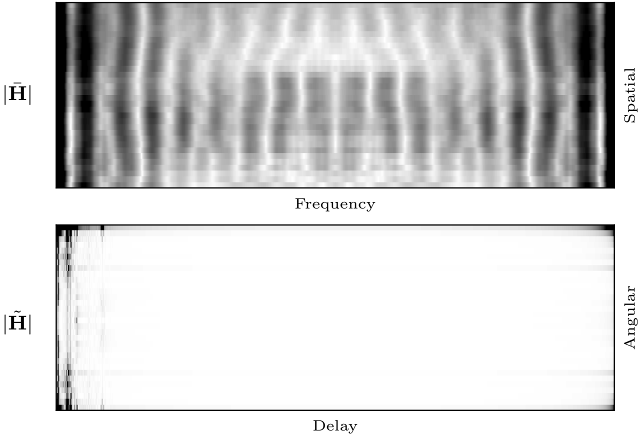
Transmitting $\bar{\mathbf{H}}$ is costly. Instead, generate estimates, $\hat{\bar{\mathbf{H}}}$, based on **compressed feedback**, $\mathbf{z}$.



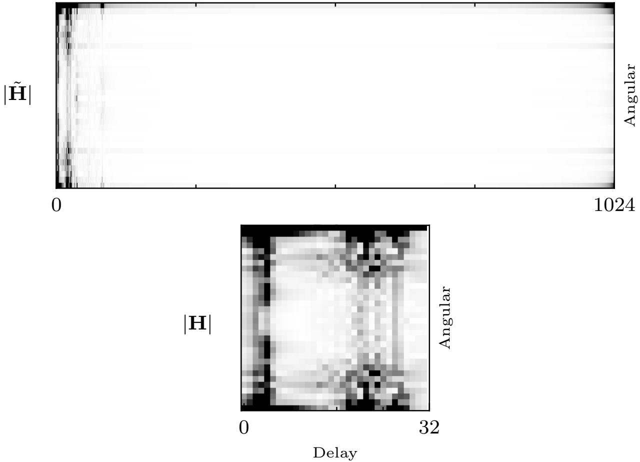
CSI Sparsity

Denote 2D inverse FFT of $\bar{\mathbf{H}}$ as

$$\tilde{\mathbf{H}} = \mathbf{F}^H \bar{\mathbf{H}} \mathbf{F}.$$



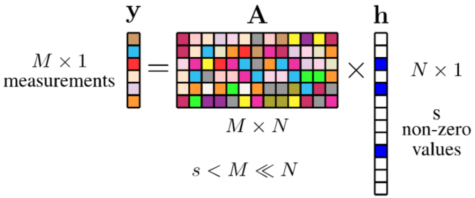
Given sparsity of $\tilde{\mathbf{H}}$, we can encode/decode a truncated version, $\mathbf{H}$.



1. Compressed Sensing (Conventional)
2. Convolutional Neural Networks (This proposal)

Find low-dimensional basis for sparse data, $\mathbf{h}$,

$$\mathbf{y} = \mathbf{A}\mathbf{h} + \mathbf{n}.$$



E. Crespo Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, "A Review of Sparse Recovery Algorithms," *IEEE Access*, vol. 7, pp. 1300–1322, 2019

CS relies on the following assumptions:

1. $\mathbf{h}$ meets a sparsity level s, number of nonzero coefficients.

CS relies on the following assumptions:

1. $\mathbf{h}$ meets a sparsity level s, number of nonzero coefficients.
2. **Restricted Isometry Property (RIP)**. For $\delta \in [0,1]$,

$$(1-\delta)\|\mathbf{h}\|^2 \leq \|\mathbf{A}\mathbf{h}\|^2 \leq (1+\delta)\|\mathbf{h}\|^2$$

   for Frobenius norm $\|\cdot\|$.

CS addresses two major issues:

1. Design of $\mathbf{A}$ (stochastic or deterministic).

CS addresses two major issues:

1. Design of $\mathbf{A}$ (stochastic or deterministic).
2. Recovery of $\hat{\mathbf{h}}$ given $\mathbf{A}$ and $\mathbf{y}$, typically via convex optimization on p-norm minimization,

$$\min \|\hat{\mathbf{h}}\|_p \text{ subject to } \|\mathbf{y} - \mathbf{A}\hat{\mathbf{h}}\|_2^2 < \epsilon.$$

CS addresses two major issues:

1. Design of $\mathbf{A}$ (stochastic or deterministic).
2. Recovery of $\hat{\mathbf{h}}$ given $\mathbf{A}$ and $\mathbf{y}$, typically via convex optimization on p-norm minimization,

$$\min \|\hat{\mathbf{h}}\|_p \text{ subject to } \|\mathbf{y} - \mathbf{A}\hat{\mathbf{h}}\|_2^2 < \epsilon.$$

Problems:

- Recovery algorithms are iterative.

CS addresses two major issues:

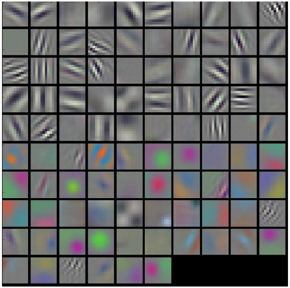
1. Design of $\mathbf{A}$ (stochastic or deterministic).
2. Recovery of $\hat{\mathbf{h}}$ given $\mathbf{A}$ and $\mathbf{y}$, typically via convex optimization on p-norm minimization,

$$\min \|\hat{\mathbf{h}}\|_p \text{ subject to } \|\mathbf{y} - \mathbf{A}\hat{\mathbf{h}}\|_2^2 < \epsilon.$$

Problems:

- Recovery algorithms are iterative.
- Complexity scales with sparsity ($M \propto s$).

- CNNs = state-of-the art performance in image processing
- Multiple layers of trainable linear functions followed nonlinear ‘activation’ functions.
- No assumptions on sparsity/RIP. Instantaneous decoding.

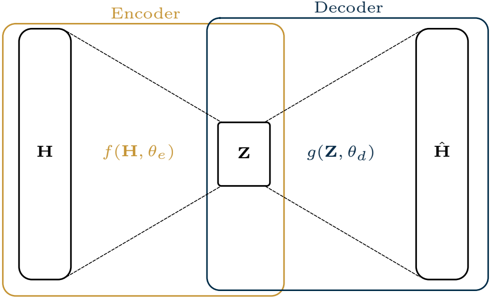


A. Karpathy, “Visualizing What ConvNets Learn," `http://cs231n.github.io/understanding-cnn/`. Accessed: Feb 24, 2020.

CNN Autoencoder

Autoencoder: Estimate $\hat{\mathbf{H}}$, latent code $\mathbf{Z}$ with **compression ratio**,

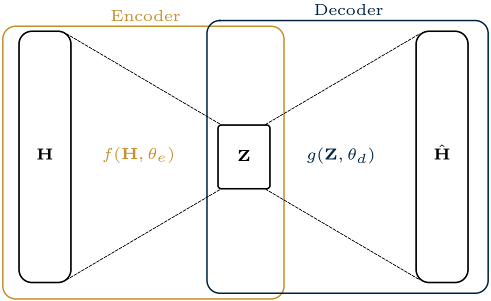
$$\mathrm{CR} = \frac{\dim(\mathbf{Z})}{\dim(\mathbf{H})} \text{ s.t. } \dim(\mathbf{Z}) < \dim(\mathbf{H}).$$



CNN Autoencoder

Autoencoder: Estimate $\hat{\mathbf{H}}$, latent code $\mathbf{Z}$ with **compression ratio**,

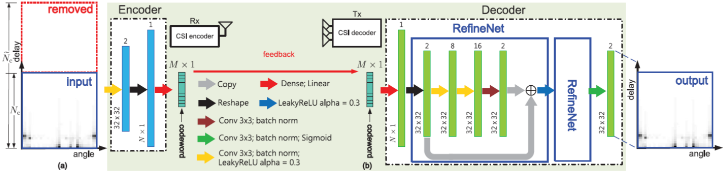
$$\mathrm{CR} = \frac{\dim(\mathbf{Z})}{\dim(\mathbf{H})} \text{ s.t. } \dim(\mathbf{Z}) < \dim(\mathbf{H}).$$



θ_e, θ_d updated to minimize **mean-squared error (MSE)**,

$$\underset{\theta_e, \theta_d}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{H}_i - g(f(\mathbf{H}_i, \theta_e), \theta_d)\|^2.$$

- CNN autoencoder for learned CSI compression and feedback [3]



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

Metrics used:

- **Normalized Mean-squared Error**

$$\text{NMSE} = \frac{1}{N}\sum_{i}^{N} \frac{\|\mathbf{H}_i - \hat{\mathbf{H}}_i\|_F^2}{\|\mathbf{H}_i\|_F^2}$$

- **Cosine Similarity**

$$\rho = \frac{1}{NN_f}\sum_{i=1}^{N}\sum_{m=1}^{N_f} \frac{|\hat{\bar{\mathbf{h}}}_{i,m}^H \bar{\mathbf{h}}_{i,m}|}{\|\hat{\bar{\mathbf{h}}}_{i,m}\|\|\bar{\mathbf{h}}_{i,m}\|},$$

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

CsiNet vs. Compressed Sensing

Metrics used:

- **Normalized Mean-squared Error**

$$\text{NMSE} = \frac{1}{N}\sum_{i}^{N}\frac{\|\mathbf{H}_i - \hat{\mathbf{H}}_i\|_F^2}{\|\mathbf{H}_i\|_F^2}$$

- **Cosine Similarity**

$$\rho = \frac{1}{NN_f}\sum_{i=1}^{N}\sum_{m=1}^{N_f}\frac{|\hat{\bar{\mathbf{h}}}_{i,m}^H \bar{\mathbf{h}}_{i,m}|}{\|\hat{\bar{\mathbf{h}}}_{i,m}\|\|\bar{\mathbf{h}}_{i,m}\|},$$

CNNs outperform CS at comparable compression ratios.

γ	Methods	Indoor NMSE	Indoor ρ	Outdoor NMSE	Outdoor ρ
1/4	LASSO	-7.59	0.91	-5.08	0.82
	BM3D-AMP	-4.33	0.80	-1.33	0.52
	TVAL3	-14.87	0.97	-6.90	0.88
	CS-CsiNet	-11.82	0.96	-6.69	0.87
	CsiNet	**-17.36**	**0.99**	**-8.75**	**0.91**
1/16	LASSO	-2.72	0.70	-1.01	0.46
	BM3D-AMP	0.26	0.16	0.55	0.11
	TVAL3	-2.61	0.66	-0.43	0.45
	CS-CsiNet	-6.09	0.87	-2.51	0.66
	CsiNet	**-8.65**	**0.93**	**-4.51**	**0.79**
1/32	LASSO	-1.03	0.48	-0.24	0.27
	BM3D-AMP	24.72	0.04	22.66	0.04
	TVAL3	-0.27	0.33	0.46	0.28
	CS-CsiNet	-4.67	0.83	-0.52	0.37
	CsiNet	**-6.24**	**0.89**	**-2.81**	**0.67**
1/64	LASSO	-0.14	0.22	-0.06	0.12
	BM3D-AMP	0.22	0.04	25.45	0.03
	TVAL3	0.63	0.11	0.76	0.19
	CS-CsiNet	-2.46	0.68	-0.22	0.28
	CsiNet	**-5.84**	**0.87**	**-1.93**	**0.59**

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

CsiNet vs. Compressed Sensing

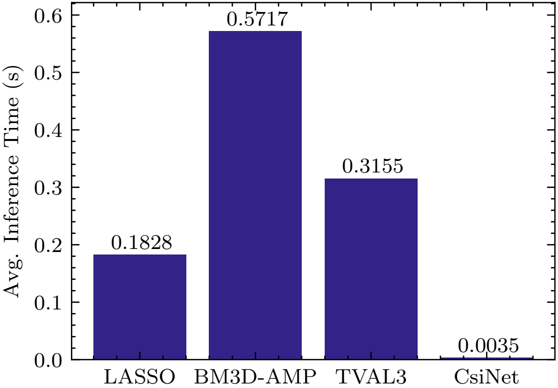


Figure: Average inference time for compressed sensing methods vs. CsiNet.

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

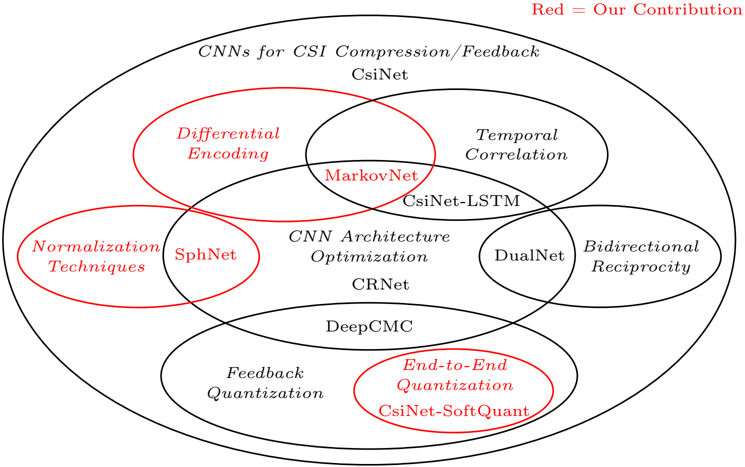
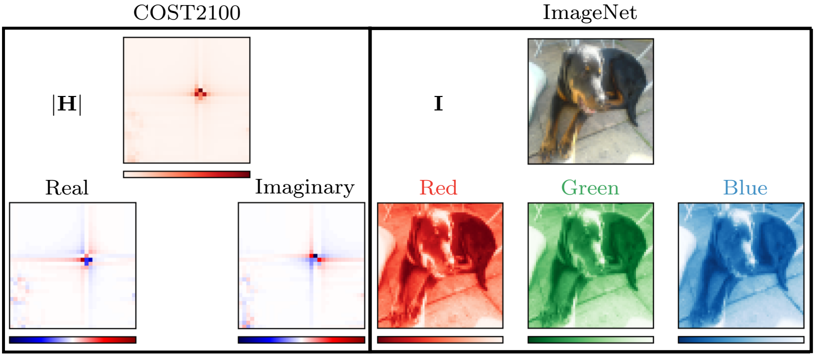


Figure: Areas of *domain knowledge* and corresponding CNNs.

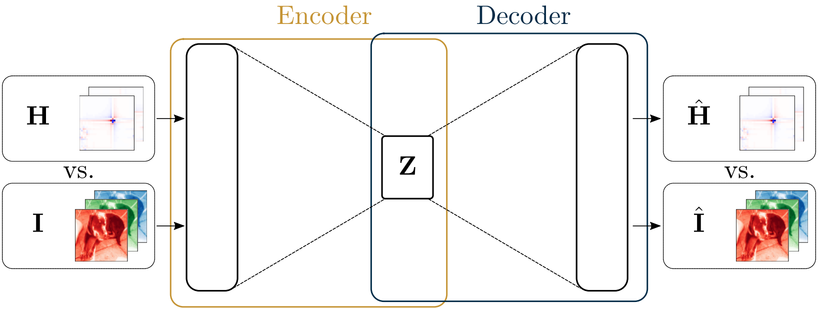
Completed Work #1: SphNet

Power-based normalization for improved CSI reconstruction accuracy.

CSI vs. Images: Channels



CSI vs. Images: CNNs



CsiNet: Minmax Normalization

- **Minmax normalization** – Find minimum, maximum of channels.

CsiNet: Minmax Normalization

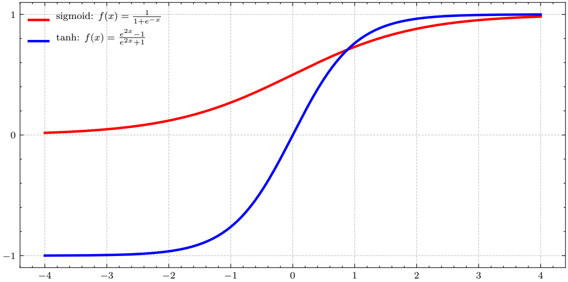
- **Minmax normalization** – Find minimum, maximum of channels.
- $H_{n,(i,j)} = (i,j)$-th element of n-th sample

$$H_{\text{minmax},n,(i,j)} = \frac{H_{n,(i,j)} - H_{\min}}{H_{\max} - H_{\min}} \in [0,1]$$

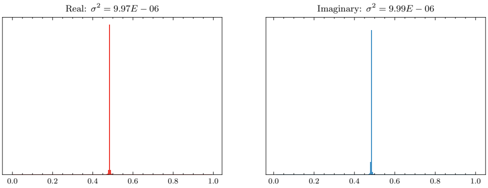
- **Minmax normalization** – Find minimum, maximum of channels.
- $H_{n,(i,j)} = (i,j)$-th element of n-th sample

$$H_{\text{minmax},n,(i,j)} = \frac{H_{n,(i,j)} - H_{\min}}{H_{\max} - H_{\min}} \in [0,1]$$

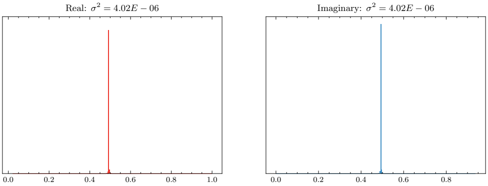
- Compatible with common **activation functions** (e.g., tanh, sigmoid)



COST2100: Minmax Normalization



(a) Indoor



(b) Outdoor

Figure: Distribution/variance of COST2100 real/imaginary channels under minmax normalization ($N = 10^5$).

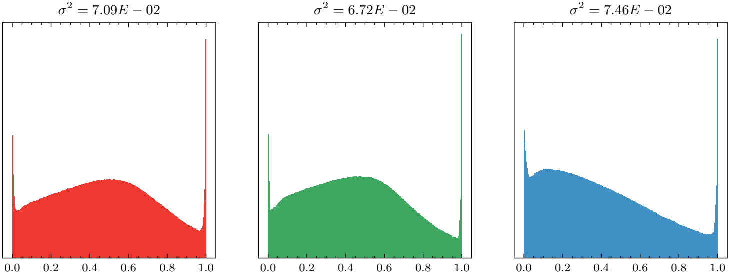


Figure: Distribution and variance of minmax-normalized ImageNet RGB channels ($N = 50000$).

Difference of **four orders of magnitude**.

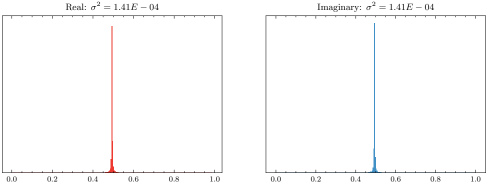
Dataset	**Env**	**Channels**	**Norm**	**Avg. Variance**
ImageNet	-	RGB	Minmax	$\underline{7.09E^{-2}}$
COST2100	Indoor	Real, Imag	Minmax	$\underline{9.98E^{-6}}$
COST2100	Outdoor	Real, Imag	Minmax	$\underline{4.02E^{-6}}$

Table: Minmax normalization applied to COST2100 and ImageNet dataset.

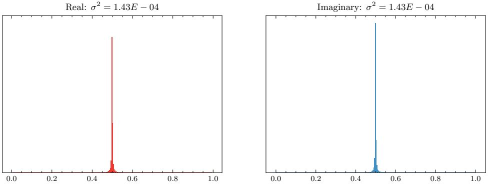
Spherical normalization – scale $\mathbf{H}$ by power. For Frobenius norm $\|\cdot\|$,

$$\check{\mathbf{H}}^n = \frac{\mathbf{H}^n}{\|\mathbf{H}^n\|}. \tag{1}$$

Then apply minmax scaling to the entire dataset.



(a) Indoor



(b) Outdoor

Figure: Distribution/variance of COST2100 real/imaginary channels under spherical normalization ($N = 10^5$).

Difference is now **two orders of magnitude**.

Dataset	**Env**	**Channels**	**Norm**	**Avg. Variance**
ImageNet	-	RGB	Minmax	$\underline{7.09E^{-2}}$
COST2100	Indoor	Real, Imag	Spherical	$\underline{1.41E^{-4}}$
COST2100	Outdoor	Real, Imag	Spherical	$\underline{1.43E^{-4}}$
COST2100	Indoor	Real, Imag	Minmax	$9.98E^{-6}$
COST2100	Outdoor	Real, Imag	Minmax	$4.02E^{-6}$

Table: Minmax vs. spherical normalization applied to COST2100 datasets compared with ImageNet.

MSE/NMSE equivalence

Spherical normalization → MSE equivalent to NMSE.

$$\mathrm{MSE} = \frac{1}{N}\sum_{k=1}^{N} \|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2, \quad \mathrm{NMSE} = \frac{1}{N}\sum_{k=1}^{N} \frac{\|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2}{\|\mathbf{H}_k\|}$$

MSE/NMSE equivalence

Spherical normalization → MSE equivalent to NMSE.

$$\mathrm{MSE} = \frac{1}{N}\sum_{k=1}^{N} \|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2, \quad \mathrm{NMSE} = \frac{1}{N}\sum_{k=1}^{N} \frac{\|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2}{\|\mathbf{H}_k\|}$$

MSE of spherically normalized estimator yields,

$$\begin{aligned}
\mathrm{MSE}_{\mathrm{Sph}} &= \frac{1}{N}\sum_{k=1}^{N} \|\check{\mathbf{H}}_k - \hat{\check{\mathbf{H}}}_k\|^2 \\
&= \frac{1}{N}\sum_{k=1}^{N} \left\| \frac{\mathbf{H}_k}{\|\mathbf{H}_k\|} - \frac{\hat{\mathbf{H}}_k}{\|\mathbf{H}_k\|} \right\|^2 \\
&= \frac{1}{N}\sum_{k=1}^{N} \frac{\|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2}{\|\mathbf{H}_k\|}.
\end{aligned}$$

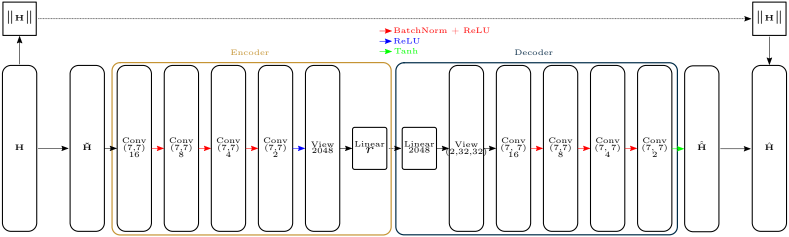


Figure: SphNet – CsiNetPro architecture with Spherical Normalization.

Z. Liu, **M. del Rosario**, X. Liang, L. Zhang, and Z. Ding, "Spherical Normalization for Learned Compressive Feedback in Massive MIMO CSI Acquisition," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2020

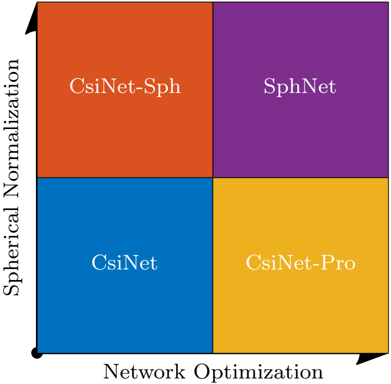


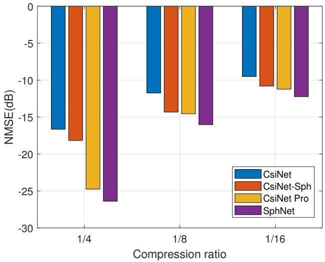
Figure: Illustration of techniques used in different models.

Z. Liu, **M. del Rosario**, X. Liang, L. Zhang, and Z. Ding, “Spherical Normalization for Learned Compressive Feedback in Massive MIMO CSI Acquisition,” in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2020

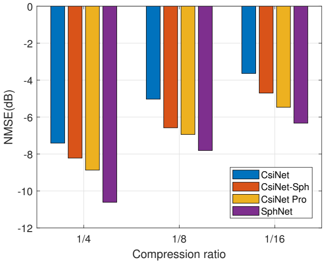
Table: Parameters for COST2100 model in this work.

Environment	**Indoor**	**Outdoor**
Num. gNB Antennas (N_T)	32	
Num. Subcarriers (N_f)	1024	
Truncation Value (R_d)	32	
Carrier Frequency	5.3 GHz	300 MHz
UE Mobility	0.001 m/s	1 m/s
UE Starting Position	20×20 m	400×400 m
Num. Channel Samples (N)	10^5	
Training/Validation Split	70%/30%	
Feedback interval	40 ms	

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, pp. 92–99, December 2012



(a) Indoor



(b) Outdoor

Figure: Ablation study for CsiNet-Pro and spherical normalization [4] (lower NMSE is better).

Z. Liu, **M. del Rosario**, X. Liang, L. Zhang, and Z. Ding, "Spherical Normalization for Learned Compressive Feedback in Massive MIMO CSI Acquisition," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2020

Completed Work #2: MarkovNet

A deep differential autoencoder for efficient temporal learning.

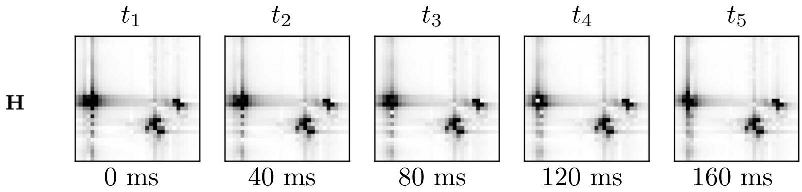


Figure: Ground truth CSI (**H**) for five timeslots (T_1 through T_5) on one outdoor sample from the validation set.

Recurrent neural networks (RNNs) contain trainable long short-term memory (LSTM) cells which learn temporal relationships.

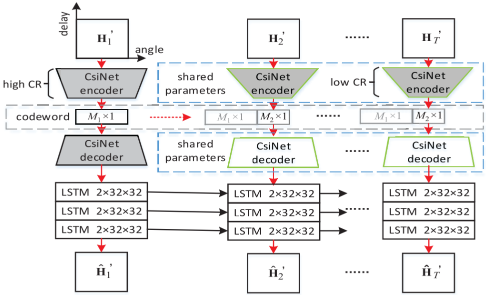


Figure: CsiNet-LSTM network architecture [6].

T. Wang, C. Wen, S. Jin, and G. Y. Li, “Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels,” *IEEE Wireless Comm. Letters*, vol. 8, pp. 416–419, April 2019

CsiNet-LSTM Performance

LSTMs improve NMSE at smaller compression ratios.

		CR	LASSO	BM3D-AMP	TVAL3	CsiNet	CsiNet-LSTM
Indoor	NMSE	1/16	-2.96	0.25	-3.20	-10.59	**-23.06**
		1/32	-1.18	20.85	-0.46	-7.35	**-22.33**
		1/64	-0.18	26.66	0.60	-6.09	**-21.24**
	ρ	1/16	0.72	0.29	0.73	0.95	**0.99**
		1/32	0.53	0.17	0.45	0.90	**0.99**
		1/64	0.30	0.16	0.24	0.87	**0.99**
	runtime	1/16	0.2471	0.3454	0.3148	**0.0001**	0.0003
		1/32	0.2137	0.5556	0.3148	**0.0001**	0.0003
		1/64	0.2479	0.6047	0.2860	**0.0001**	0.0003
	NMSE↓	1/16-1/64	94%	105	1.19	42%	**8%**
Outdoor	NMSE	1/16	-1.09	0.40	-0.53	-3.60	**-9.86**
		1/32	-0.27	18.99	0.42	-2.14	**-9.18**
		1/64	-0.06	24.42	0.74	-1.65	**-8.83**
	ρ	1/16	0.49	0.23	0.46	0.75	**0.95**
		1/32	0.32	0.16	0.28	0.63	**0.94**
		1/64	0.19	0.16	0.19	0.58	**0.93**
	runtime	1/16	0.2122	0.4210	0.3145	**0.0001**	0.0003
		1/32	0.2409	0.6031	0.2985	**0.0001**	0.0003
		1/64	0.0166	0.5980	0.2850	**0.0001**	0.0003
	NMSE↓	1/16-1/64	94%	60	2.40	54%	**10%**

T. Wang, C. Wen, S. Jin, and G. Y. Li, "Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels," *IEEE Wireless Comm. Letters*, vol. 8, pp. 416–419, April 2019

Problem: Number of parameters/FLOPs for RNNs is large.

Table: Model size/computational complexity per timeslot for CsiNet-LSTM and CsiNet. M: million.

	Parameters		**FLOPs**	
CR	**CsiNet-LSTM**	**CsiNet**	**CsiNet-LSTM**	**CsiNet**
1/4	132.7 M	2.1 M	412.9 M	7.8 M
1/8	123.2 M	1.1 M	410.8 M	5.7 M
1/16	118.5 M	0.5 M	409.8 M	4.7 M
1/32	116.1 M	0.3 M	409.2 M	4.1 M
1/64	115.0 M	0.1 M	409.0 M	3.9 M

For short enough feedback interval, CSI data form a Markov chain,

$$\mathbf{H}_t = \gamma \mathbf{H}_{t-1} + \mathbf{V}_t,$$

with $\gamma \in \mathbb{R}^+$ and i.i.d $\mathbf{V}_t \sim \mathcal{CN}(\mathbf{0}, \Sigma_V)$.

Z. Liu †, **M. del Rosario †**, and Z. Ding, "A Markovian Model-Driven Deep Learning Framework for Massive MIMO CSI Feedback," *arXiv e-prints*, Sept.

2020. Submitted to IEEE Transactions on Wireless Communications († equal contribution)

The ordinary least-squares solution, γ, is given as

$$\gamma = \frac{\mathrm{Trace}(\mathbb{E}\left[\mathbf{H}_{t-1}^{H}\mathbf{H}_{t}\right])}{\mathbb{E}\|\mathbf{H}_{t}^{H}\mathbf{H}_{t}\|^{2}}.$$

The ordinary least-squares solution, γ, is given as

$$\gamma = \frac{\mathrm{Trace}(\mathbb{E}\left[\mathbf{H}_{t-1}^{H}\mathbf{H}_t\right])}{\mathbb{E}\|\mathbf{H}_t^{H}\mathbf{H}_t\|^2}.$$

Utilize estimator, $\hat{\gamma}$, based on the sample statistics,

$$\hat{\gamma} = \frac{\sum_{i=1}^{N} \mathrm{Trace}(\left[\mathbf{H}_{t-1}^{H}(i)\mathbf{H}_t(i)\right])}{\sum_{i=1}^{N} \|\mathbf{H}_t^{H}(i)\mathbf{H}_t(i)\|^2},$$

for training set of size N.

Using $\hat{\gamma}$, train encoder on estimation error as

$$\begin{aligned}\mathbf{E}_t &= \mathbf{H}_t - \hat{\gamma}\hat{\mathbf{H}}_{t-1} \\ \mathbf{z}_t &= f_{e,t}(\mathbf{E}_t).\end{aligned}$$

Jointly train a decoder,

$$\begin{aligned}\hat{\mathbf{E}}_t &= f_{d,t}(\mathbf{z}_t) \\ \hat{\mathbf{H}}_t &= \hat{\mathbf{E}}_t + \hat{\gamma}\hat{\mathbf{H}}_{t-1}.\end{aligned}$$

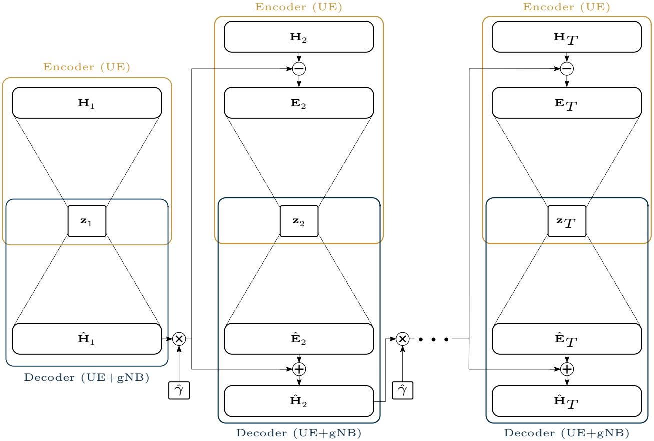
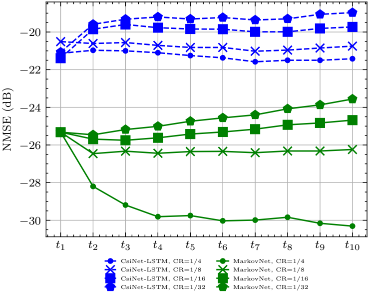
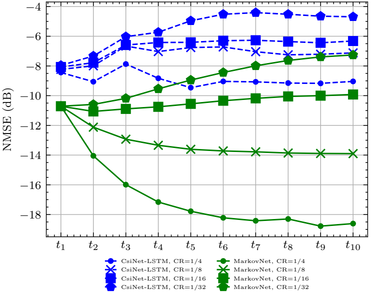


Figure: MarkovNet architecture. Networks at $t \geq 2$ predict estimation error, $\hat{\mathbf{E}}_t$.



(a) Indoor



(b) Outdoor

Figure: NMSE (lower is better) comparison of MarkovNet and CsiNet-LSTM at multiple CRs.

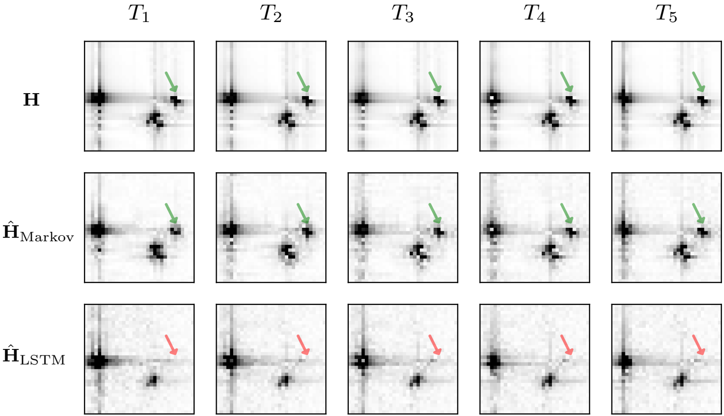


Figure: Ground truth ($\mathbf{H}$), MarkovNet estimates ($\hat{\mathbf{H}}_{\text{Markov}}$), and CsiNet-LSTM estimates ($\hat{\mathbf{H}}_{\text{LSTM}}$) on from outdoor test set (CR $= \frac{1}{4}$).

Table: Model size/computational complexity of tested temporal networks (CsiNet-LSTM, MarkovNet) and comparable non-temporal network (CsiNet). M: million.

	Parameters		
	CsiNet-LSTM	**MarkovNet**	**CsiNet**
CR=1/4	132.7 M	2.1 M	2.1 M
CR=1/8	123.2 M	1.1 M	1.1 M
CR=1/16	118.5 M	0.5 M	0.5 M
CR=1/32	116.1 M	0.3 M	0.3 M
CR=1/64	115.0 M	0.1 M	0.1 M
	FLOPs		
	CsiNet-LSTM	**MarkovNet**	**CsiNet**
CR=1/4	412.9 M	44.5 M	7.8 M
CR=1/8	410.8 M	42.4 M	5.7 M
CR=1/16	409.8 M	41.3 M	4.7 M
CR=1/32	409.2 M	40.8 M	4.1 M
CR=1/64	409.0 M	40.5 M	3.9 M

Current Work: CsiNet-SoftQuant

An end-to-end trained autoencoder with learned feedback quantization.

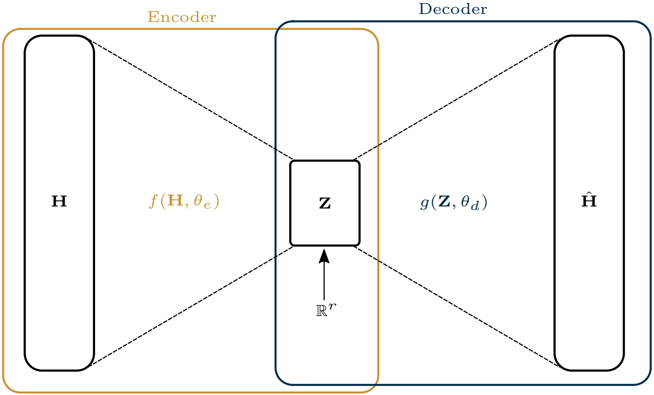


Figure: Autoencoder architecture with r-dimensional real-valued latent feedback elements.

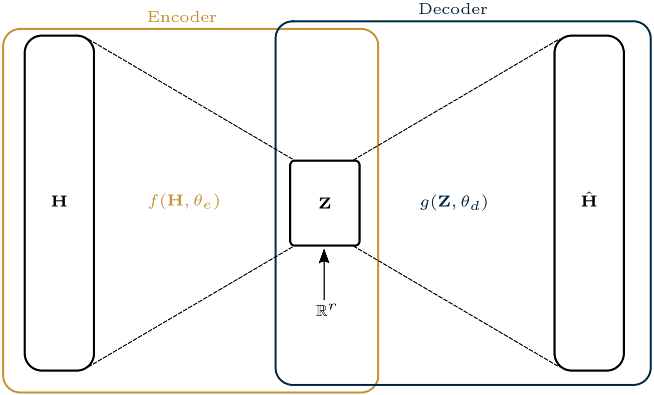


Figure: Autoencoder architecture with r-dimensional real-valued latent feedback elements.

Problem: Feedback elements must be discrete-valued. How to quantize?

One solution: Uniform quantization and arithmetic encoding of latent vectors.

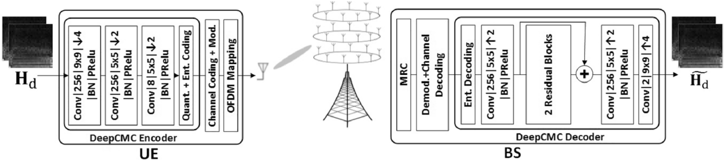


Figure: Architecture for DeepCMC [8].

Q. Yang, M. B. Mashhadi, and D. Gündüz, "Deep Convolutional Compression For Massive MIMO CSI Feedback," in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2019

Uniform Quantization

One solution: Uniform quantization and arithmetic encoding of latent vectors.

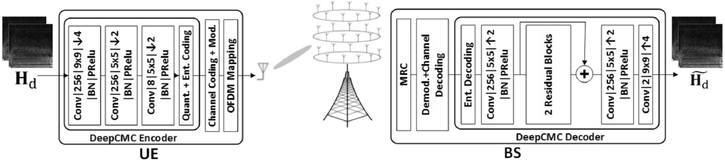


Figure: Architecture for DeepCMC [8].

Is fixed quantization scheme optimal?

Q. Yang, M. B. Mashhadi, and D. Gündüz, "Deep Convolutional Compression For Massive MIMO CSI Feedback," in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2019

Soft-to-hard vector quantization (SHVQ) [9] – Given codebook $\mathbf{C} \in \mathbb{R}^{m \times L}$, soft vector assignment for j-th latent vector $\tilde{\mathbf{z}}_j$ is

$$\phi(\tilde{\mathbf{z}}_j) = \left[\frac{\exp(-\sigma \|\tilde{\mathbf{z}}_j - \mathbf{c}_\ell\|^2)}{\sum_{i=1}^{L} \exp(-\sigma \|\tilde{\mathbf{z}}_j - \mathbf{c}_i\|^2)} \right]_{\ell \in [L]} \in \mathbb{R}^L, \tag{2}$$

E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. Van Gool, “Soft-to-hard Vector Quantization for End-to-end Learning Compressible Representations,” *Advances in Neural Information Processing Systems*, vol. 2017-Decem, no. NIPS, pp. 1142–1152, 2017

Soft-to-hard vector quantization (SHVQ) [9] – Given codebook $\mathbf{C} \in \mathbb{R}^{m \times L}$, soft vector assignment for j-th latent vector $\tilde{\mathbf{z}}_j$ is

$$\phi(\tilde{\mathbf{z}}_j) = \left[\frac{\exp(-\sigma \|\tilde{\mathbf{z}}_j - \mathbf{c}_\ell\|^2)}{\sum_{i=1}^{L} \exp(-\sigma \|\tilde{\mathbf{z}}_j - \mathbf{c}_i\|^2)} \right]_{\ell \in [L]} \in \mathbb{R}^L, \tag{2}$$

$\sigma =$ *temperature* parameter controls degree of quantization,

$$\lim_{\sigma \to \infty} \phi(\tilde{\mathbf{z}}_j) = \text{onehot}(\tilde{\mathbf{z}}_j) = \begin{cases} 1 & \ell = \operatorname*{argmax}_{\ell} \phi(\tilde{\mathbf{z}}_j)[\ell] \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. Van Gool, “Soft-to-hard Vector Quantization for End-to-end Learning Compressible Representations,” *Advances in Neural Information Processing Systems*, vol. 2017-Decem, no. NIPS, pp. 1142–1152, 2017

Soft assignments $\phi \rightarrow$ probability masses over codewords,

$$q_j = \phi(\tilde{\mathbf{z}}_j).$$

Based on finite samples, define histogram probability estimates p_j,

$$p_j = \frac{|\{e_l(\mathbf{z}_i)|l \in [m], i \in [N], e_l(\mathbf{z}_i) = j\}|}{mN}.$$

Target for the rate loss the crossentropy between p_j and q_j,

$$H(\phi) := H(p, q) = -\sum_{j=1}^{L} p_j \log q_j = H(p) + D_{\mathrm{KL}}(p\|q).$$

E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. Van Gool, "Soft-to-hard Vector Quantization for End-to-end Learning Compressible Representations," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, no. NIPS, pp. 1142–1152, 2017

Rate-Distortion Loss

Loss function for soft quantization = regularized rate-distortion

$$\underset{\theta_e,\theta_d,\mathbf{C}}{\operatorname{argmin}} \overbrace{L_d(\mathbf{H},\hat{\mathbf{H}})}^{\text{distortion}} + \lambda \overbrace{L_{\ell 2}(\theta_e,\theta_d,\mathbf{C})}^{\ell_2 \text{ penalty}} + \beta \overbrace{L_r(\theta_e,\mathbf{C})}^{\text{rate}} \tag{4}$$

Where loss terms are defined as

Term	**Definition**
$L_d(\mathbf{H},\hat{\mathbf{H}})$	$\frac{1}{N}\sum_{i=1}^{N} \lVert \mathbf{H}_i - g(Q(f(\mathbf{H}_i,\theta_e),\mathbf{C}),\theta_d)\rVert^2$
$L_{\ell 2}(\theta_e,\theta_d,\mathbf{C})$	$\lVert\theta_e\rVert^2 + \lVert\theta_d\rVert^2 + \lVert\mathbf{C}\rVert^2$
$L_r(\theta_e,\mathbf{C})$	$mH(\phi)$

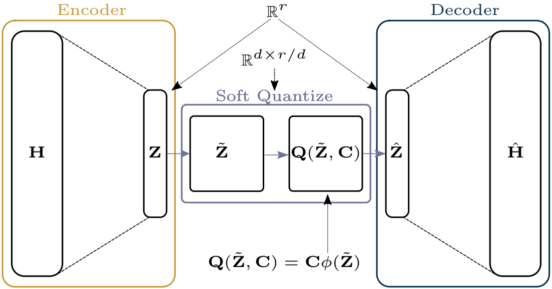


Figure: Abstract architecture for CsiNet-SoftQuant [8]. SoftQuantize layer ($Q(\tilde{\mathbf{Z}})$) is a continuous, softmax-based relaxation of a d-dimensional quantization of the latent layer $\mathbf{Z}$.

M. del Rosario and Z. Ding, "Trainable Codewords and Compression Bounds for Deep Learning-based Multi-Antenna CSI Feedback," May 2021.

Submitted to IEEE GLOBECOM 2021

Results: Rate-Distortion (Outdoor)

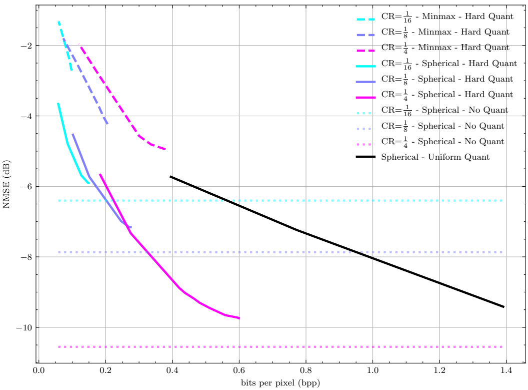


Figure: Rate-distortion of CsiNet-SoftQuant under minmax and spherical normalization ($L = 1024$, $d = 4$).

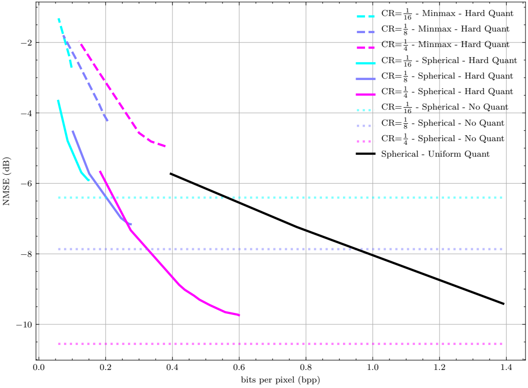


Figure: Rate-distortion of CsiNet-SoftQuant under minmax and spherical normalization ($L = 1024$, $d = 4$).

- **Question: What is the limit of compression?**

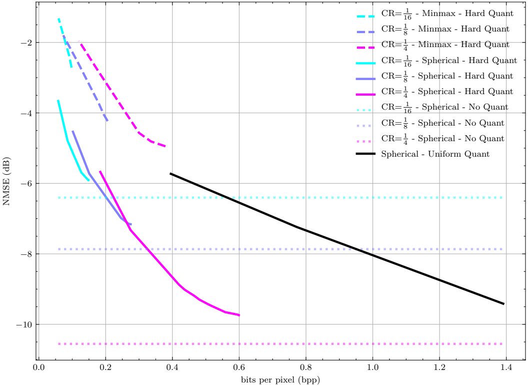


Figure: Rate-distortion of CsiNet-SoftQuant under minmax and spherical normalization ($L = 1024$, $d = 4$).

- **Question: What is the limit of compression?**
- **Answer: Entropy of CSI.**

Assume i.i.d. $\mathbf{H}_{(i,j)}$ for i-th (j-th) row (col).

L. Kozachenko and N. N. Leonenko, "Sample Estimate of the Entropy of a Random Vector," *Problemy Peredachi Informatsii*, vol. 23, no. 2, pp. 9–16, 1987

Assume i.i.d. $\mathbf{H}_{(i,j)}$ for i-th (j-th) row (col).

The differential entropy of the (i, j)-th element is

$$h(\mathbf{H}_{(i,j)}) = -\int p(\mathbf{H}(i,j) = k) \log p(\mathbf{H}_{(i,j)} = k) dk,$$

L. Kozachenko and N. N. Leonenko, "Sample Estimate of the Entropy of a Random Vector," *Problemy Peredachi Informatsii*, vol. 23, no. 2, pp. 9–16, 1987

Assume i.i.d. $\mathbf{H}_{(i,j)}$ for i-th (j-th) row (col).

The differential entropy of the (i, j)-th element is

$$h(\mathbf{H}_{(i,j)}) = -\int p(\mathbf{H}(i,j) = k) \log p(\mathbf{H}_{(i,j)} = k) dk,$$

Resort to Kozachenko–Leonenko (KL) estimator [11]. Average over elements in $\mathbf{H}$,

$$\hat{h}(\mathbf{H}) = \frac{1}{R_d n_T} \sum_{i}^{R_d} \sum_{j}^{n_T} \hat{h}(\mathbf{H}_{(i,j)}),$$

for KL estimator $\hat{h}$.

L. Kozachenko and N. N. Leonenko, "Sample Estimate of the Entropy of a Random Vector," *Problemy Peredachi Informatsii*, vol. 23, no. 2, pp. 9–16, 1987

Theorem 8.3.1 from [12] – for small interval $\Delta = \frac{1}{2^b}$, entropy of quantized r.v. related to its differential entropy as,

$$H(\mathbf{H}^{\Delta}) = h(\mathbf{H}) + b,$$

T. M. Cover, *Elements of Information Theory*. John Wiley & Sons, 1999

Quantized vs. Differential Entropy

Theorem 8.3.1 from [12] – for small interval $\Delta = \frac{1}{2^b}$, entropy of quantized r.v. related to its differential entropy as,

$$H(\mathbf{H}^{\Delta}) = h(\mathbf{H}) + b,$$

Thus, the differential entropy estimator admits an estimate for the entropy of the quantized CSI, $\hat{\mathbf{H}}^{\Delta}$.

T. M. Cover, *Elements of Information Theory*. John Wiley & Sons, 1999

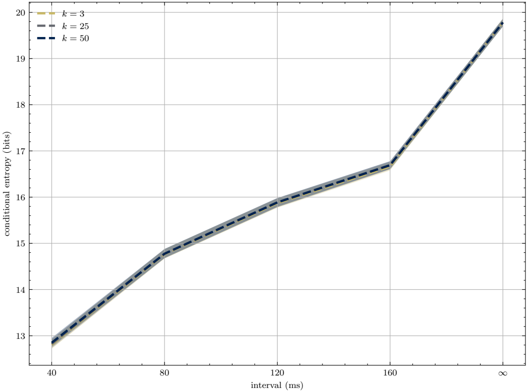
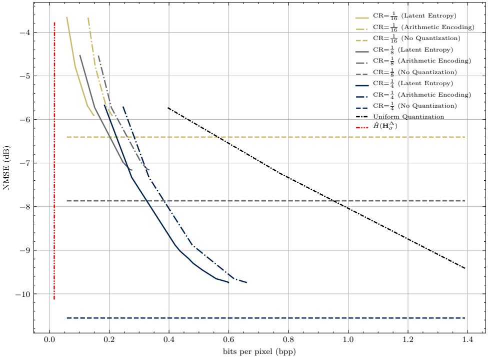


Figure: Mean conditional entropy estimates vs. feedback interval $\hat{H}(\mathbf{H}^{\Delta}) = \hat{h}(\mathbf{H}) + n$ with 95% c.i.

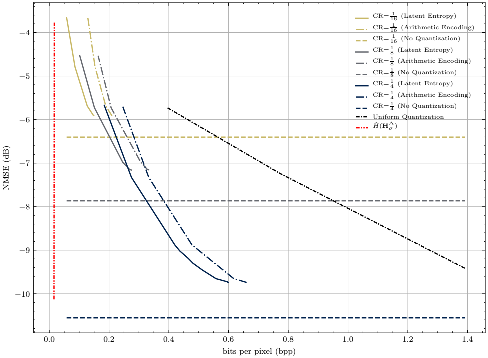


- Entropy bound for CSI compression

M. del Rosario and Z. Ding, “Trainable Codewords and Compression Bounds for Deep Learning-based Multi-Antenna CSI Feedback,” May 2021.

Submitted to IEEE GLOBECOM 2021

Results: Rate-Distortion Bound



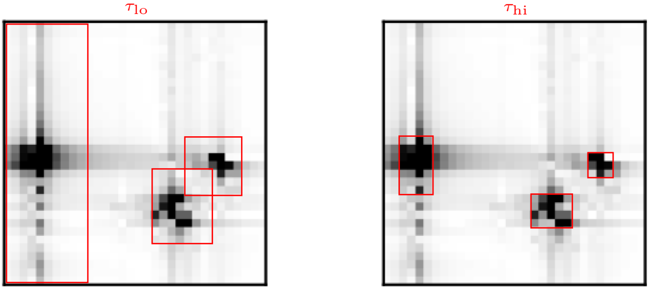
- Entropy bound for CSI compression
- **Future Work: Achieving rate-optimal compression**

M. del Rosario and Z. Ding, “Trainable Codewords and Compression Bounds for Deep Learning-based Multi-Antenna CSI Feedback,” May 2021.

Submitted to IEEE GLOBECOM 2021

Future Work: Region of Interest Compression

$$L_{\text{MSE,ROI}} = \frac{1}{N_{\text{ROI}}} \sum_{i \in \mathbf{S}} \left\| \mathbf{H}_i - g(f(\mathbf{H}_i, \theta_e), \theta_d) \right\|^2.$$



(a) Low threshold (b) High threshold

Figure: Hypothetical bounding boxes based on threshold, τ, where $\tau_{\text{lo}} < \tau_{\text{hi}}$. The set of ROI pixels constitute $\mathbf{S}$.

Future Work: Region of Interest Compression

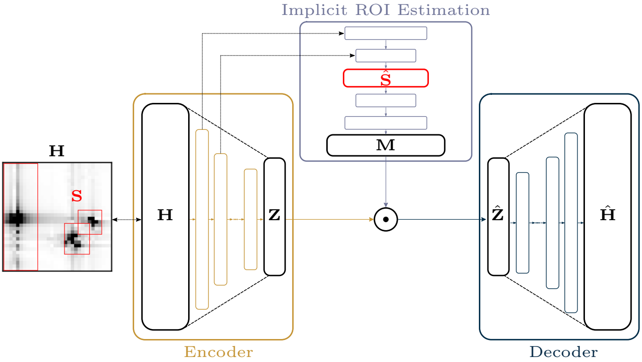


Figure: Abstract architecture for potential ROI-based compression network for CSI estimation.

Future Work: Temporal Soft Quantization

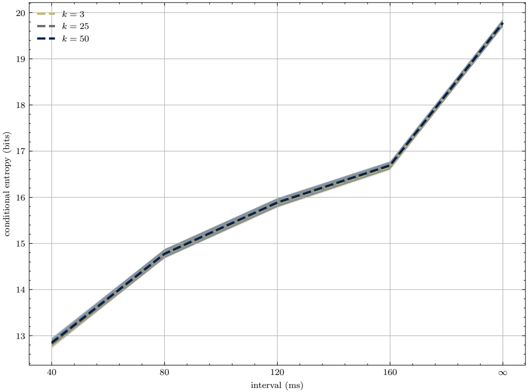


Figure: Shorter feedback intervals → lower conditional entropy.

Future Work: Temporal Soft Quantization

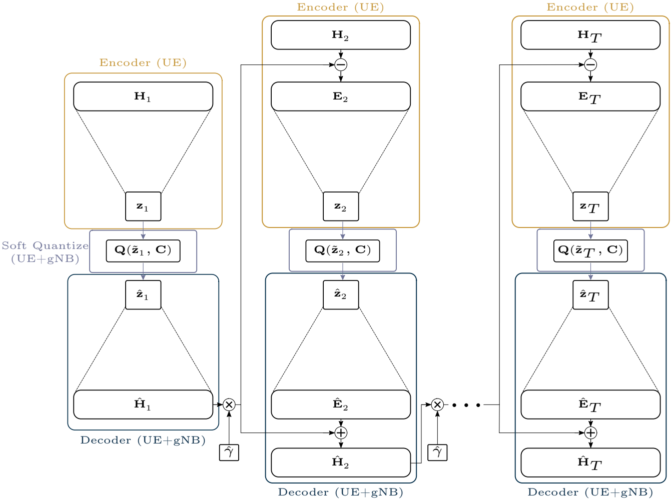


Figure: Abstract architecture for MarkovNet with SoftQuant layers.

- Z. Liu, **M. del Rosario**, X. Liang, L. Zhang, and Z. Ding, "Spherical Normalization for Learned Compressive Feedback in Massive MIMO CSI Acquisition," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2020
- Z. Liu †, **M. del Rosario †**, and Z. Ding, "A Markovian Model-Driven Deep Learning Framework for Massive MIMO CSI Feedback," *arXiv e-prints*, Sept. 2020. Submitted to IEEE Transactions on Wireless Communications
- **M. del Rosario** and Z. Ding, "Trainable Codewords and Compression Bounds for Deep Learning-based Multi-Antenna CSI Feedback," May 2021. Submitted to IEEE GLOBECOM 2021

Acknowledgements

- QE Committee

Acknowledgements

- QE Committee
- Prof. Ding, lab mates, collaborators

Acknowledgements

- QE Committee
- Prof. Ding, lab mates, collaborators
- My parents, my brother

Acknowledgements

- QE Committee
- Prof. Ding, lab mates, collaborators
- My parents, my brother
- My SO

Questions?

mdelrosa@ucdavis.edu

[1] E. Björnson, E. G. Larsson, and T. L. Marzetta, “Massive MIMO: Ten Myths and One Critical Question,” *IEEE Communications Magazine*, vol. 54, no. 2, pp. 114–123, 2016.

[2] E. Crespo Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, “A Review of Sparse Recovery Algorithms,” *IEEE Access*, vol. 7, pp. 1300–1322, 2019.

[3] 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.

[4] Z. Liu, **M. del Rosario**, X. Liang, L. Zhang, and Z. Ding, “Spherical Normalization for Learned Compressive Feedback in Massive MIMO CSI Acquisition,” in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2020.

[5] 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, pp. 92–99, December 2012.

[6] T. Wang, C. Wen, S. Jin, and G. Y. Li, “Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels,” *IEEE Wireless Comm. Letters*, vol. 8, pp. 416–419, April 2019.

[7] Z. Liu †, **M. del Rosario †**, and Z. Ding, “A Markovian Model-Driven Deep Learning Framework for Massive MIMO CSI Feedback,” *arXiv e-prints*, Sept. 2020.

Submitted to IEEE Transactions on Wireless Communications.

[8] Q. Yang, M. B. Mashhadi, and D. Gündüz, “Deep Convolutional Compression For Massive MIMO CSI Feedback,” in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2019.

[9] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. Van Gool, “Soft-to-hard Vector Quantization for End-to-end Learning Compressible Representations,” *Advances in Neural Information Processing Systems*, vol. 2017-Decem, no. NIPS, pp. 1142–1152, 2017.

[10] **M. del Rosario** and Z. Ding, “Trainable Codewords and Compression Bounds for Deep Learning-based Multi-Antenna CSI Feedback,” May 2021.

Submitted to IEEE GLOBECOM 2021.

[11] L. Kozachenko and N. N. Leonenko, “Sample Estimate of the Entropy of a Random Vector,” *Problemy Peredachi Informatsii*, vol. 23, no. 2, pp. 9–16, 1987.

[12] T. M. Cover, *Elements of Information Theory.*

John Wiley & Sons, 1999.

Appendix

D-AMP = Denoising approximate message passing. Initialize $x^0 = \mathbf{0}$, and alternate between:

$$x^{t+1} = D_{\hat{\sigma}^t}(x^t + \mathbf{A}^* z^t)$$

$$z^t = y - \mathbf{A}x^t + z^{t-1} \frac{\mathrm{div} D_{\hat{\sigma}^{t-1}}(x^{t-1} + \mathbf{A}^* z^{t-1})}{m}$$

where $\hat{\sigma}^t = \mathrm{Var}(x^t + \mathbf{A}^* z^t)$, $D_{\hat{\sigma}_t}$ = denoising algorithm.

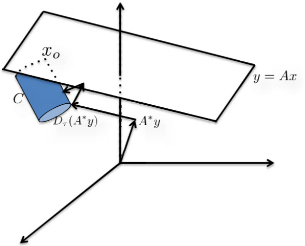


Figure: Subspaces of interest in D-AMP.

BM3D-AMP: Definition

BM3D-AMP = D-AMP with *block matching 3D collaborative filtering (BM3D)*.

- Combination of non-local means (NLM) and wavelet thresholding.
- Procedure:
  1. Compare patches of pixels in images
  2. Group similar patches
  3. 2D (DCT or Bior Wavelet) + 1D Haar wavelet transforms on group
  4. Shrink coefficients in groups ($N \to M$)
  5. Perform inverse transform by 1) hard thresholding and 2) Wiener filter ($M \to N$)

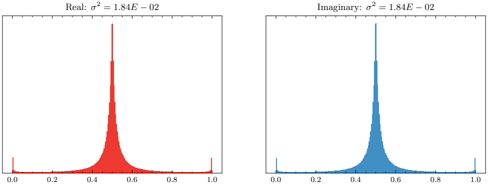
Tanh Normalization

Given mean μ, standard deviation σ w.r.t $\mathbf{H}$,

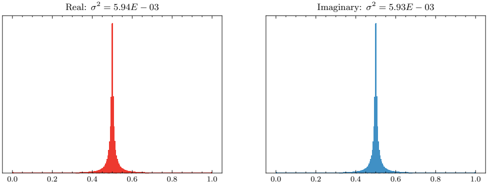
$$H_{\tanh}(i,j) = \tanh\left(\frac{H(i,j)-\mu}{2\nu\sigma}\right) + 1.$$

Scale parameter ν chosen by designer.

COST2100 (Indoor): Tanh Normalization



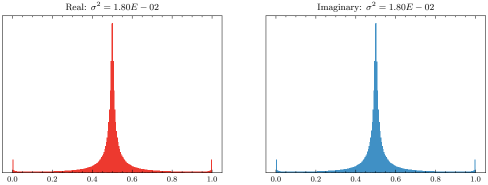
(a) $\nu = 1$



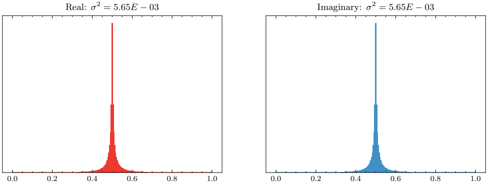
(b) $\nu = 3$

Figure: Distribution/variance of indoor COST2100 real/imaginary channels under tanh normalization ($N = 9.9\dot{1}0^5$).

COST2100 (Outdoor): Tanh Normalization



(a) $\nu = 1$



(b) $\nu = 3$

Figure: Distribution/variance of outdoor COST2100 real/imaginary channels under tanh normalization ($N = 10^5$).

Rather than scalar $\hat{\gamma} \in \mathbb{R}^+$, we can derive a multivariate p-step predictor, $\mathbf{W}_1, \ldots, \mathbf{W}_p$. Given p prior CSI samples, the mean-square optimal predictor $\hat{H}_t$ is a linear combination of these the prior CSI samples,

$$\hat{\mathbf{H}}_t = \mathbf{H}_{t-1}\mathbf{W}_1 + \cdots + \mathbf{H}_{t-p}\mathbf{W}_p + \mathbf{E}_t. \tag{5}$$

Error terms are uncorrelated with the CSI samples (i.e. $\mathbf{H}_{t-i}^{H}\mathbf{E}_t = 0$ for all $i \in [0, \ldots, p]$), and we pre-multiply by $\mathbf{H}_{t-i}^{H}$,

$$\begin{aligned} \mathbf{H}_{t-i}^{H}\hat{\mathbf{H}}_t &= \mathbf{H}_{t-i}^{H}\mathbf{H}_{t-1}\mathbf{W}_1 + \cdots + \mathbf{H}_{t-i}^{H}\mathbf{H}_{t-p}\mathbf{W}_p + \mathbf{H}_{t-i}^{H}\mathbf{E}_t \\ &= \mathbf{H}_{t-i}^{H}\mathbf{H}_{t-1}\mathbf{W}_1 + \cdots + \mathbf{H}_{t-i}^{H}\mathbf{H}_{t-p}\mathbf{W}_p. \end{aligned} \tag{6}$$

Denote the correlation matrix $\mathbf{R}_i = \mathbb{E}[\mathbf{H}_{t-i}^H \mathbf{H}_t]$. Presume CSI matrices arise from a stationary process, implying the following properties:

1. $\mathbf{R}_i = \mathbb{E}[\mathbf{H}_{t-i}^H \mathbf{H}_t] = \mathbb{E}[\mathbf{H}_t^H \mathbf{H}_{t+i}]$
2. $\mathbf{R}_i = \mathbf{R}_{-i}^H$

Taking the expectation, write (6) as a linear combination of $\mathbf{R}$,

$$\mathbf{R}_{i+1} = \mathbf{R}_i \mathbf{W}_1 + \cdots + \mathbf{R}_{i-p+1} \mathbf{W}_p.$$

For p CSI samples, write a system of p equations, admitting the following,

$$\begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \dots \\ \mathbf{R}_p \end{bmatrix} = \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_1^H & \dots & \mathbf{R}_{p-1}^H \\ \mathbf{R}_1 & \mathbf{R}_0 & \dots & \mathbf{R}_{p-2}^H \\ \vdots & & \ddots & \vdots \\ \mathbf{R}_{p-1} & \mathbf{R}_{p-2} & \dots & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \dots \\ \mathbf{W}_p \end{bmatrix}.$$

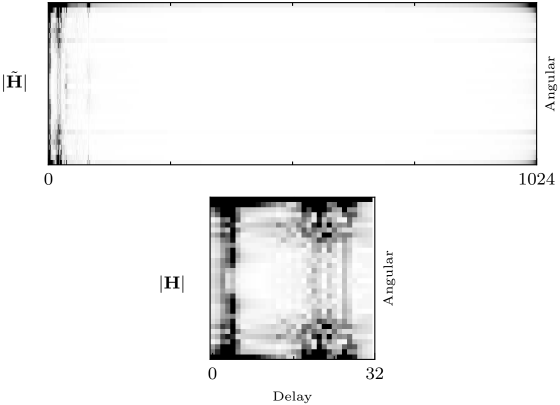
Solving for the coefficient matrices admits the solution

$$\begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \dots \\ \mathbf{W}_p \end{bmatrix} = \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_1^H & \dots & \mathbf{R}_{p-1}^H \\ \mathbf{R}_1 & \mathbf{R}_0 & \dots & \mathbf{R}_{p-2}^H \\ \vdots & & \ddots & \vdots \\ \mathbf{R}_{p-1} & \mathbf{R}_{p-2} & \dots & \mathbf{R}_0 \end{bmatrix}^+ \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \dots \\ \mathbf{R}_p \end{bmatrix}, \tag{7}$$

where $[\cdot]^+$ denotes the Moore-Penrose pseudoinverse.

NMSE: All vs. Truncated

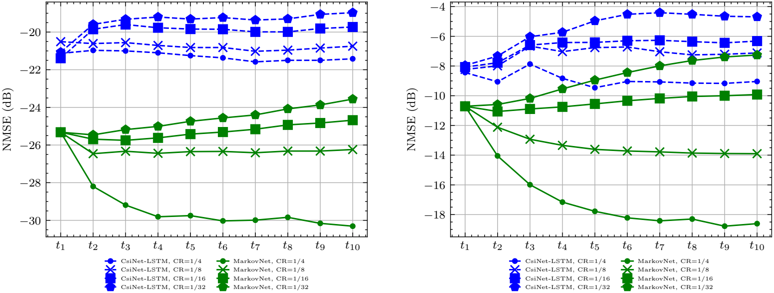
$$\mathrm{NMSE}_{\mathrm{all}} = \frac{1}{N}\sum_{i}^{N}\frac{\|\tilde{\mathbf{H}}_i - \hat{\tilde{\mathbf{H}}}_i\|^2}{\|\tilde{\mathbf{H}}_i\|^2},\ \mathrm{NMSE}_{\mathrm{truncate}} = \frac{1}{N}\sum_{i}^{N}\frac{\|\mathbf{H}_i - \hat{\mathbf{H}}_i\|^2}{\|\mathbf{H}_i\|^2},$$



MarkovNet: All vs. Truncated

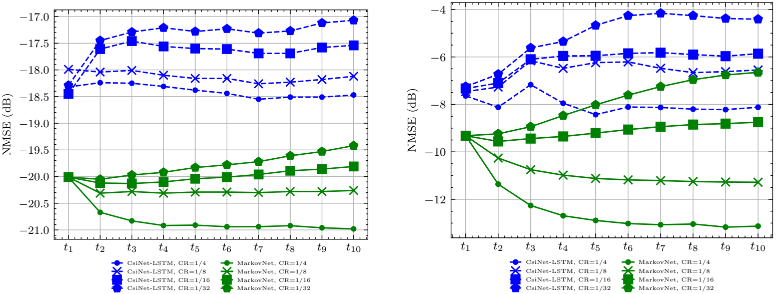
		MarkovNet		**CsiNet-LSTM**	
Env	**CR**	$\text{NMSE}_{\text{truncate}}$	NMSE_{all}	$\text{NMSE}_{\text{truncate}}$	NMSE_{all}
Indoor	$\frac{1}{4}$	-29.26	-20.81	-21.28	-18.4
	$\frac{1}{8}$	-26.25	-20.26	-20.76	-18.12
	$\frac{1}{16}$	-25.27	-19.99	-19.96	-17.67
	$\frac{1}{32}$	-24.62	-19.78	-19.41	-17.34
Outdoor	$\frac{1}{4}$	-16.8	-12.4	-8.89	-7.99
	$\frac{1}{8}$	-13.19	-10.86	-7.17	-6.60
	$\frac{1}{16}$	-10.45	-9.13	-6.65	-6.15
	$\frac{1}{32}$	-8.87	-7.92	-5.33	-4.99

Table: NMSE of truncated vs. full CSI matrices.



(a) Indoor (b) Outdoor

Figure: NMSE$_{\text{truncated}}$ comparison of MarkovNet and CsiNet-LSTM at various compression ratios (CR).



(a) Indoor (b) Outdoor

Figure: $\mathrm{NMSE_{all}}$ comparison of MarkovNet and CsiNet-LSTM at various compression ratios (CR).

Results: Rate-Distortion (Minmax, Soft vs. Hard)

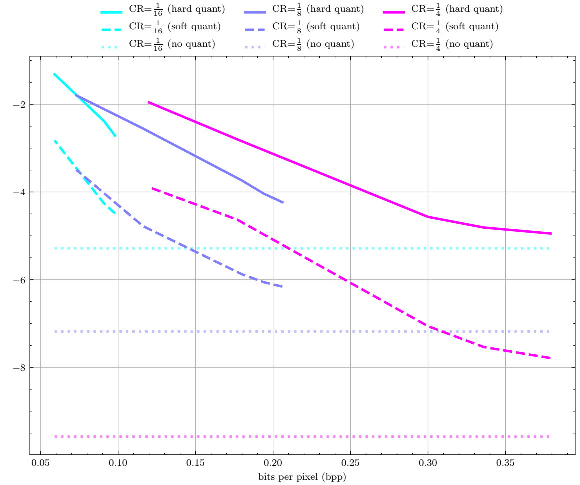


Figure: Rate distortion of CsiNet-SoftQuant under minmax normalization using: $L = 1024$ centers, $d = 4$.

Results: Rate-Distortion (Spherical, Soft vs. Hard)

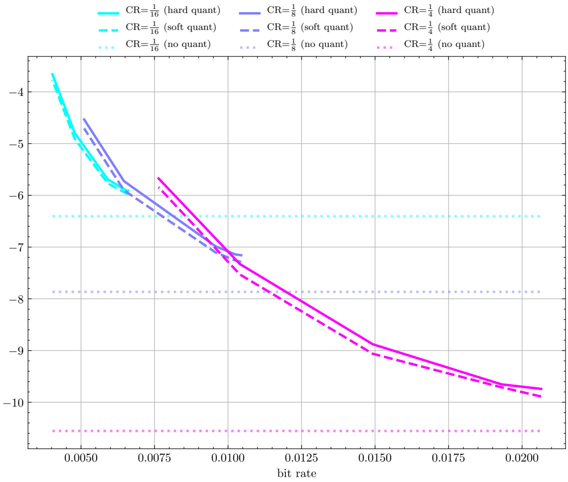


Figure: Rate distortion of CsiNet-SoftQuant using: $L = 1024$ centers, CR$= \frac{1}{4}$, $d = 4$. Bit rates are realized under arithmetic coding of quantized features.

Using an estimator , with additive Gaussian noise, v

$$H_{\sigma,(i,j)} = H_{(i,j)} + v \text{ for i.i.d } v \sim \mathcal{N}(0, \sigma^2).$$

Denote corrupted CSI matrices $\mathbf{H}_\sigma = \left[H_{\sigma,(i,j)}\right]_{i \in [R_d], j \in [N_b]}$.

For different noise levels σ, calculate bounds $\hat{H}(\mathbf{H}_\sigma^\Delta)$ to establish a rate-distortion curve.

Given probability measure P on $\mathbb{R}^d$ with density p, the entropy is

$$H(p) = -\int_{\mathbb{R}^d} p(x) \log p(x) dx$$

For $N \geq 1$, i.i.d. $X_1, \ldots, X_{N+1} \sim P$, we have the following for $i = 1, \ldots, N+1$

$$R_i^N = \min\{|X_i - X_j| : j = 1, \ldots, N+1, j \neq i\}$$
$$Y_i^N = N(R_i^N)^d$$

where $|\cdot|$ is a given norm. Define the following,

$$B(x, r) = \{y \in \mathbb{R}^d : |y - x| \leq r\}$$
$$v_d = \int_{B(0,1)} dx$$
$$\gamma = -\int_0^\infty e^{-x} \log x dx \approx 0.577 \text{(Euler constant)}$$

The Kozachenko-Leonenko (KL) Estimator is,

$$\hat{h} = \frac{1}{N+1} \sum_{i=1}^{N+1} \log Y_i^N + \gamma + \log v_d.$$

Y_i^n can be defined w.r.t. the k-th nearest neighbor, i.e.

$$R_i^N = \mathrm{KNN}(\min\{|X_i - X_j| : j = 1, \ldots, N+1, j \neq i\})$$
$$Y_i^N = N(R_i^N)^d$$

Given CSI data, $\mathbf{H}$, assume i.i.d. $\mathbf{H}_{(i,j)}$ for i-th (j-th) row (col).

Entropy Estimation

Given CSI data, $\mathbf{H}$, assume i.i.d. $\mathbf{H}_{(i,j)}$ for i-th (j-th) row (col).

- Quantized CSI, $\mathbf{H}^{\Delta}$
- Interval $\Delta = \frac{1}{2^b}$ at b bits.
- Entropy of the (i,j)-th element is

$$H(\mathbf{H}^{\Delta}_{(i,j)}) = -\sum_{k}^{2^b} p(\mathbf{H}^{\Delta}_{(i,j)} = k) \log p(\mathbf{H}^{\Delta}_{(i,j)} = k),$$

with histogram estimate $p(\mathbf{H}^{\Delta}_{(i,j)} = k)$.

Given CSI data, $\mathbf{H}$, assume i.i.d. $\mathbf{H}_{(i,j)}$ for i-th (j-th) row (col).

- Quantized CSI, $\mathbf{H}^{\Delta}$
- Interval $\Delta = \frac{1}{2^b}$ at b bits.
- Entropy of the (i,j)-th element is

$$H(\mathbf{H}^{\Delta}_{(i,j)}) = -\sum_{k}^{2^b} p(\mathbf{H}^{\Delta}_{(i,j)} = k) \log p(\mathbf{H}^{\Delta}_{(i,j)} = k),$$

with histogram estimate $p(\mathbf{H}^{\Delta}_{(i,j)} = k)$.

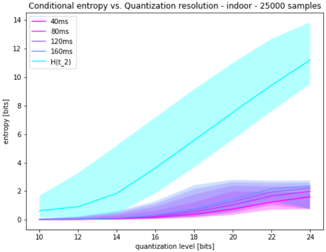
Upper bound on the entropy of the full CSI matrix is

$$H(\mathbf{H}^{\Delta}) = \frac{1}{R_d n_T} \sum_{i}^{R_d} \sum_{j}^{n_T} H(\mathbf{H}^{\Delta}_{(i,j)}).$$

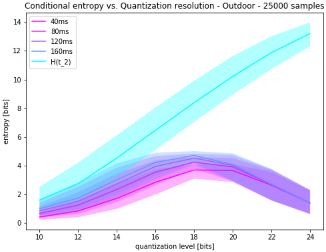
Conditional entropy estimate for quantized CSI defined as,

$$\hat{H}(\mathbf{H}_{t_2}^{\Delta}|\mathbf{H}_{t_1}^{\Delta}) = \hat{H}(\mathbf{H}_{t_2}^{\Delta}, \mathbf{H}_{t_1}^{\Delta}) - \hat{H}(\mathbf{H}_{t_1}^{\Delta})$$

for a feedback interval $t_{\text{interval}} = t_2 - t_1$ s.t. $t_2 > t_1$.



(a) Indoor



(b) Outdoor

Figure: Mean entropy/conditional entropy estimates $H(\mathbf{H}^{\Delta})$ with 95% c.i. for quantized i.i.d COST2100 elements vs. quantization level (bits).

Denote the probability mass in the k-th quantized bin as

$$f(H_k)\Delta = \int_{k\Delta}^{(k+1)\Delta} f(H_{i,j})\, dH_{i,j}.$$

Define the quantized r.v. $H_{i,j}^{\Delta}$ as

$$H_{(i,j)}^{\Delta} = H_k \quad \text{if } k\Delta \leq \mathbf{H}_{(i,j)} < (k+1)\Delta,$$

and the probability that $H_{(i,j)}^{\Delta} = H_k$ is

$$p_k = \int_{k\Delta}^{(k+1)\Delta} f(H_{(i,j)}) dH_{(i,j)} = f(H_k)\Delta$$

T. M. Cover, *Elements of Information Theory*. John Wiley & Sons, 1999

The entropy of the quantized r.v. is

$$
\begin{aligned}
H(H^{\Delta}_{(i,j)}) &= -\sum_{-\infty}^{\infty} p_k \log p_k \\
&= -\sum_{-\infty}^{\infty} f(H_k)\Delta \log f(H_k)\Delta \\
&= -\sum_{-\infty}^{\infty} f(H_k)\Delta \log f(H_k) - \sum_{-\infty}^{\infty} f(H_k)\Delta \log \Delta \\
&= -\sum_{-\infty}^{\infty} f(H_k)\Delta \log f(H_k) - \log \Delta \qquad (8)
\end{aligned}
$$

T. M. Cover, *Elements of Information Theory*. John Wiley & Sons, 1999

The first term in (8) approaches

$$\lim_{\Delta\to 0} -\sum_{-\infty}^{\infty} f(H_k)\Delta \log f(H_k) = -\int f(H_{(i,j)}) \log f(H_{(i,j)}) dH_{(i,j)}$$

$$= h(H_{(i,j)}),$$

leading to the following,

$$\lim_{\Delta\to 0} H(H^{\Delta}_{(i,j)}) = h(H_{(i,j)}) - \log \Delta.$$

Recall that $\Delta = \frac{1}{2^b}$, which implies the following

$$\lim_{b\to\infty} H(H^{\Delta}_{(i,j)}) = h(H_{(i,j)}) + b. \square$$

T. M. Cover, *Elements of Information Theory*. John Wiley & Sons, 1999

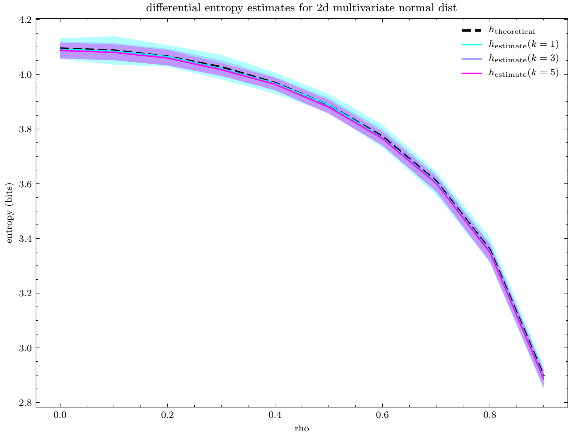


Figure: Differential entropy and estimates for 2d multivariate normal distribution. Estimates are based on the KL estimator [11] using the NPEET library [13].