Deep CNN for Wideband mmWave Massive MIMO Channel Estimation using Frequency Correlation

Peihao Dong[*†], Hua Zhang[*], Geoffrey Ye Li[†],

Navid NaderiAlizadeh[‡], and Ivan Simões Gaspar[‡]

[*]National Mobile Communications Research Laboratory, Southeast University, Nanjing, 210096 China

[†]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA

[‡]Intel Corporation, Santa Clara, CA, USA

Outlines

- Background and Motivation
- System Model
- CNN based Channel Estimation
- Numerical Results
- Conclusions

Background and Motivation

mmWave massive MIMO systems

- Phase shifter based hybrid architecture is widely used to reduce the implementation complexity and cost
- Channel estimation is challenging under this hybrid architecture

Why deep learning (DL)?

- MMSE channel estimation is hindered by the difficulty of acquiring the ideal channel covariance matrix and by the high computational complexity due to the large antenna dimension
- Compressive sensing based methods perform unsatisfactorily in the practical complicated channel and also suffer from high complexity caused by iterations
- DL is more capable to extract the inherent characteristics underlying the channel from the large amount of data and provides the potential to estimate the channel more accurately with lower complexity by using the efficient parallel computing methods

System Model

Transmitter and receiver

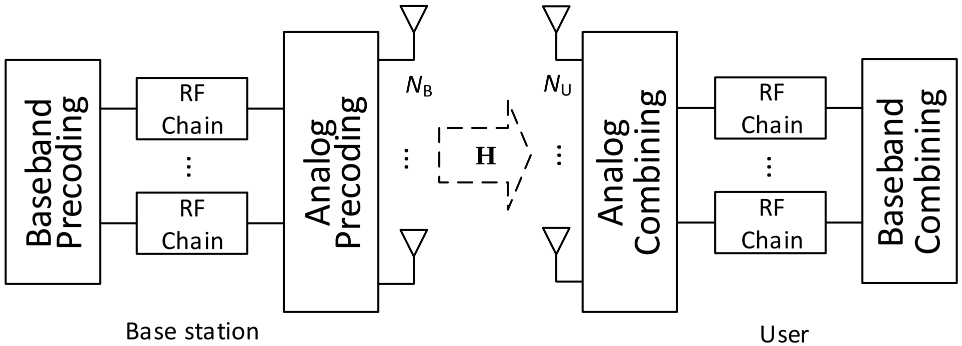

Fig. 1. System model.

Channel model

- The delay domain channel from the BS to the user is denoted as

$$\mathbf{H}(\boldsymbol{\tau}) = \sqrt{\frac{N_{\rm B}N_{\rm U}}{L}} \sum_{l=1}^{L} \alpha_l \delta(\tau - \tau_l)\mathbf{a}_{\rm U}(\varphi_l)\mathbf{a}_{\rm B}^H(\phi_l) \quad (1)$$

  - L is the number of paths, α_l and τ_l are the propagation gain and delay of the lth path, φ_l and ϕ_l are the AoA and AoD at the user and the BS of the lth path
- The frequency domain channel of the kth subcarrier in OFDM is

$$\mathbf{H}_k = \sqrt{\frac{N_{\rm B}N_{\rm U}}{L}} \sum_{l=1}^{L} \alpha_l e^{-j2\pi\tau_l f_s \frac{k}{K}} \mathbf{a}_{\rm U}(\varphi_l)\mathbf{a}_{\rm B}^H(\phi_l) \quad (2)$$

  - f_s denotes the sampling rate and K is the number of OFDM subcarriers

SOUTHEAST UNIVERSITY Georgia Institute of Technology

System Model

❑ Signal model for channel estimation

- To estimate $\mathbf{H}_k$, the BS transmits pilot signal $x_{k,u}$ on the beamforming vector $\mathbf{f}_{k,u} \in \mathbb{C}^{N_\mathrm{B}\times 1}$, $u = 1,\cdots,M_\mathrm{B}$, during M_B successive instants and the user employs M_U combining vectors $\mathbf{w}_{k,v} \in \mathbb{C}^{N_\mathrm{U}\times 1}$, $v = 1,\cdots,M_\mathrm{U}$, to process each beamforming vector.
- The received pilots associated with the kth subcarrier after combining at the user is written as

$$\mathbf{Y}_k = \mathbf{W}_k^H \mathbf{H}_k \mathbf{F}_k \mathbf{X}_k + \widetilde{\mathbf{N}}_k \ , \qquad (3)$$

  - ✓ $\mathbf{W}_k = [\mathbf{w}_{k,1},\cdots,\mathbf{w}_{k,M_\mathrm{U}}]$ and $\mathbf{F}_k = [\mathbf{f}_{k,1},\cdots,\mathbf{f}_{k,M_\mathrm{B}}]$ are combining matrix and beamforming matrix, respectively
  - ✓ $\mathbf{X}_k$ is an $M_\mathrm{B} \times M_\mathrm{B}$ diagonal matrix with its uth diagonal element being $x_{k,u}$
  - ✓ $\widetilde{\mathbf{N}}_k = \mathbf{W}_k^H \mathbf{N}_k$ denotes the effective noise after combining at the user and $\mathbf{N}_k$ is additive white Gaussian noise (AWGN) before combining

SOUTHEAST UNIVERSITY
Georgia Institute of Technology

CNN based Channel Estimation
-Algorithm Description

- **Signal preprocessing**
  - Assume the worst case that $\mathbf{W}_k = \mathbf{W}$, $\mathbf{F}_k = \mathbf{F}$, and $\mathbf{X}_k = \sqrt{P}\mathbf{I}$ for all subcarriers with pilots
  - The received pilot matrix, $\mathbf{Y}_k$, is vectorized as

$$\bar{\mathbf{y}}_k = \mathrm{vec}(\mathbf{Y}_k) = \sqrt{P}(\mathbf{F}^T \otimes \mathbf{W}^H)\mathrm{vec}(\mathbf{H}_k) + \mathrm{vec}\big(\tilde{\mathbf{N}}_k\big)$$

$$= \mathbf{Q}\bar{\mathbf{h}}_k + \bar{\mathbf{n}}_k \quad (4)$$

    - ✓ Specifically, $\mathbf{Q} = \sqrt{P}(\mathbf{F}^T \otimes \mathbf{W}^H)$, $\bar{\mathbf{h}}_\mathbf{k} = \mathrm{vec}(\mathbf{H}_\mathbf{k})$, $\bar{\mathbf{n}}_\mathbf{k} = \mathrm{vec}\big(\tilde{\mathbf{N}}_\mathbf{k}\big)$
  - $\bar{\mathbf{y}}_k$ is further processed and we can obtain the processed pilot matrix at subcarrier k as

$$\mathbf{R}_k = \mathrm{vec}^{-1}(\bar{\mathbf{r}}_k) = \mathrm{vec}^{-1}\big(\mathbf{Q}^\dagger \bar{\mathbf{y}}_k\big) \quad (5)$$

  - The processed pilot matrices at S successive subcarriers, $\mathbf{R}_{k_0}, \mathbf{R}_{k_0+1}, \cdots, \mathbf{R}_{k_0+S-1}$, within one coherence bandwidth will be input into the CNN simultaneously for joint channel estimation

CNN based Channel Estimation
-Algorithm Description

CNN offline training

- Training set: the nth sample has the form of $(\underline{\mathbf{R}}_n, \underline{\mathbf{H}}_n)$, where $\underline{\mathbf{R}}_n, \underline{\mathbf{H}}_n \in \mathbb{C}^{N_\mathrm{U} \times N_\mathrm{B} \times S}$ are the input and target data, respectively, and the sth 2D matrices are $\mathbf{R}^n_{k_0+s-1}$ and $\frac{\mathbf{H}^n_{k_0+s-1}}{c}$, respectively. $\mathbf{R}^n_{k_0+s-1}$ is the processed pilot matrix at subcarrier $k_0 + s - 1$ given by (5) and $\mathbf{H}^n_{k_0+s-1}$ is the corresponding true channel matrix. $c > 0$ is a scaling constant to make the value range of the vast majority of target data match the activation function.

- Basic idea of offline training: Input the tentatively estimated channel matrices of S subcarriers, $\underline{\mathbf{R}}_n$, into the CNN to approximate the corresponding scaled true channels $\underline{\mathbf{H}}_n$. Minimize the MSE loss function over all training samples as

$$\mathrm{MSE}_{\mathrm{Loss}} = \frac{1}{N_{\mathrm{tr}} c^2} \sum_{n=1}^{N_{\mathrm{tr}}} \sum_{s=1}^{S} \left\| \mathbf{H}^n_{k_0+s-1} - \widehat{\mathbf{H}}^n_{k_0+s-1} \right\|_F^2 \tag{6}$$

SOUTHEAST UNIVERSITY Georgia Institute of Technology

CNN based Channel Estimation
-Algorithm Description

❑ Illustration for offline training

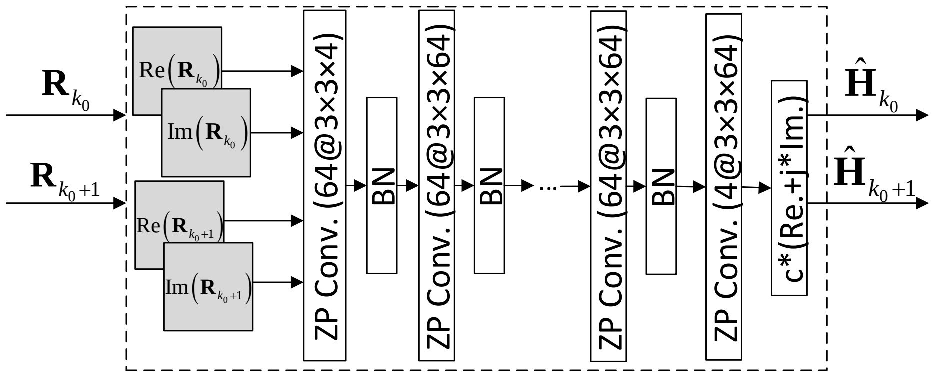

Fig. 2. Proposed CNN for joint channel estimation.

- $N_{\mathrm{B}} = M_{\mathrm{B}} = 32$, $N_{\mathrm{U}} = M_{\mathrm{U}} = 16$, $S = 2$
- $\mathbf{R}_{k_0}^n$, $\mathbf{R}_{k_0+1}^n$ with separated real and imaginary parts are input into CNN
- They are processed by 9 zero padding convolutional layers with ReLU and batch normalization and the output layer with tangent function
- CNN outputs the estimated real and imaginary parts of the scaled channel matrices. Then $\widehat{\mathbf{H}}_{k_0}^n$ and $\widehat{\mathbf{H}}_{k_0+1}^n$ can be obtained
- Calculate MSE in (6) and minimize it for each epoch

CNN based Channel Estimation
-Algorithm Description

- **Online testing**
  - After the centralized training, the CNN will be deployed at the receiver to obtain the estimated channel matrices, $\widehat{\mathbf{H}}_{k_0}, \widehat{\mathbf{H}}_{k_0+1}, \cdots, \widehat{\mathbf{H}}_{k_0+S-1}$, by jointly processing the pilot matrices, $\mathbf{R}_{k_0}, \mathbf{R}_{k_0+1}, \cdots, \mathbf{R}_{k_0+S-1}$.
- **Channel statistic mismatch between training and testing**
  - If the actual channel model differs from that in the training stage, a straightforward solution is fine-tuning but it is hindered by the difficulty to collect the true channel.
  - The offline trained CNN is quite robust to the new channel statistics that are not observed before, which implies that further online fine-tuning might only provide marginal performance improvement and hence is not necessary.

東南大學 SOUTHEAST UNIVERSITY Georgia Institute of Technology

CNN based Channel Estimation
-Complexity analysis

- **Complexity of CNN based approach**
  - Metric: floating point operations (FLOPs)
  - Complexity of preprocessing in (4) and (5):

$$C_{\mathrm{CNN},1} \sim \mathcal{O}\left(S N_{\mathrm{B}}^2 N_{\mathrm{U}}^2\right) \tag{7}$$

  - Complexity of CNN testing:

$$C_{\mathrm{CNN},2} \sim \mathcal{O}\left(\sum_{l=1}^{L_c} M_{1,l} M_{2,l} F_l^2 N_{l-1} N_l\right) \tag{8}$$

  - The total complexity of proposed CNN based approach:

$$C_{\mathrm{CNN}} \sim \mathcal{O}\left(S N_{\mathrm{B}}^2 N_{\mathrm{U}}^2 + \sum_{l=1}^{L_c} M_{1,l} M_{2,l} F_l^2 N_{l-1} N_l\right) \tag{9}$$

SOUTHEAST UNIVERSITY 東南大學 Georgia Institute of Technology

CNN based Channel Estimation
-Complexity analysis

- **Complexity of MMSE channel estimation**
  - Complexity of least square (LS) channel estimation:
$$C_{\mathrm{MMSE},1} \sim \mathcal{O}\left(S N_{\mathrm{B}}^{2} N_{\mathrm{U}}^{2}\right) \tag{10}$$
  - Complexity of refining the LS channel estimation:
$$C_{\mathrm{MMSE},2} \sim \mathcal{O}\left(S^{3} N_{\mathrm{B}}^{3} N_{\mathrm{U}}^{3}\right) \tag{11}$$
  - The total complexity of MMSE channel estimation:
$$C_{\mathrm{MMSE}} \sim \mathcal{O}\left(S^{3} N_{\mathrm{B}}^{3} N_{\mathrm{U}}^{3}\right) \tag{12}$$

- **CNN based approach vs. MMSE**

l	$M_{1,l}$	$M_{2,l}$	F_l	N_{l-1}	N_l
1	16	32	3	4	64
2~9	16	32	3	64	64
10	16	32	3	64	4

  - $C_{\mathrm{CNN}} \sim \mathcal{O}(10^8)$, $C_{\mathrm{MMSE}} \sim \mathcal{O}(10^9)$

Simulation Results

Simulation settings

- System parameters:
  - Channel model: 3GPP TR 38.901 Release 15

Parameter	Setting value
$N_{\mathrm{B}}, M_{\mathrm{B}}$	32
$N_{\mathrm{U}}, M_{\mathrm{U}}$	16
f_c	28GHz
K	64
f_{s}	100MHz
L	3

- CNN settings:

Settings for proposed NN	
Training set	81,000
Validation set	9,000
Testing set	19,000
Optimizer	adam
Epochs	800
Learning rate	10^{-4} (200 epochs)→5×10^{-5} (400 epochs) →10^{-5} (200 epochs)
Batch size	128
CNN structure	Layer 1: 64@3×3×4(Relu) Layer 2~9: 64@3×3×64 (Relu) Layer 10: 4@3×3×64 (tanh)
S	2
c	2

SOUTHEAST UNIVERSITY Georgia Institute of Technology

Simulation Results

❑ Normalized Mean-Squared Error Performance:

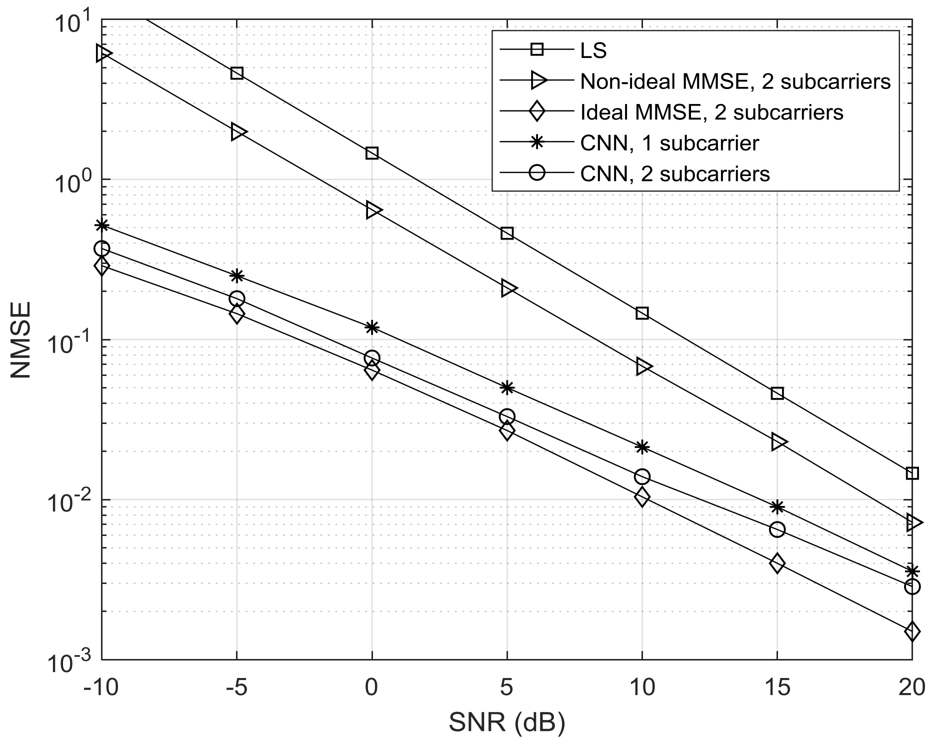

Fig. 3. NMSE versus transmit SNR for the proposed CNN based channel estimation and the existing methods.

- ✓ Urban micro (UMi) street non-line of sight (NLOS) scenario
- ✓ Frequency correlation is helpful to improve the channel estimation accuracy
- ✓ Through offline training, the CNN based channel estimation outperforms the non-ideal MMSE with estimated covariance matrix significantly yet requiring lower estimation complexity
- ✓ The performance of the CNN based approach is very close to the ideal MMSE with true covariance matrix, especially at the low and medium SNRs

Simulation Results

❑ Normalized Mean-Squared Error Performance:

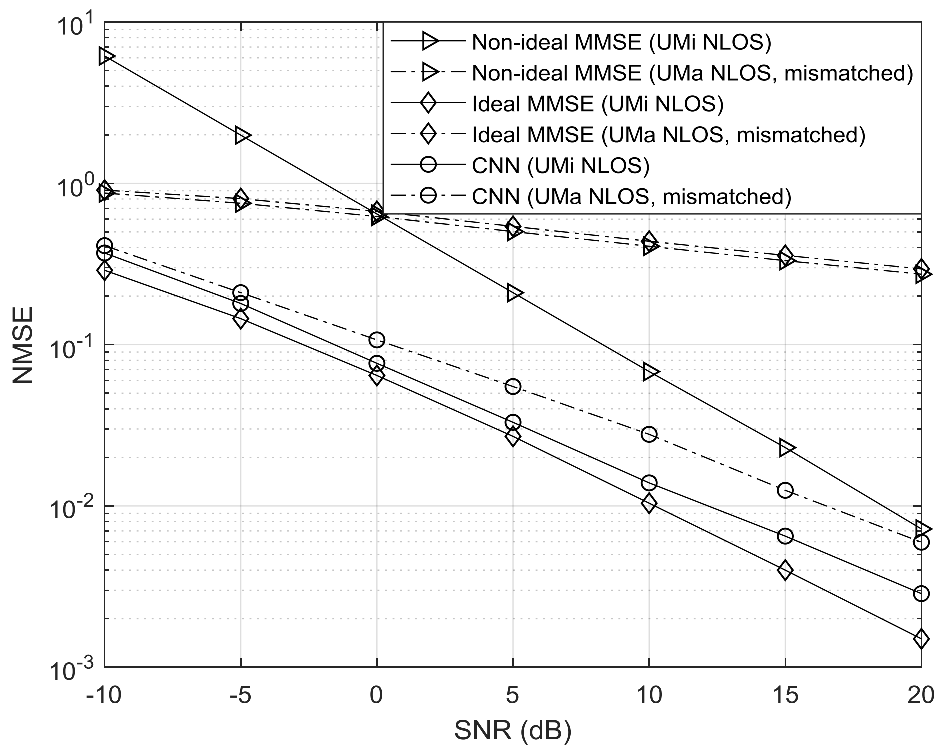

Fig. 4. Robustness for different scenarios.

- ✓ **CNN:**
  Training in UMi scenario
  Testing in both UMi and urban macro (UMa) scenarios
- ✓ **MMSE:**
  Estimating covariance matrix in UMi scenario
  Estimating channel matrix both UMi and UMa scenarios
- ✓ CNN based channel estimation exhibits good robustness when facing the significantly different channel statistics. Even under the mismatched UMa NLOS scenario, the CNN based approach still outperforms the non-ideal MMSE without mismatch.

Conclusions

- We propose a deep CNN based joint channel estimation approach over multiple adjacent subcarriers for mmWave massive MIMO-OFDM systems.
- The proposed approach is with reduced complexity but outperforms the non-ideal MMSE and is close to the ideal MMSE.
- In the case with channel statistics mismatch, the proposed approach exhibits good robustness and outperforms the mismatched ideal and non-ideal MMSE significantly.

Thank you for listening!