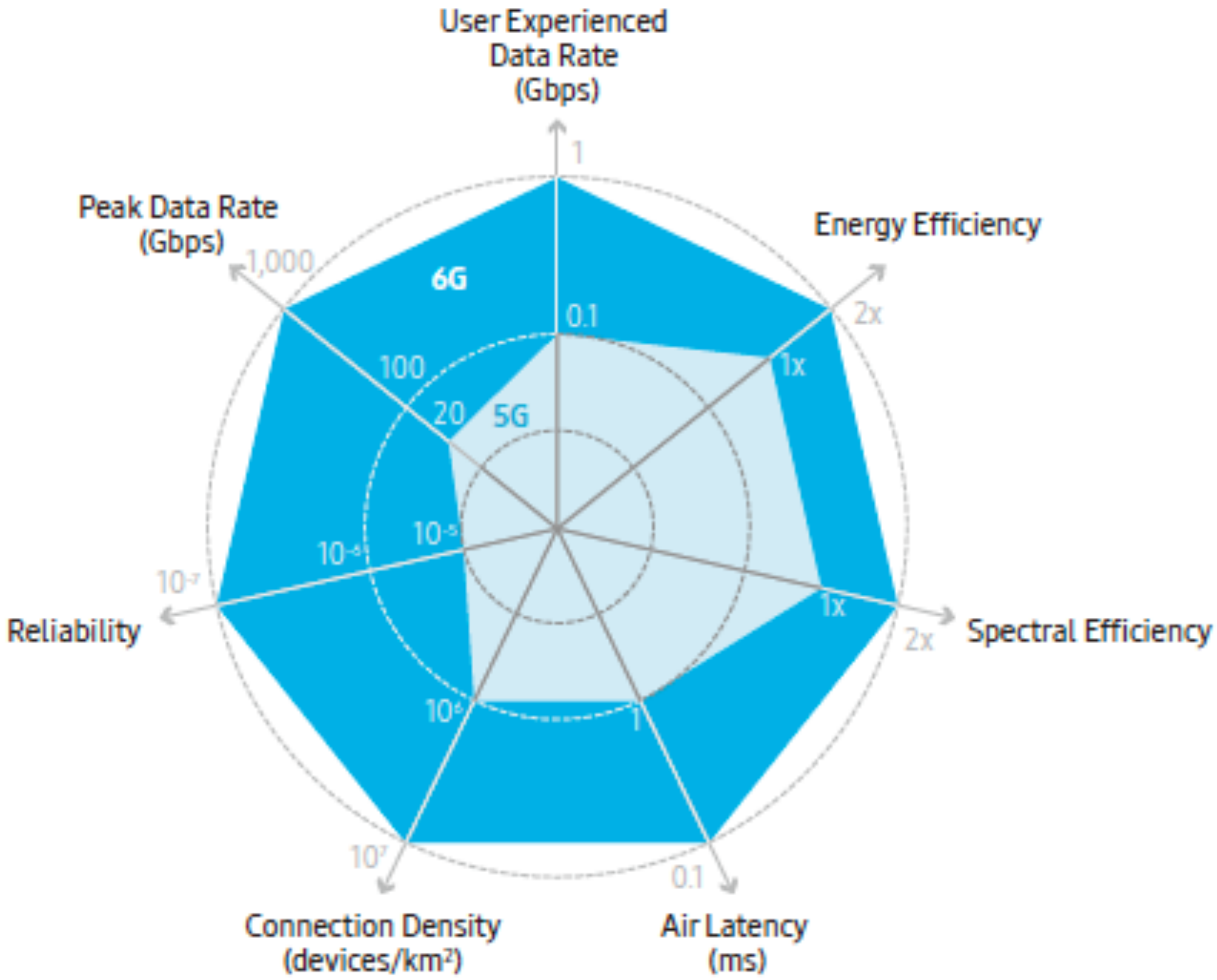
Deep Learning Aided Intelligent Sensing and Identification for Secure Wireless Communications : Part I

Tomoaki Ohtsuki

Dept. of Information and Computer Sciences
Faculty of Science and Technology
Keio University, Japan

6G



Samsung 6G, 2020

Challenges for 6G

- Big data from mobile devices and IoT
- Complex and dynamically changing environments
- High performance requirements
  - Peak data rate: 1 Tbps
  - Connection density: $10/m^2$
  - Latency: 0.1 ms
  - …

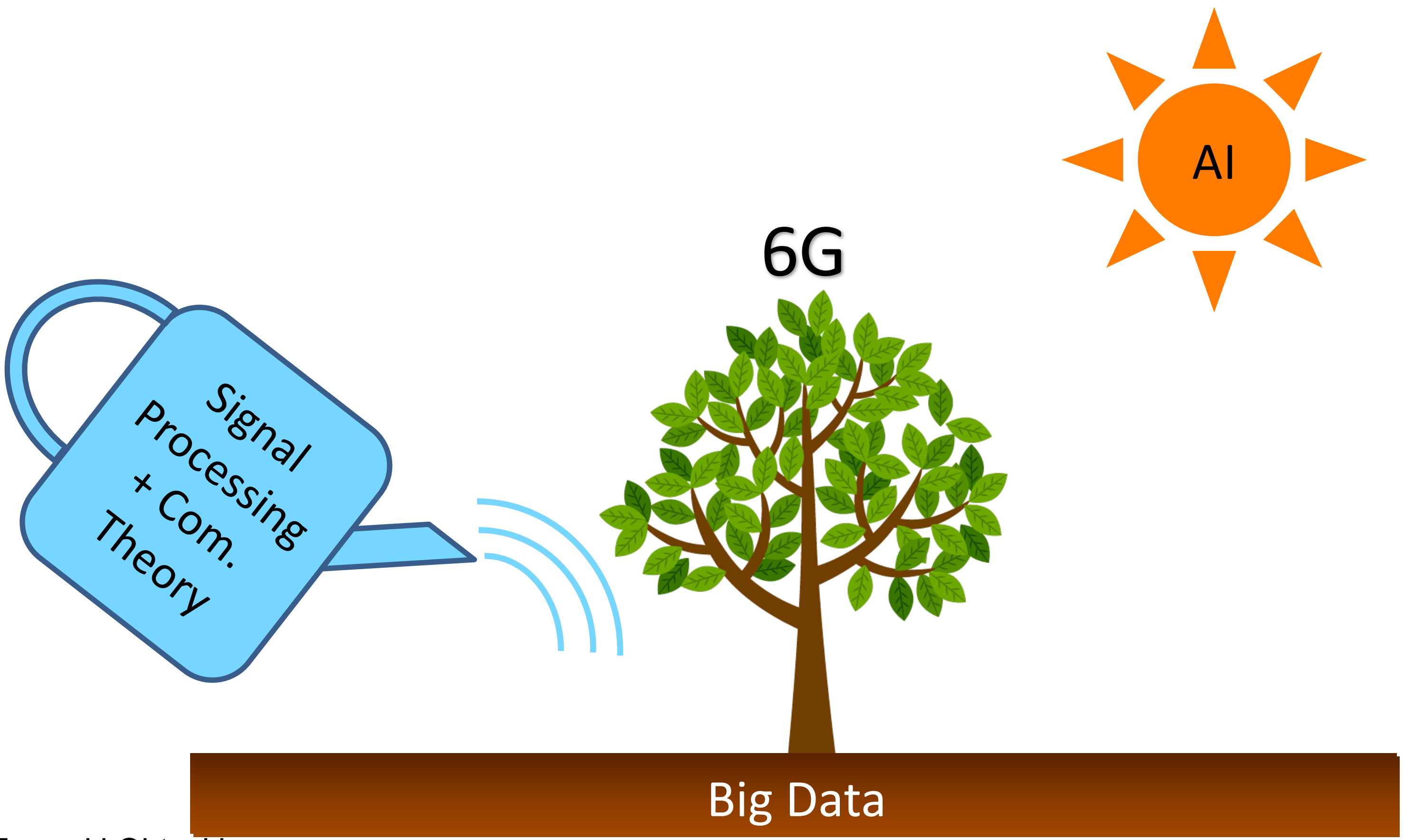
Six Key Technologies for 6G

1. AI/ML-driven air interface design and optimization
2. Expansion into new spectrum bands and new cognitive spectrum sharing methods
3. The integration of localization and sensing capabilities into system definition
4. The achievement of extreme performance requirements on latency and reliability
5. New network architecture paradigms involving sub-networks and RAN-core convergence
6. New security and privacy schemes

Nokia 6G, 2020

AI is a Key Enabler of 6G

- AI has great potentials for addressing big data issues to achieve high requirements in complex and dynamic environments (but with signal processing, communication theory and so on)



Deep Learning

- A neural network (NN) is a multilayer perception that defines a mapping of an input vector $\mathbf{x} \in \mathbb{R}^n$ to an output vector $\mathbf{y} \in \mathbb{R}^k$

$$\hat{f}(\mathbf{x}, \boldsymbol{\theta}) : \mathbb{R}^n \mapsto \mathbb{R}^k$$

- $\boldsymbol{\theta}$: parameters that determine the behavior of the NN
- DL describes the process of finding good values for $\boldsymbol{\theta}$ from data to achieve a desired behavior

J. Hoydis and S. Cammerer, IEEE GCOM2018

Deep Learning-based Wireless Communications

- Wireless communication systems: complex, many imperfections and nonlinearities.
  - DL-based communication systems (or processing block) that does not require a mathematically tractable model might be able to better optimize for such imperfections.

- Conv. block-based optimization does not always provide a best possible end-to-end performance.
- Joint optimization of signal processing blocks is often computationally prohibitive.
  - A leaned end-to-end optimization can provide a superior performance.

Deep Learning-based Wireless Communications

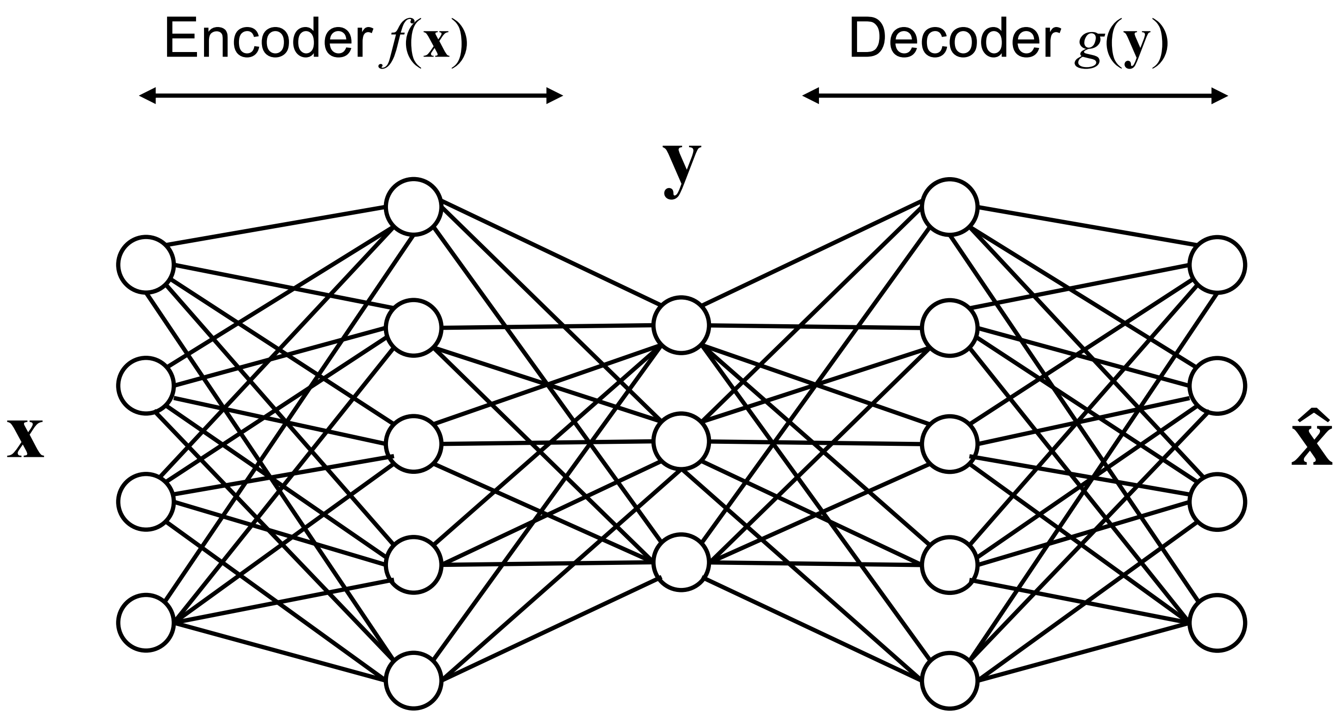
- Physical Layer
  - Signal processing
- Data Link Layer
- Network Layer
  - Routing, scheduling, resource allocation
- Network Security
  - Flow identification
  - Intrusion detection
- Network Level Mobile Data Analysis
- App-Level Mobile Data Analysis

Deep Learning-based Wireless Communications

Physical Layer

- Modulation classification
- Error correction coding/decoding
- Interference management
- MIMO detection
- MIMO precoding
- Channel estimation
- Noise estimation
- …

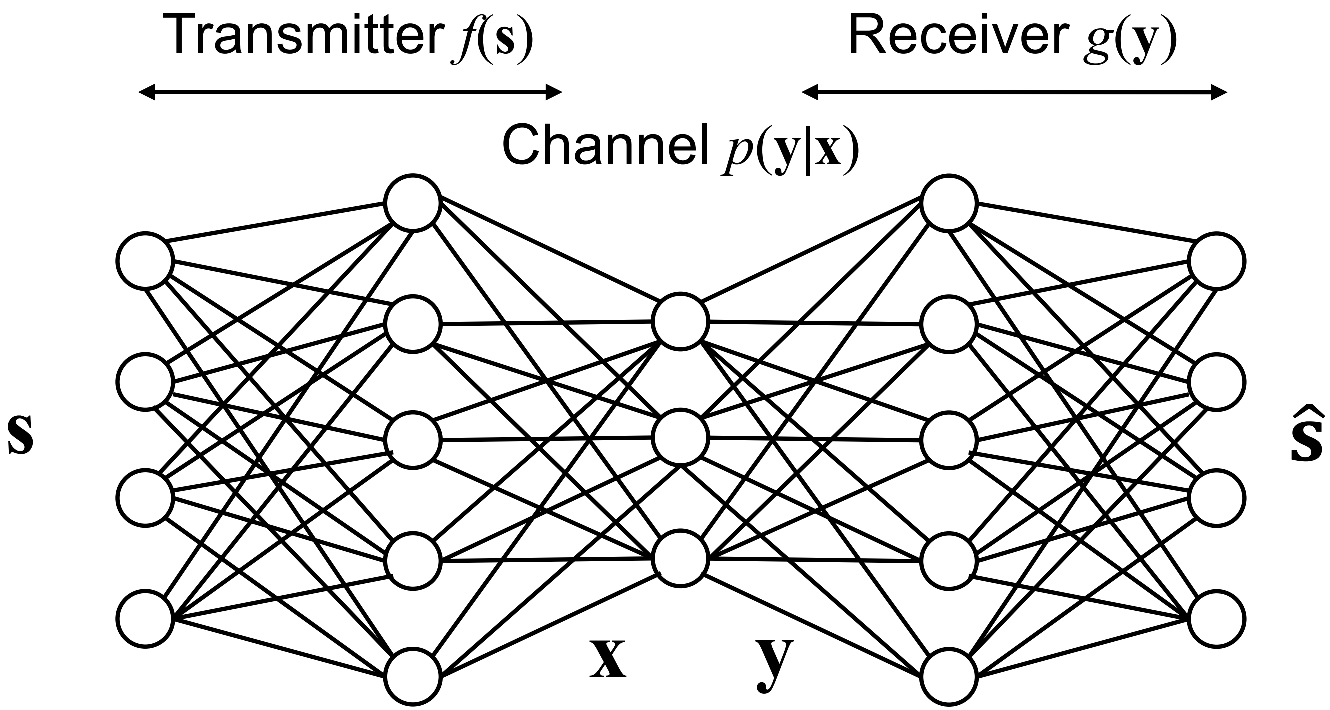
Autoencoder



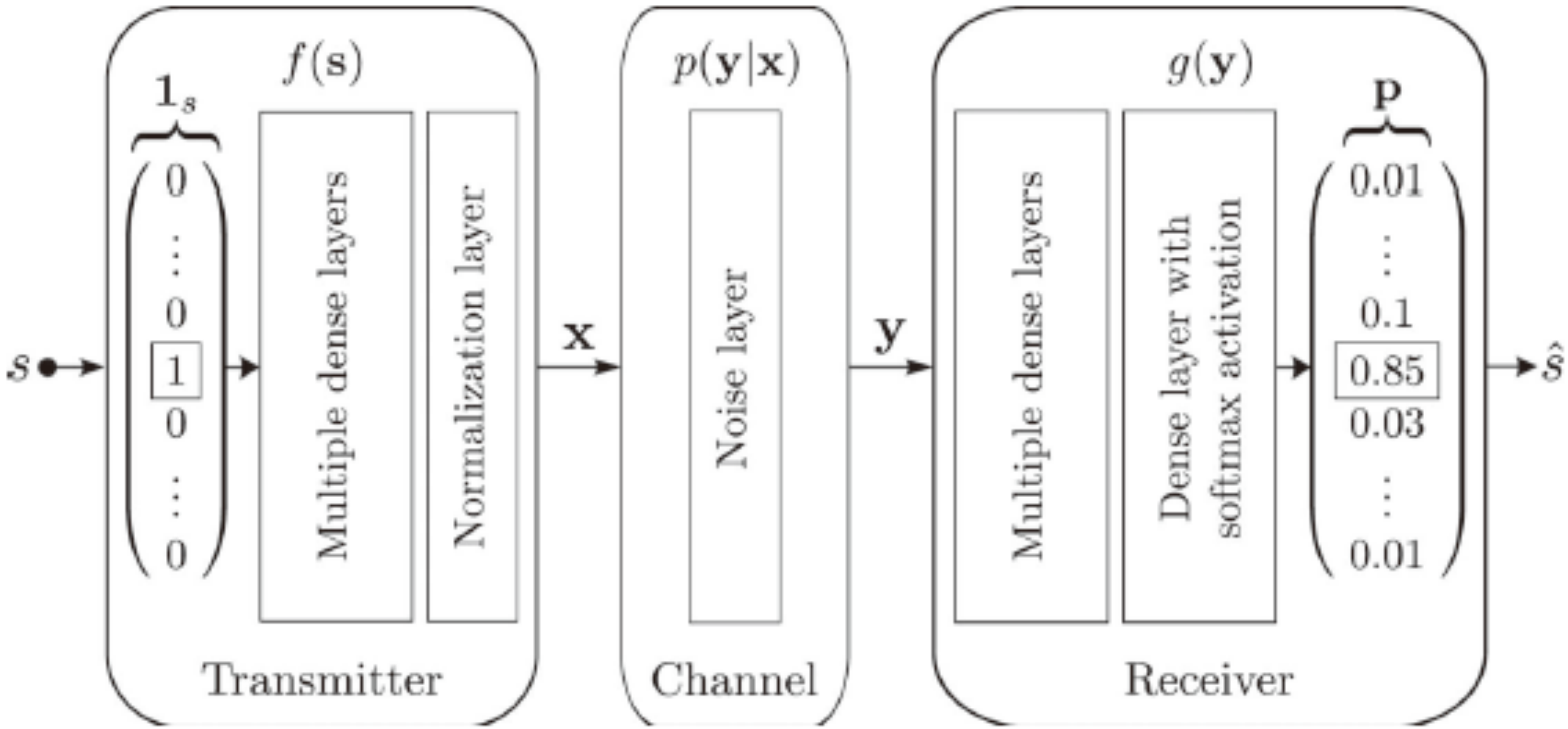
Find a useful representation $\mathbf{y} \in \mathbb{R}^r$ of $\mathbf{x} \in \mathbb{R}^n$ at intermediate layer through learning to reproduce the input at the output

- Incomplete autoencoder: $r < n$
- Overcomplete autoencoder: $r \geq n$

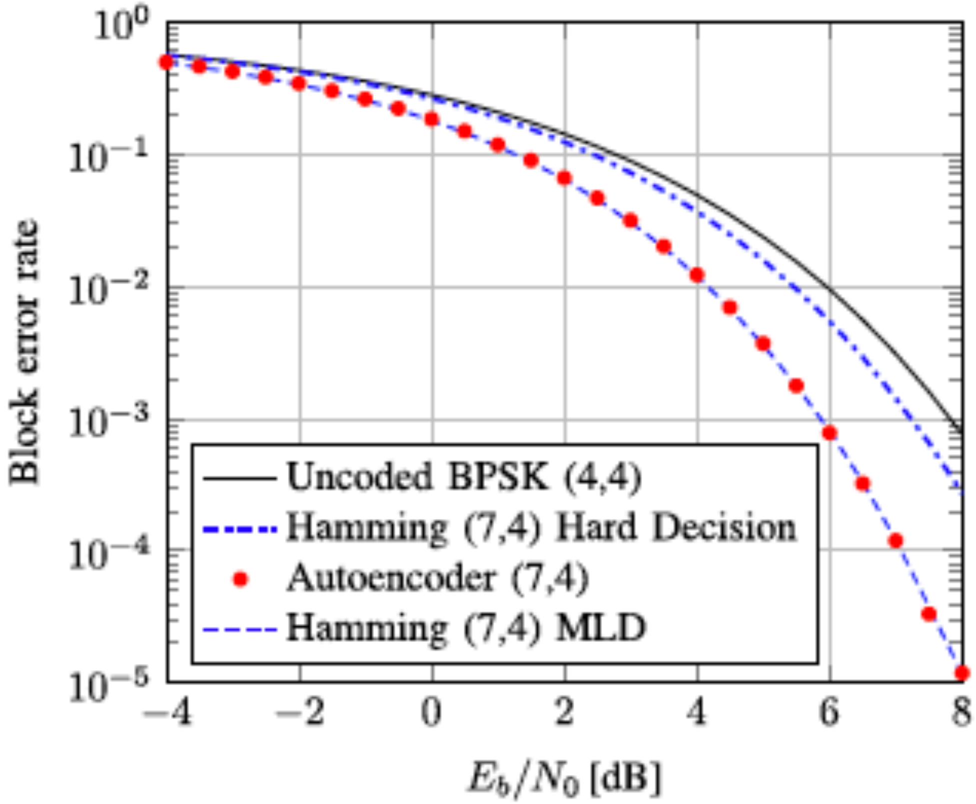
Modeling Communication System as Autoencoder



Modeling Communication System as Autoencoder



Modeling Communication System as Autoencoder



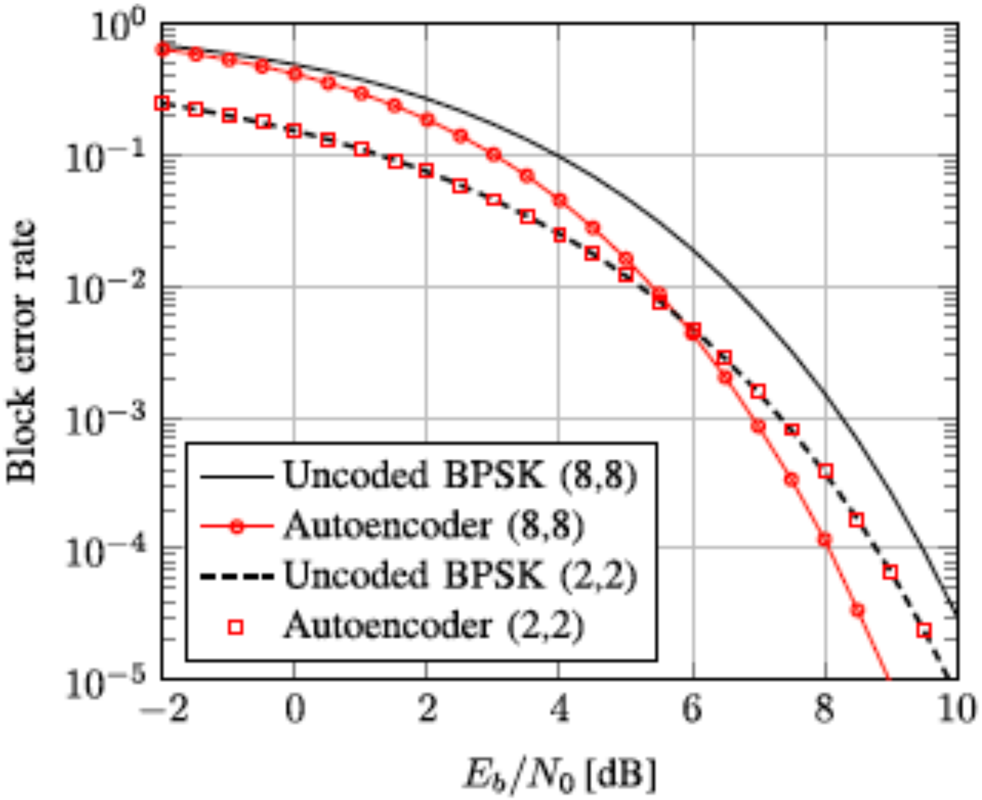
(a)

$R = 4/7$

- Autoencoder has learned without any prior knowledge an encoder and decoder function that together achieve the same performance as the Hamming (7,4) code with MLD.

T. O'Shea and J. Hoydis, IEEE Trans. on Cognitive Communications and Networking, 2017

Modeling Communication System as Autoencoder



(b)

$R = 1$

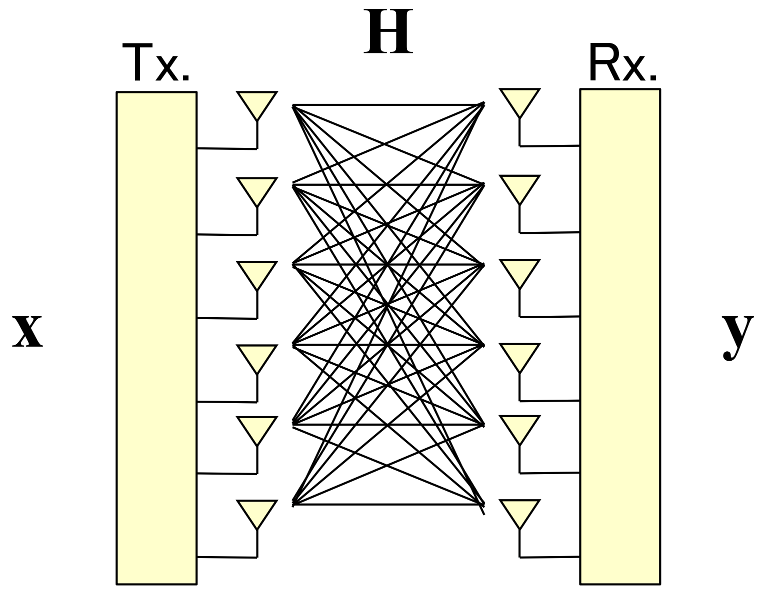
- While autoencoder achieves the same BLER as uncoded BPSK for (2,2), it outperforms the latter for (8,8) over the full range of Eb/N0.
  - This implies that it has learned some joint coding and modulation scheme, such that a coding gain is achieved.

T. O'Shea and J. Hoydis, IEEE Trans. on Cognitive Communications and Networking, 2017

MIMO

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}$$

- $\mathbf{x}$: Transmit signal
- $\mathbf{H}$: Channel matrix
- $\mathbf{w}$: Noise



MIMO Detection

- **Linear detectors**
  - Maximum ratio combining (MRC): $\mathbf{H}^T\mathbf{y}$
  - Zero forcing (ZF): $(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}$
  - Minimum mean squared error (MMSE): $(\mathbf{H}^T\mathbf{H} + \sigma^2\mathbf{I})^{-1}\mathbf{H}^T\mathbf{y}$

- **Non-linear detectors**
  - Maximum likelihood detection (MLD)
  - Sphere detection
  - Ordered successive detection (OSD)
  - Semidefinite relaxation (SDR)
  - Approximate message passing (AMP)
  - …

Deep MIMO Detection

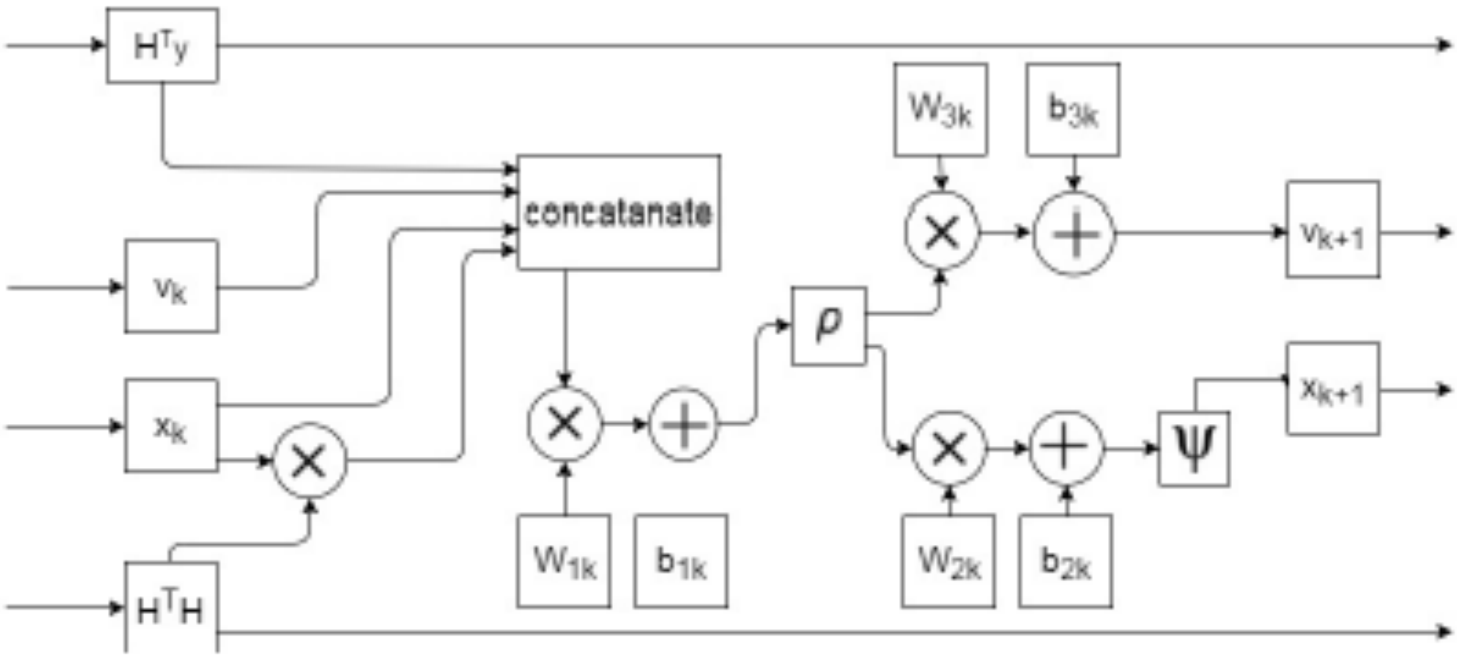
- N. Samuel, T. Diskin, and A. Wiesel, “Deep MIMO Detection,” arXiv:1706.01151, 2017
- Unfolding a projected gradient descent like solution for ML optimization

$$\begin{aligned}\hat{\mathbf{x}}_{k+1} &= \Pi\left[\hat{\mathbf{x}}_k - \delta_k \left.\frac{\partial\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2}{\partial\mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_k}\right] \\ &= \Pi\left[\hat{\mathbf{x}}_k - \delta_k\mathbf{H}^T\mathbf{y} + \delta_k\mathbf{H}^T\mathbf{H}\mathbf{x}_k\right],\end{aligned}$$

  - $\Pi[\ ]$: a nonlinear projection operator
  - δ_k : a step size

- Each iteration is a linear combination of $\hat{\mathbf{x}}_k$, $\mathbf{H}^{\mathrm{T}}\mathbf{y}$, $\mathbf{H}^{\mathrm{T}}\mathbf{H}\,\hat{\mathbf{x}}_k$ followed by a projection (non-linearity)
- Lift input into a higher-dimensional space, enrich iterations with trainable parameters, and apply standard DL non-linearities

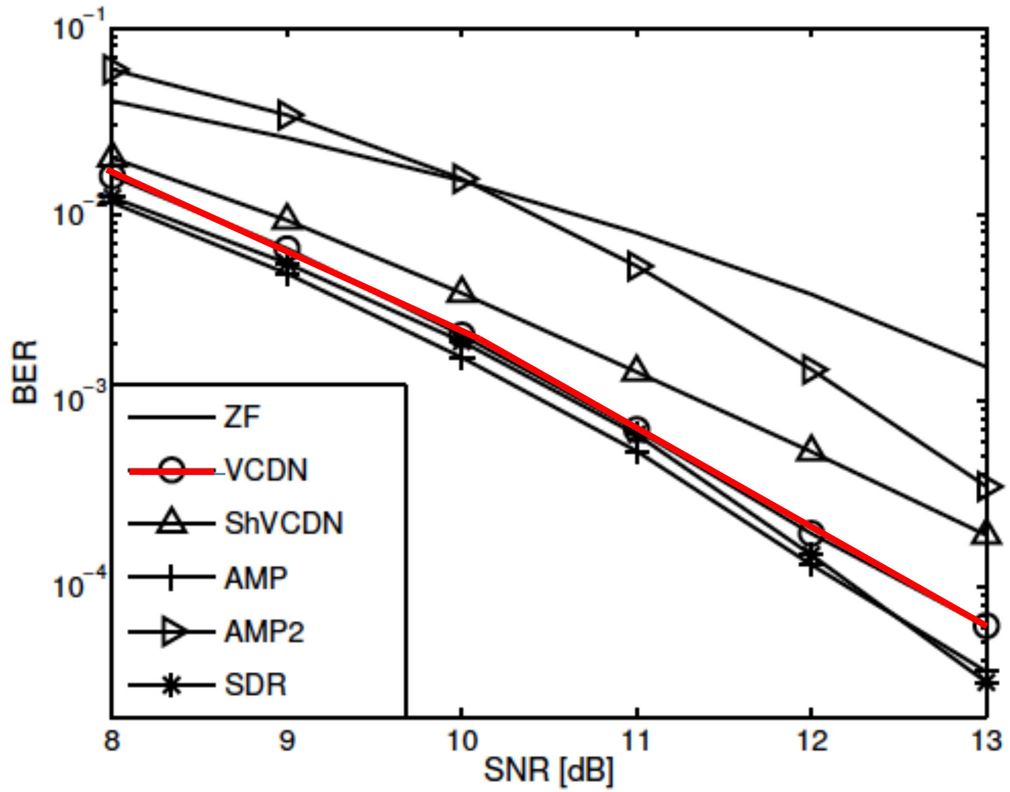
A Single Layer of Detection Network



- An arbitrary number of layers can be concatenated

N. Samuel, T. Diskin, and A. Wiesel, arXiv:1706.01151, 2017

BER of Detection Network

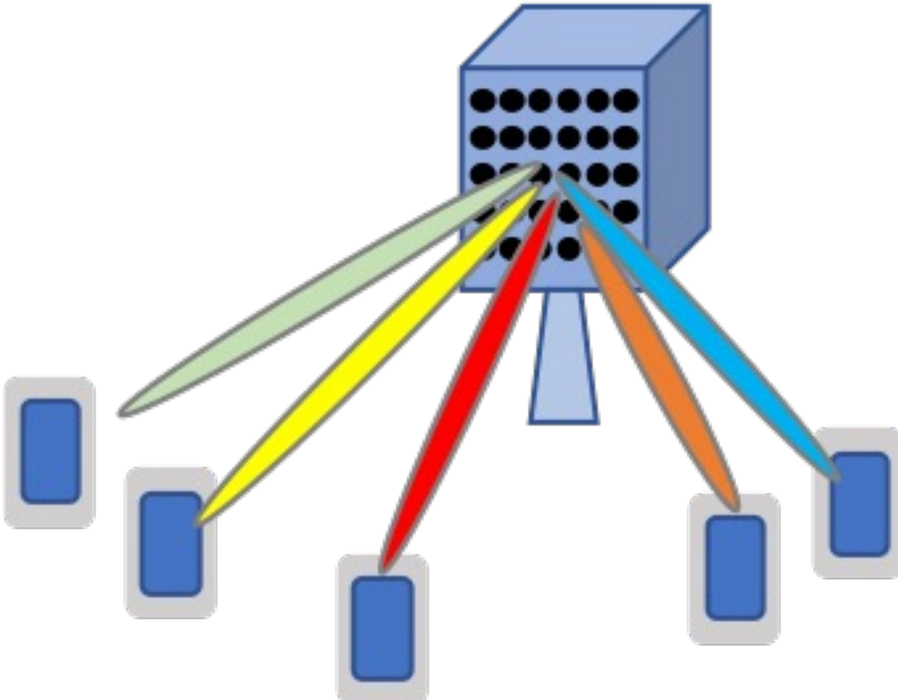


- 30 x 60 MIMO, random i.i.d. Gaussian channels
- Added a residual feature from ResNet
- Performance comparable to SDR/AMP but 30x faster

N. Samuel, T. Diskin, and A. Wiesel, arXiv:1706.01151, 2017

Massive MIMO

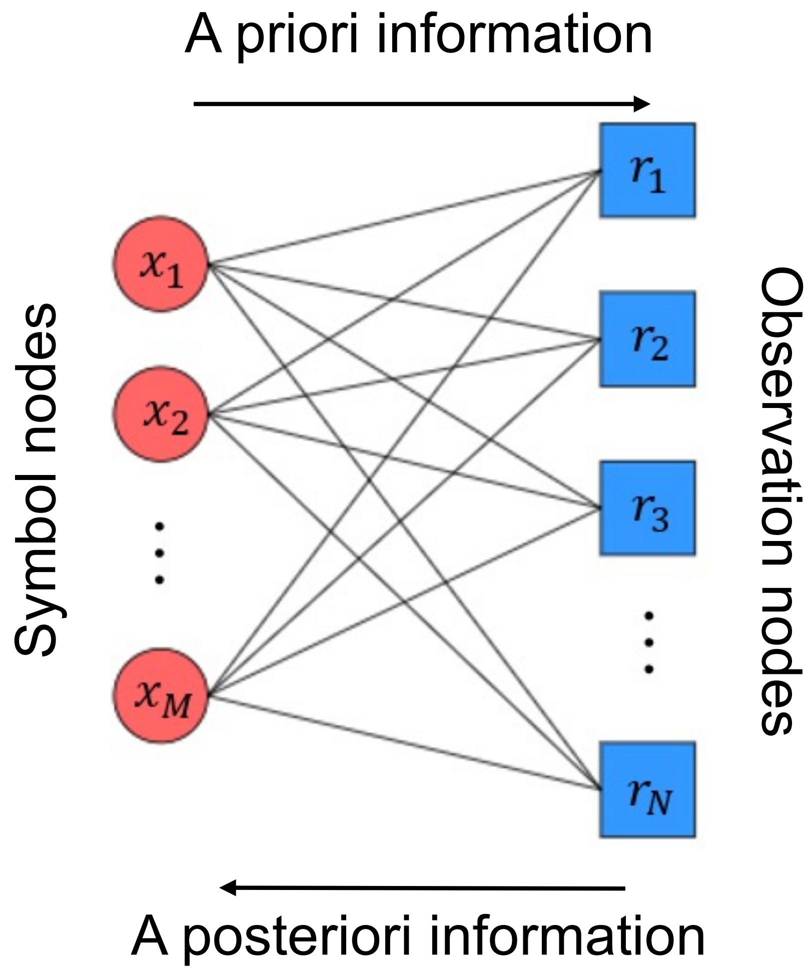
- A very large number of antenna arrays at BS
- Significantly increased spectral efficiency
- If the number of users increases, detection of a large number of streams is required.



Massive MIMO Detection

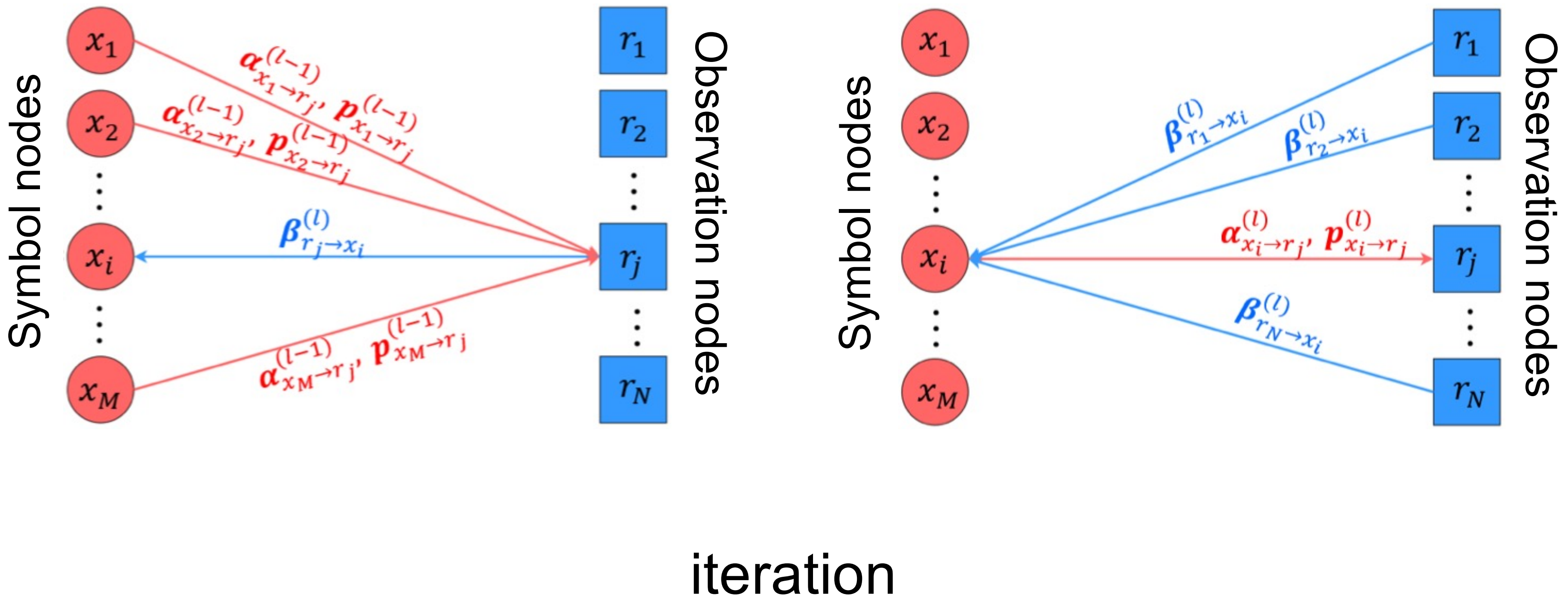
- MAP detection: $O(L^N)$
- Spatial filtering (MMSE, ZF): $O(N^3)$
- QR decomposition based algorithm: $O(N^3)$
- Detection based on belief propagation (BP): $O(N^2)$
  - N : # of antennas, L : symbol constellation size

MIMO BP Detection



A factor graph representation

MIMO BP Detection



MIMO BP Detection

Suboptimality of BP detection

- MIMO channel matrix has a loop ➔ not guaranteed to converge to the MAP
- Antenna correlation can aggravate the looping effect due to less randomness

Dumped BP detector

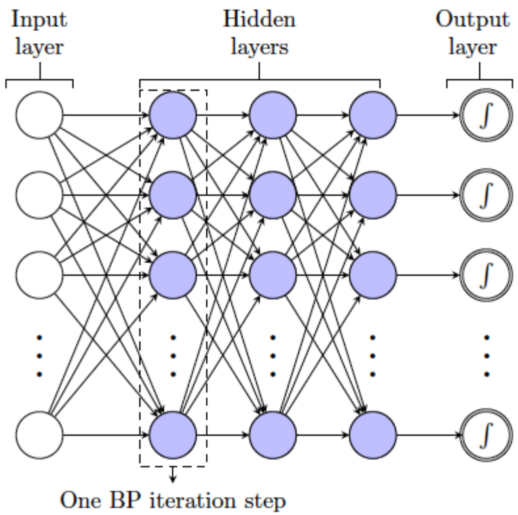
Messages at the l-th iteration

$$p_{i,j}^{(l)} \Leftarrow (1-\delta_{i,j}^{(l)})p_{i,j}^{(l)} + \delta_{i,j}^{(l)} p_{i,j}^{(l-1)}$$

$\delta \in [0,1]$: damping factor

Deep Unfolding

- J. R. Hershey, J. L. Roux, and F. Weninger, "Deep Unfolding: Model-Based Inspiration of Novel Deep Architectures," arXiv:1409.2574, 2014.
- Unfold the iterations into a layer-wise structure analogous to a neural network.
- De-couple the model parameters across layers to obtain novel neural-network-like architectures that can easily be trained discriminatively using gradient-based methods.



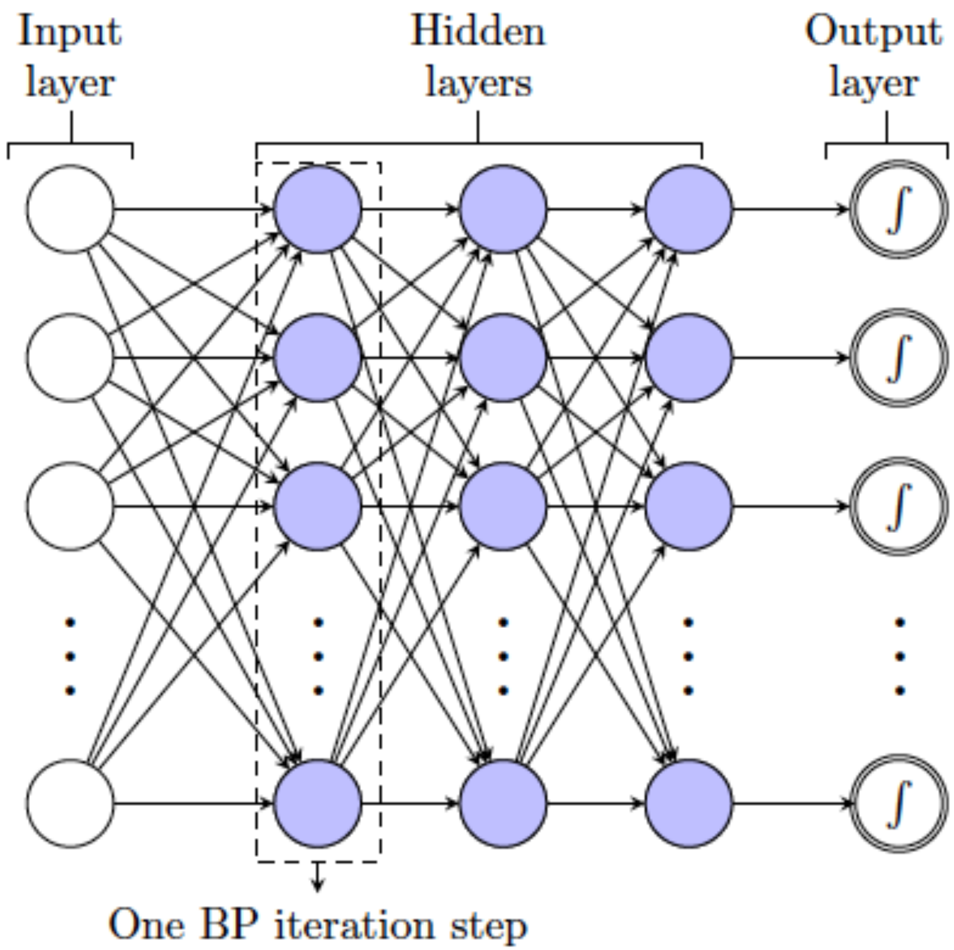
BP FG and DNN

Similarities

BP FG	DNN
Nodes	Neurons
Transmitted signals $\mathbf{x}$	Input data $\mathbf{x}$
Received signals $\mathbf{x}$	Output data $\mathbf{y}$
l-th iteration	l-th hidden layer
Belief messages $\alpha^{(l)}$, $\beta^{(l)}$, $p^{(l)}$,	Hidden signals $\mathbf{x}_l$
Message updating rules	Mapping function between layers
Correction factor δ	Parameters θ

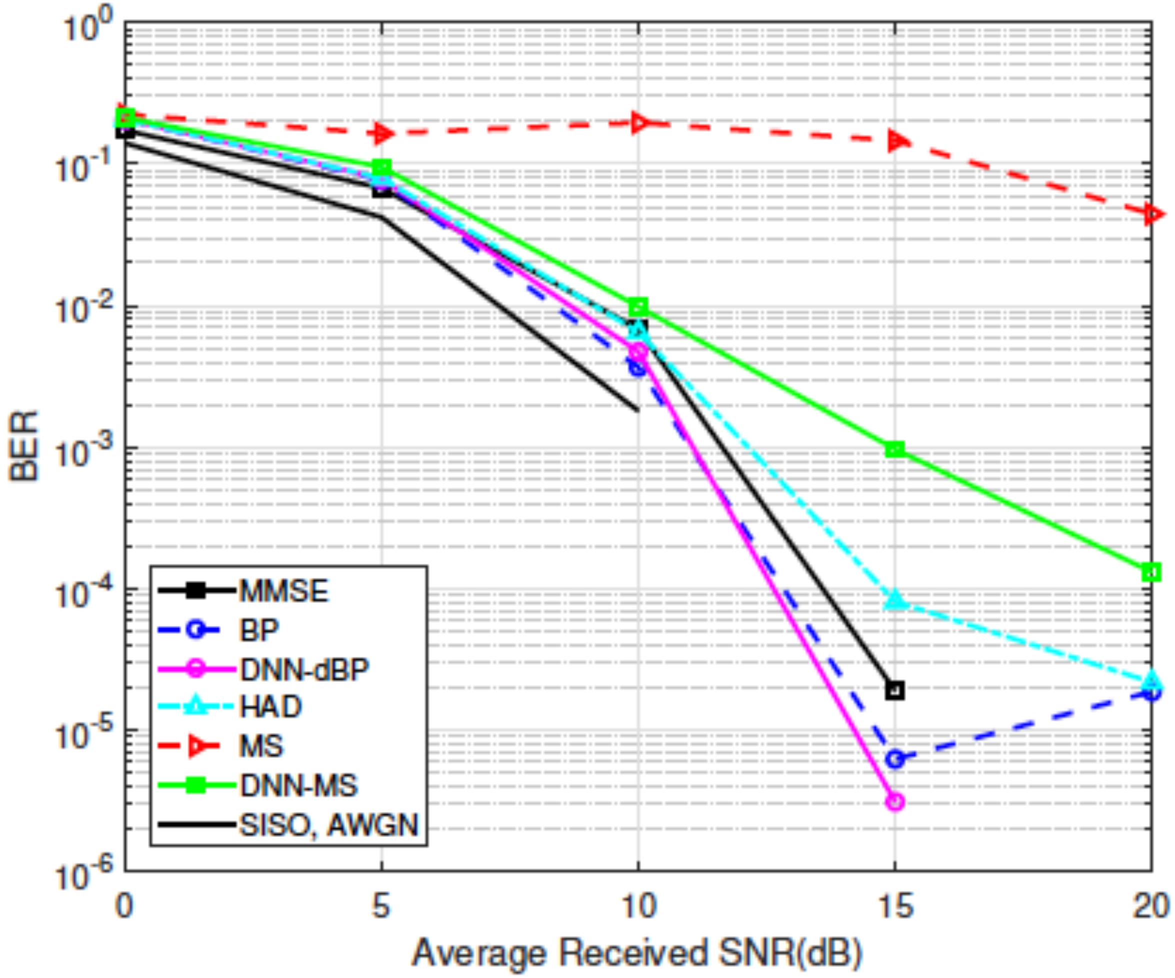
X. Tan et al., arXiv:1804.01002v2, 2018

DNN Detector



The structure of the DNN detector with 2 BP iterations

DNN Detector: BER



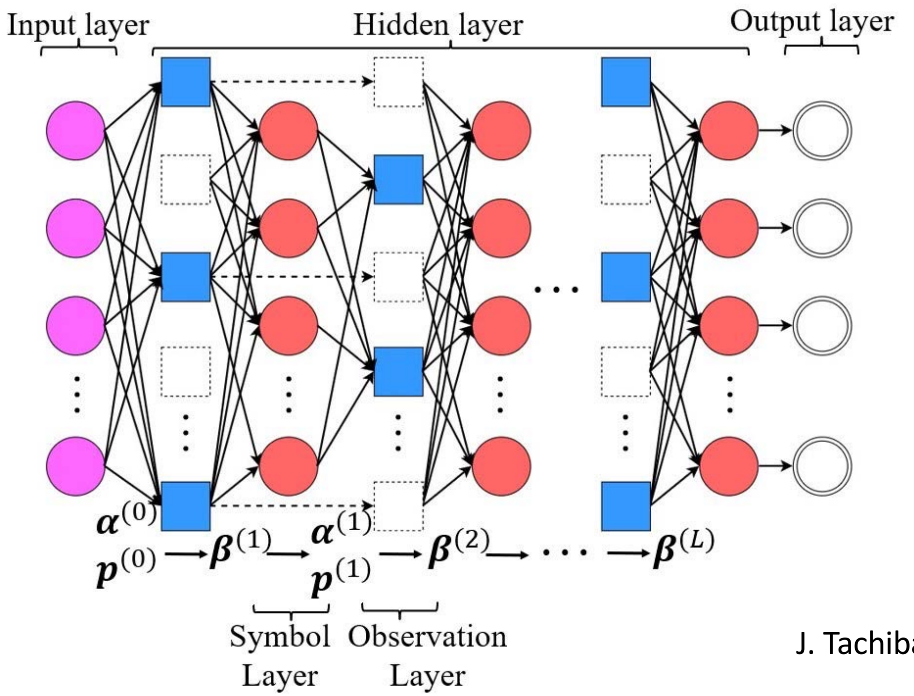
8 x 32 MIMO, i.i.d. Rayleigh channels, 16QAM,
7 hidden layers

X. Tan et al., arXiv:1804.01002v2, 2018

BP Detector with Node Selection and Dumping Factor Learning

- J. Tachibana and T. Ohtsuki, “Learning and Analysis of Damping Factor Learning in Massive MIMO Detection Using BP Algorithm With Node Selection,” IEEE Access 2020
- **Node Selection** : Updating not all but some observation nodes with low correlation in one iteration (one layer)

➔ Reduction of short loops

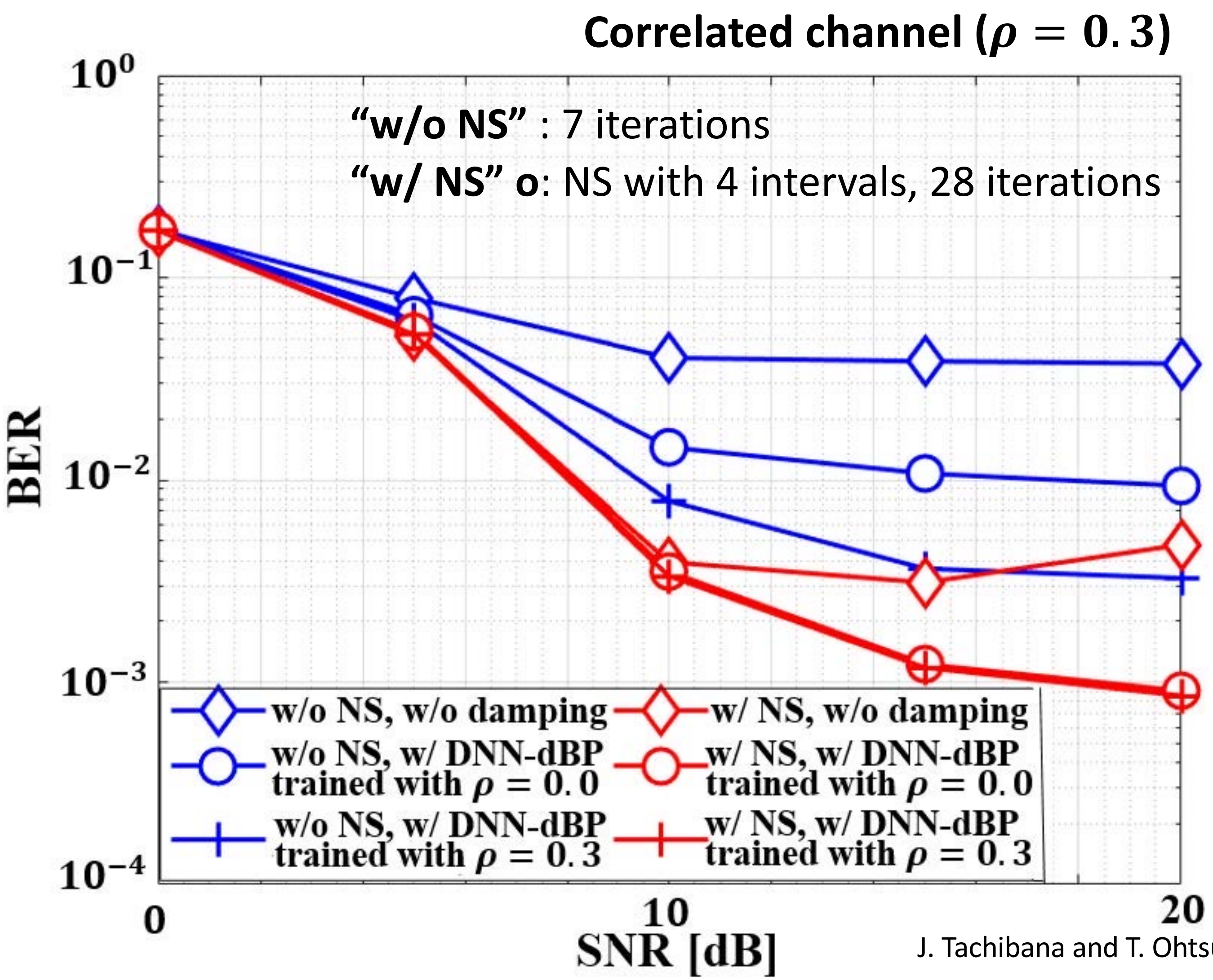


J. Tachibana and T. Ohtsuki, IEEE Access 2020

Keio University

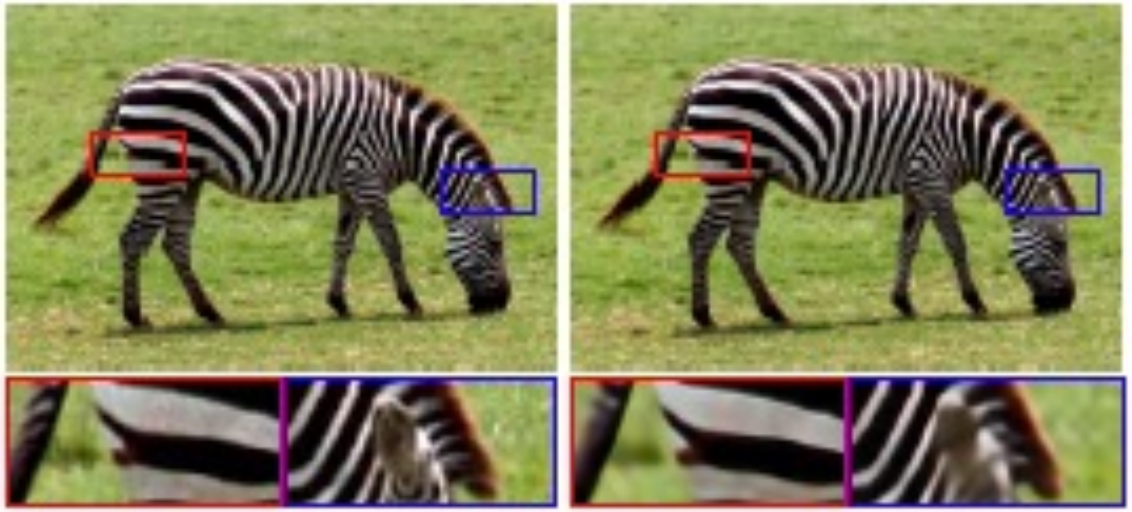
Tomoaki Ohtsuki

BP Detector with Node Selection and Dumping Factor Learning



J. Tachibana and T. Ohtsuki, IEEE Access 2020

Image Restoration



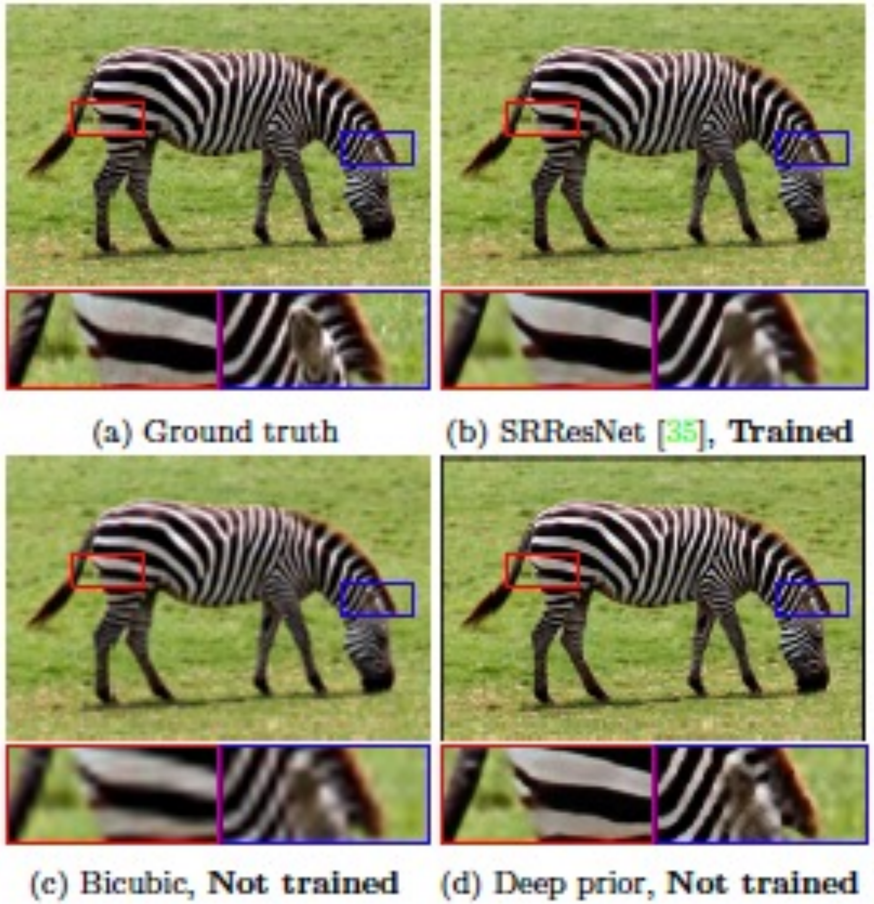
(a) Ground truth (b) SRResNet [35], Trained

Deep convolutional networks : a popular tool for image restoration and generation. Their excellent performance is due to their ability to learn realistic image priors from a large number of example images?

https://dmitryulyanov.github.io/deep_image_prior

Deep Image Prior

Structure of the ConvNets imposes a strong prior!



https://dmitryulyanov.github.io/deep_image_prior

Deep Image Prior

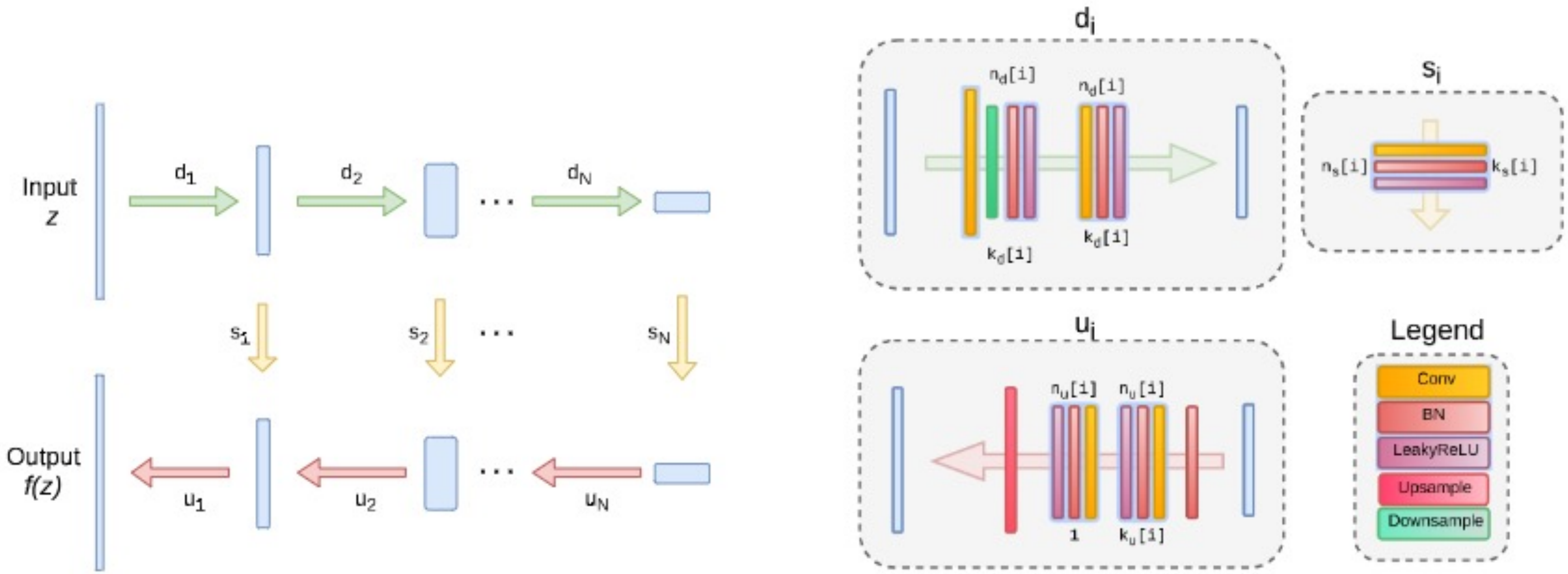
- Contrary to the belief that learning is necessary for building good image priors, a great deal of image statistics is captured by the structure of a convolutional image generator independent of learning.

https://dmitryulyanov.github.io/deep_image_prior

Keio University

Tomoaki Ohtsuki

Architecture used in the experiments



https://dmitryulyanov.github.io/deep_image_prior

Deep Image Prior

x : Clean image

x_0 : Corrupted/degraded image (observed)

z : Code vector

f_θ : Neural network with parameters θ

$E(x, x_0)$: Task-dependent data term

1. Initialize z
   - For example, with uniform noise $U(-1,1)$
2. Solve $\theta^* = \arg\min_\theta E(f_\theta(z); x_0)$
   - With any gradient-based method

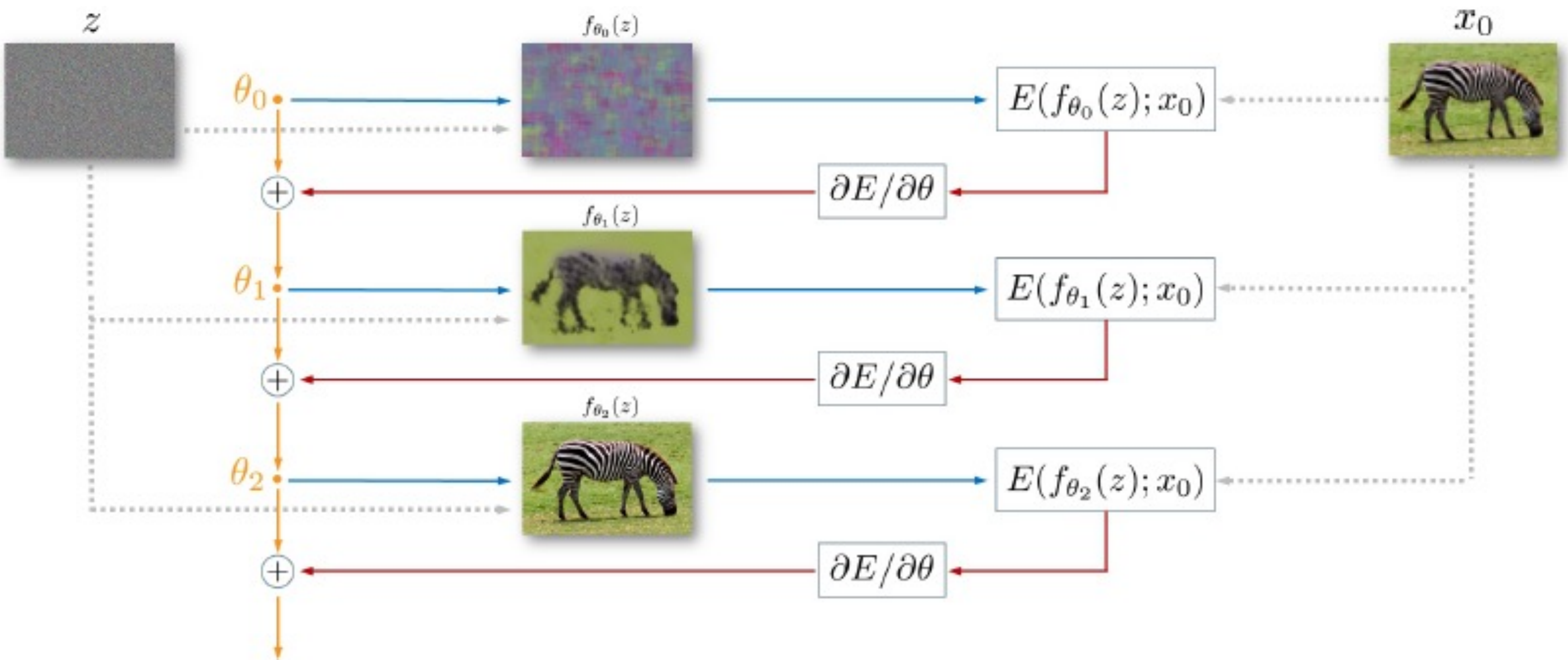
$$\theta^{k+1} = \theta^k - \alpha \frac{\partial E(f_\theta(z); x_0)}{\partial \theta};$$

3. Solution

$$x^* = f_{\theta^*}(z)$$

https://dmitryulyanov.github.io/deep_image_prior

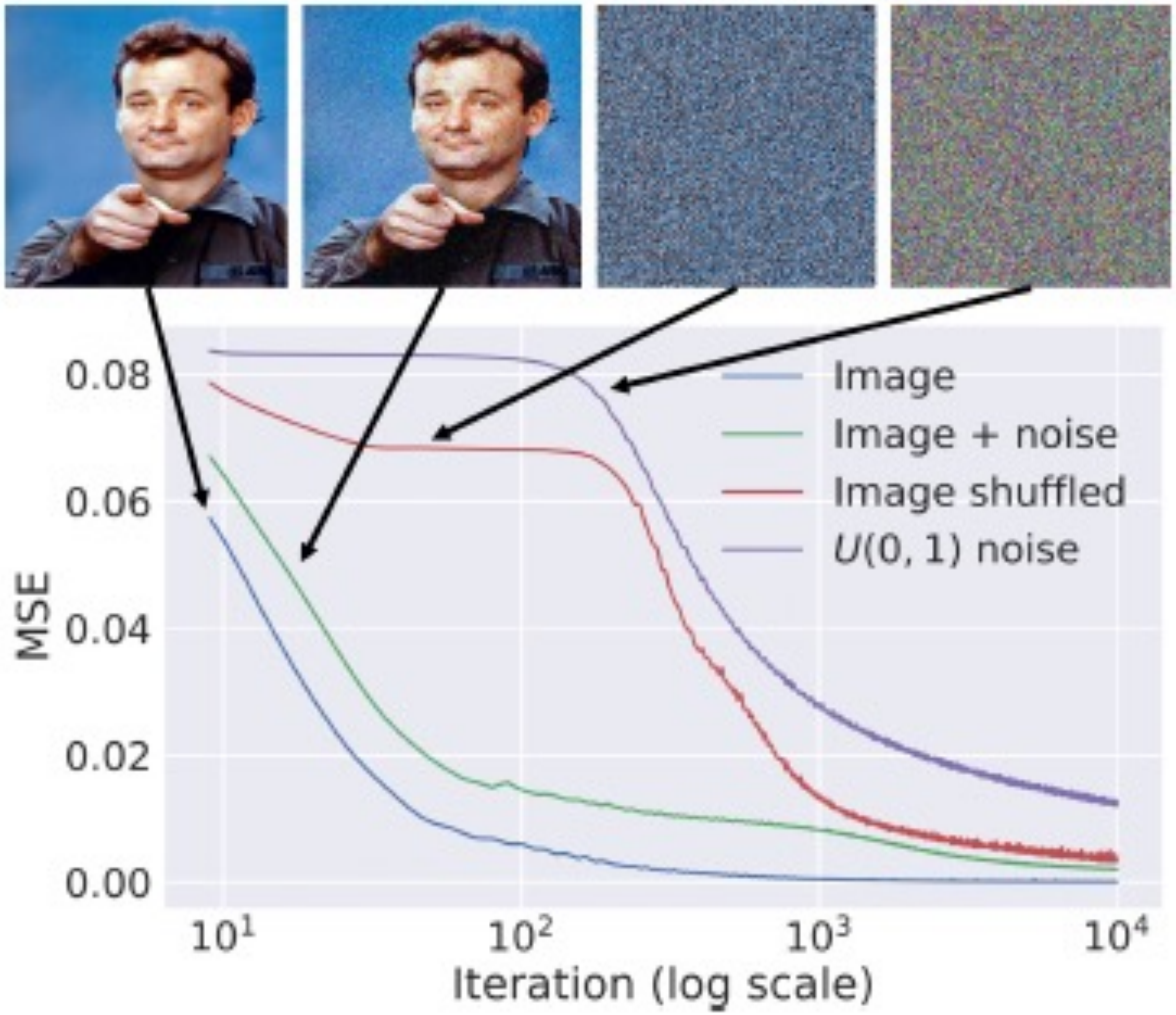
Image Restoration Using DIP



Starting from a random weights θ_0, we iteratively update them to minimize the data term $E(f_\theta(z); x_0)$. At every iteration, the weights are mapped to an image $x = f_\theta(z)$. The image x is used to compute the task-dependent loss $E(x, x_0)$. The gradient of the loss w.r.t. the weights θ is then computed and used to update the parameters.

https://dmitryulyanov.github.io/deep_image_prior

DIP : Learning Curves



The network has high impedance to noise and low impedance to signal.

- For most applications, the number of iterations in the optimization process is restricted to a certain number.

https://dmitryulyanov.github.io/deep_image_prior

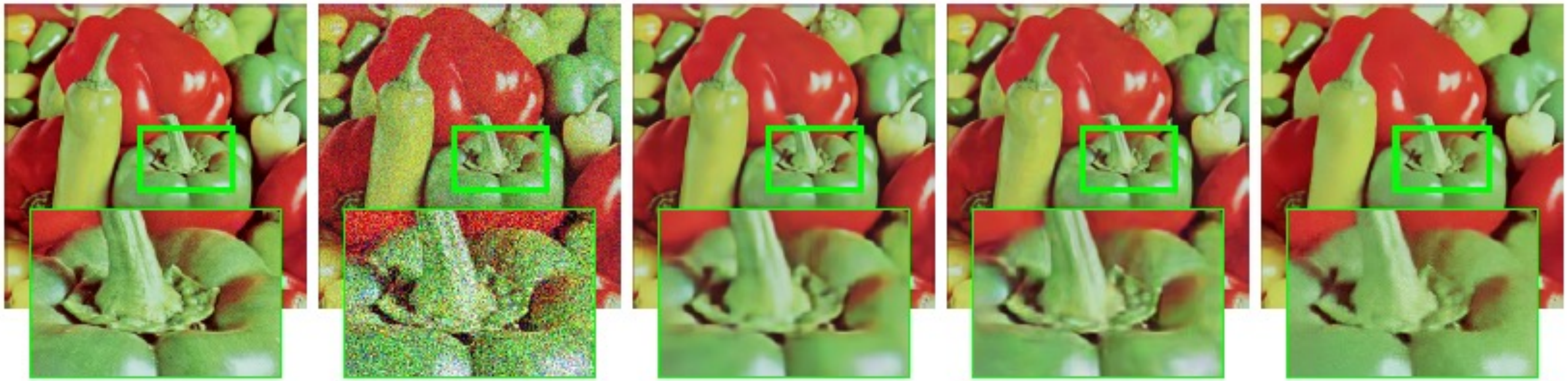
Deep Image Prior : Data Term

x : Clean image
x_0 : Corrupted/degraded image (observed)
m : Binary mask

Objective: $\theta^* = \arg\min_{\theta} E(f_\theta(z); x_0)$

- **Denoising**: $E(x, x_0) = \|x - x_0\|^2$ Needs early stopping
- **Inpainting**: $E(x, x_0) = \|(x - x_0) \odot m\|^2$, where $\odot$ is Hadamard's product, m is binary mask
- **Super-resolution**: $E(x, x_0) = \|d(x) - x_0\|^2$, where $d(\cdot)$ is a downsampling operator to resize the image
- **Feature-inversion**: $E(x, x_0) = \|\phi(x) - \phi(x_0)\|^2$, where ϕ is the first several layers of a neural network trained to perform

Denoising



(a) GT (b) Input (c) Ours (d) CBM3D (e) NLM

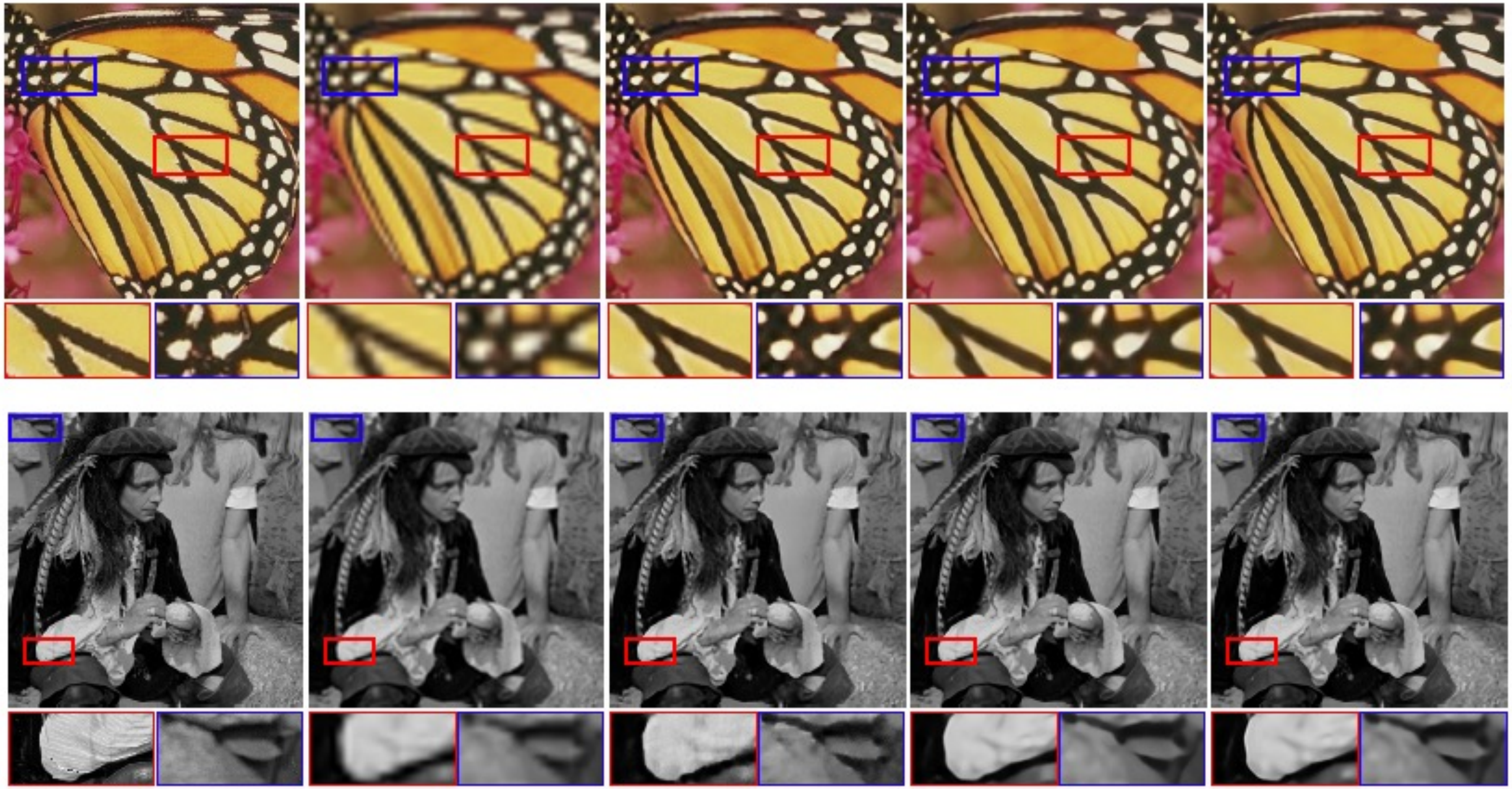
https://dmitryulyanov.github.io/deep_image_prior

Inpainting : Text

(a) Original image (b) Corrupted image (c) Shepard networks [44] (d) Deep Image Prior

https://dmitryulyanov.github.io/deep_image_prior

Super Resolution



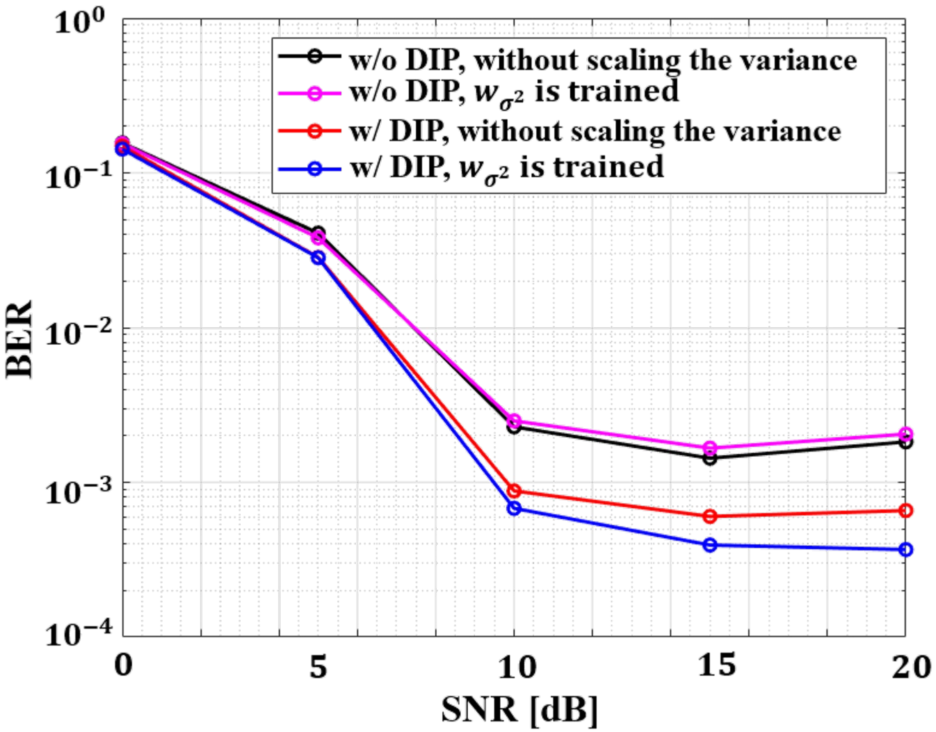
(a) Original image (b) Bicubic, **Not trained** (c) Ours, **Not trained** (d) LapSRN, **Trained** (e) SRResNet, **Trained**

https://dmitryulyanov.github.io/deep_image_prior

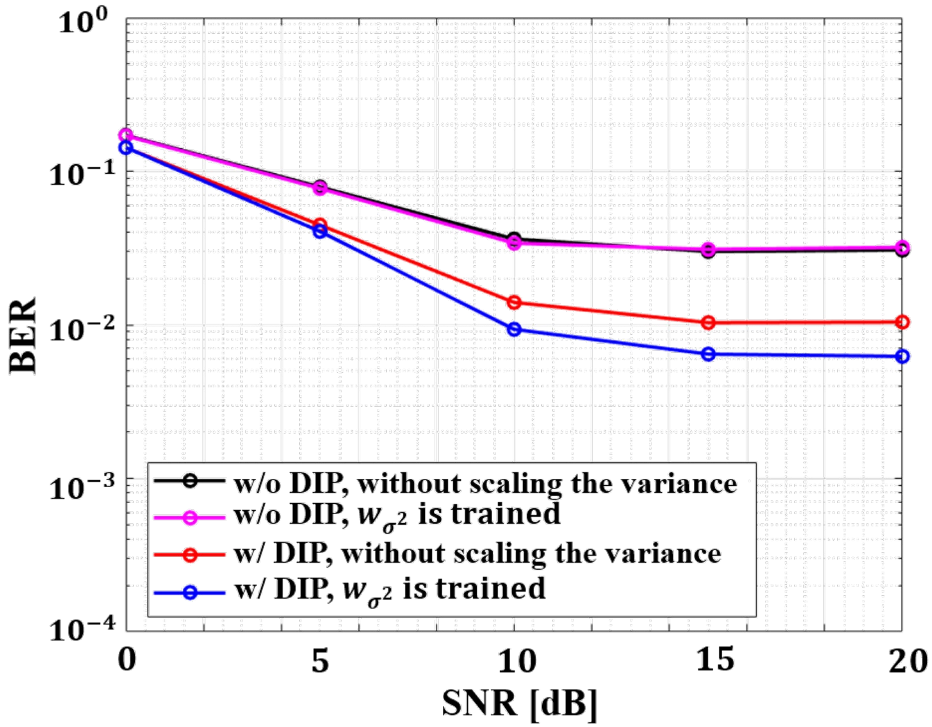
Massive MIMO BP Detection Using DIP with DNN-Trained Scaling Factor

- Tachibana and Ohtsuki, IEICE RCS, Mar. 5, 2021
- T. Ohtsuki, "Machine Learning in 6G Wireless Communications," IEICE Trans. Commun., 2022.

- Uncorrelated Ch. ($\rho = 0.0$)



- Correlated Ch. ($\rho = 0.3$)

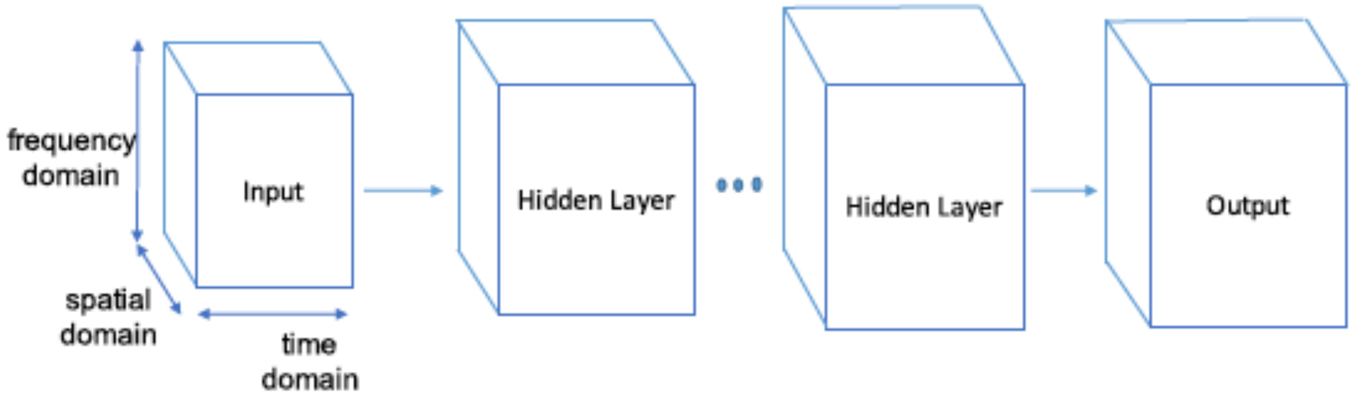


16 x 16 MIMO

DIP in Wireless Communications

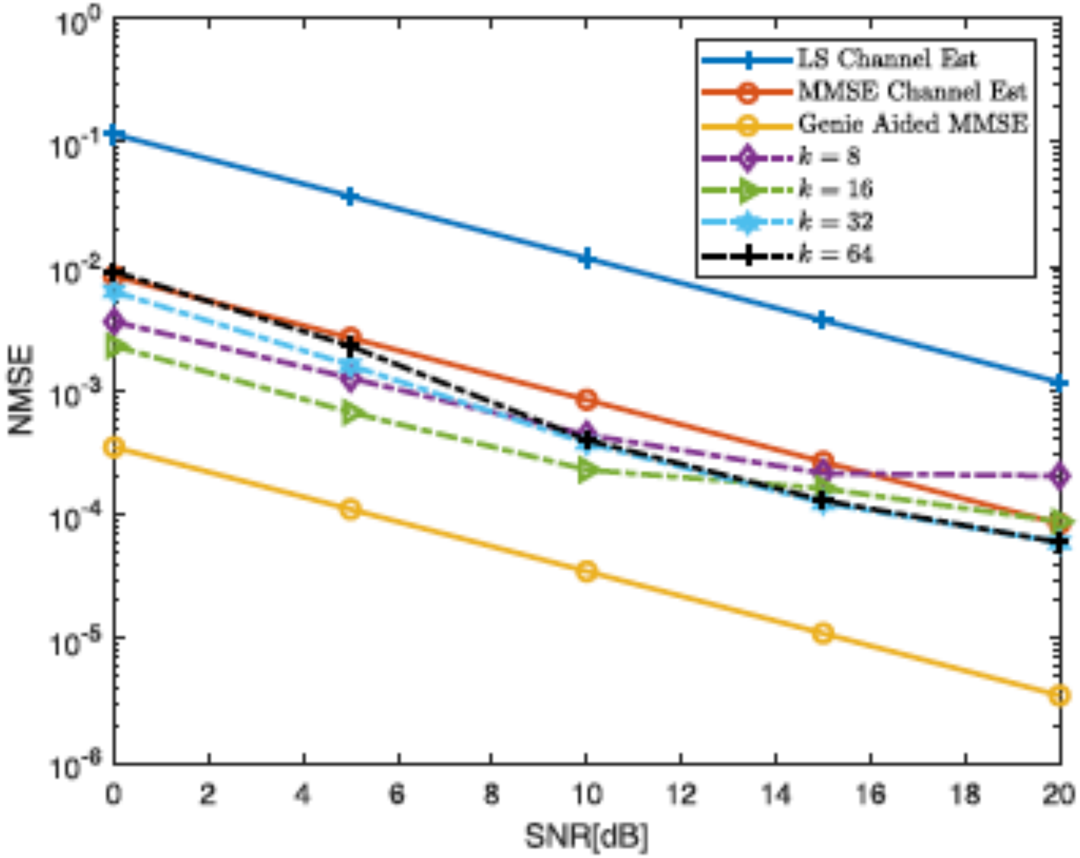
- E. Valebi et al., “Massive MIMO Channel Estimation With an Untrained Deep Neural Network,” IEEE TW, Mar. 2020

- Denoising the received signal via DIP followed by LS channel estimation



The DNN architecture to denoise and inpaint the received signal before channel estimation for a 3-dimensional communication signal.

DIP in Wireless Communications



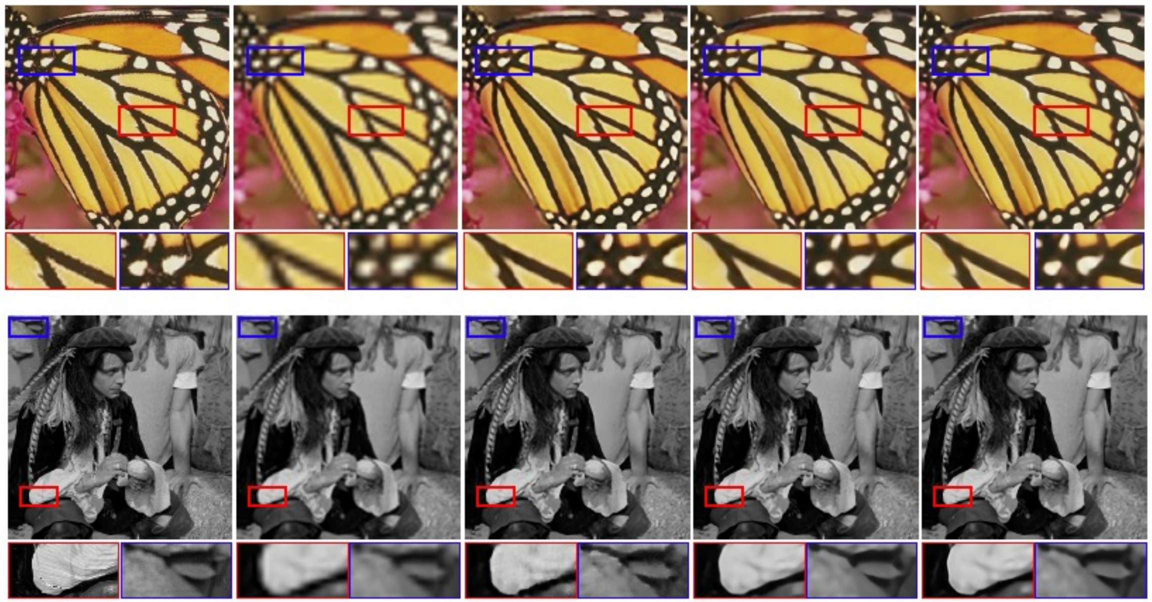
Kronecker channel model with $\rho = 0.5$

NMSE for different spatial dimension of the hidden layers k and $M = 64$ Tx antennas.

E. Valebi et al., IEEE TW, Mar. 2020

Super Resolution

- Create a high-resolution image from one or more low resolution images



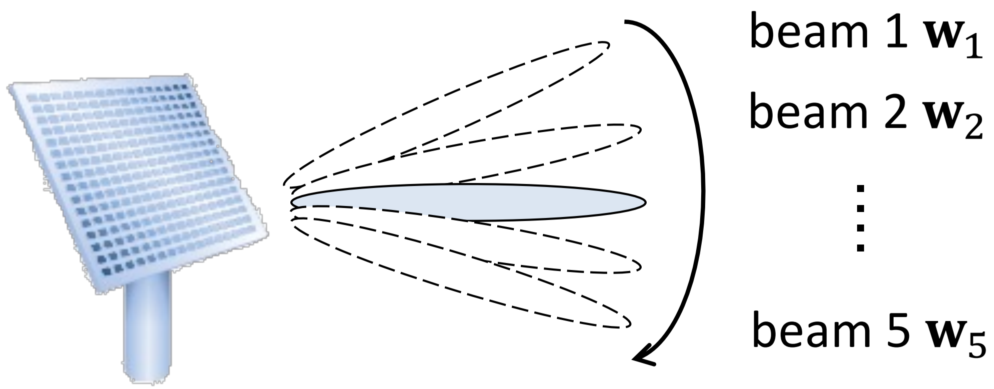
(a) Original image (b) Bicubic, **Not trained** (c) Ours, **Not trained** (d) LapSRN, **Trained** (e) SRResNet, **Trained**

https://dmitryulyanov.github.io/deep_image_prior

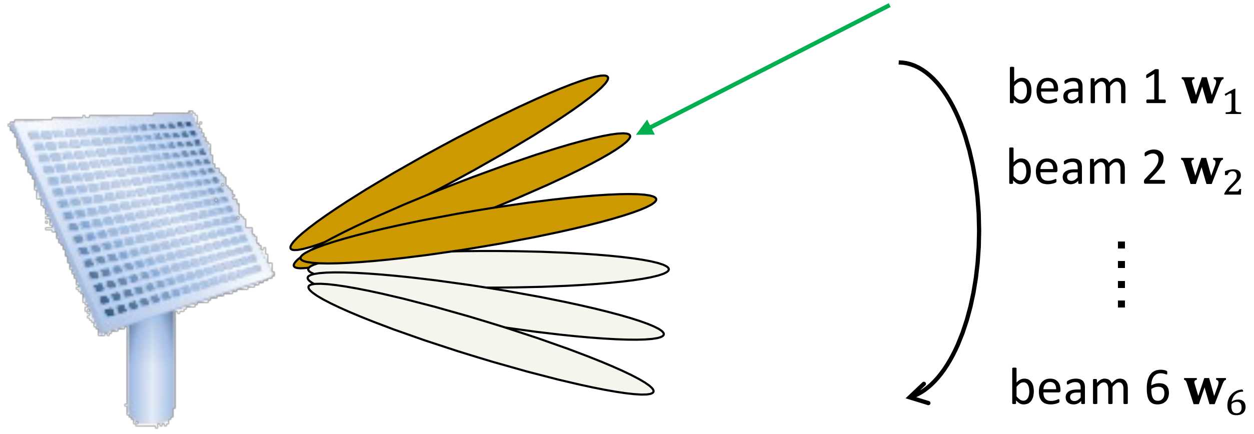
Super Resolution in Wireless Communications

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021

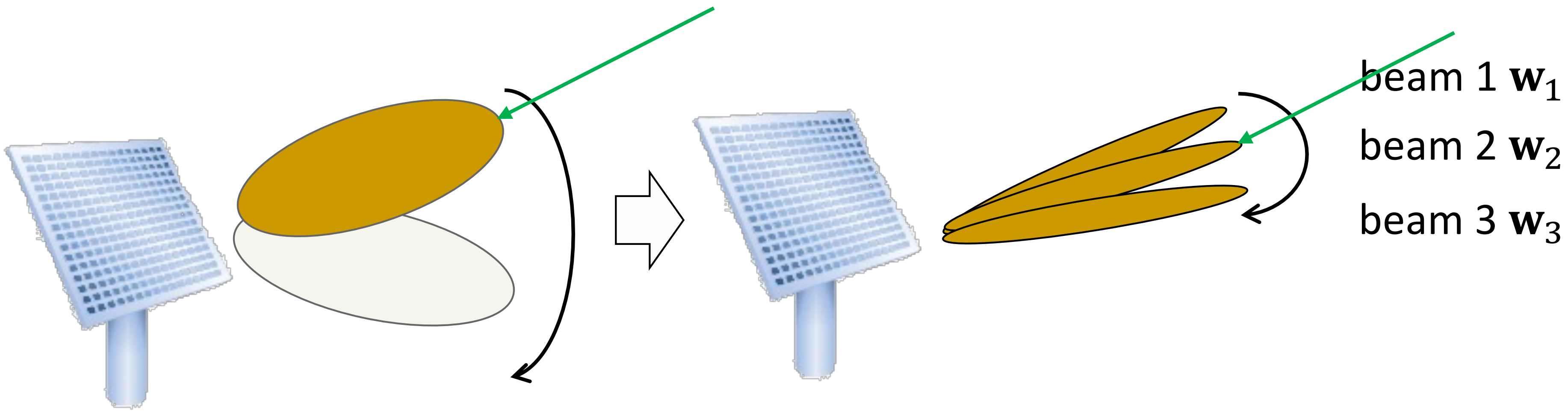
- Super resolution technique is applied to reduce overhead of beam search



Conventional Beam Searching Algorithm



Exhaustive search



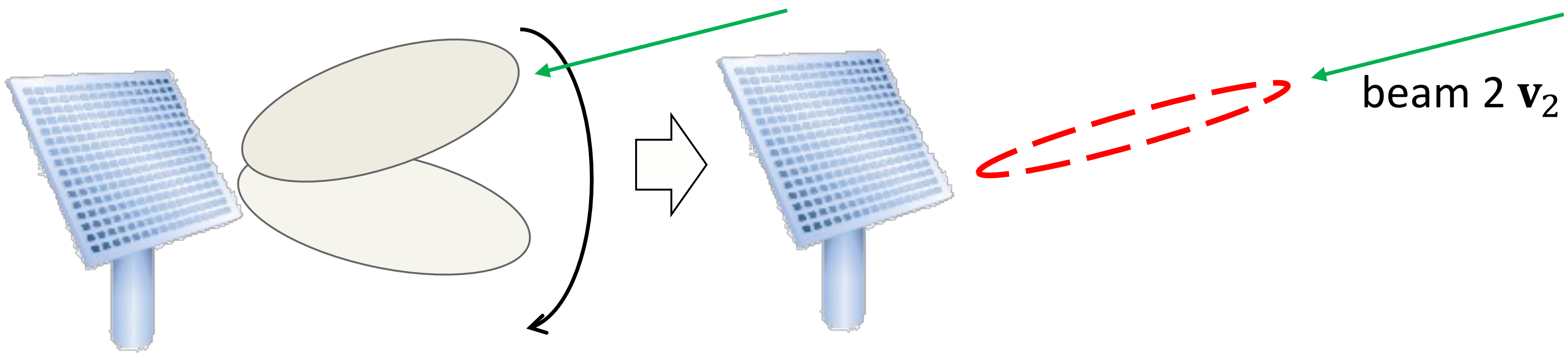
Hierarchical search

Deep Learning-based Beam Searching Algorithm

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021

- Our proposed model estimates the beam qualities with narrow beams based on beam measurements with wide beams.

➔ Increase the resolution of images : super resolution

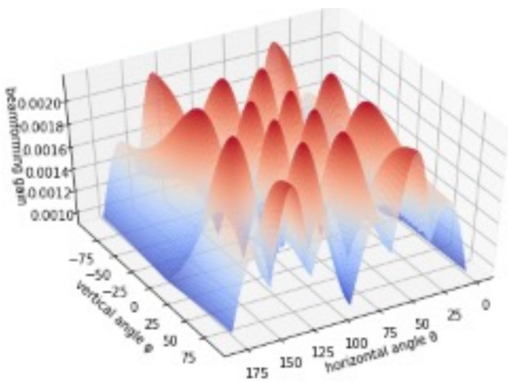


Deep learning-based beam search

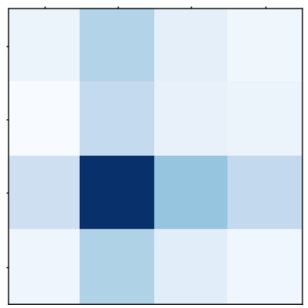
Deep Learning-based Beam Searching Algorithm

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021

4 × 4 DFT beam

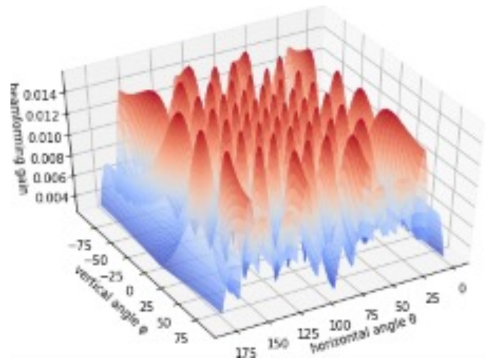


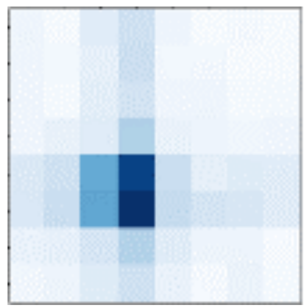
Low-resolution beam domain image



prediction

8 × 8 DFT beam



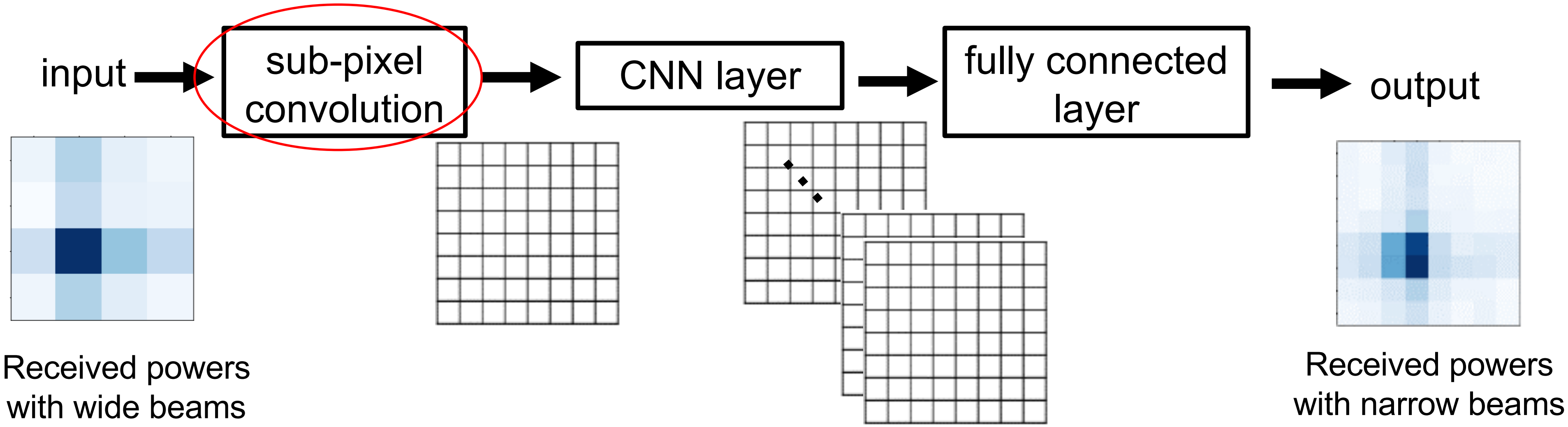


Tomoaki Ohtsuki

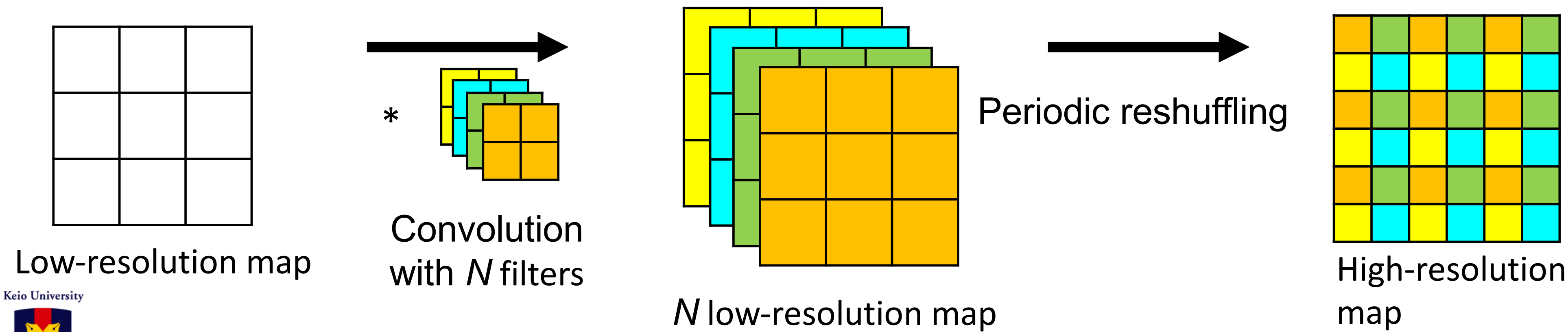
Deep Learning-based Beam Searching Algorithm

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021

Deep Learning-based Beam Searching Algorithm : Beam Quality Estimation

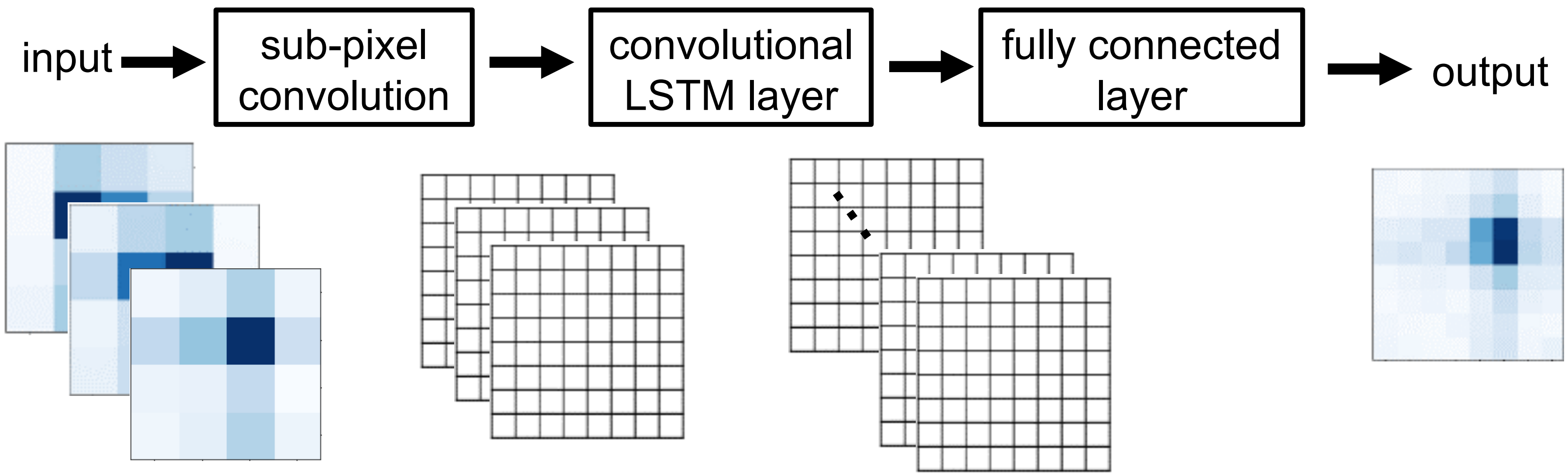


- Sub-pixel convolution (J. Caballero, et.al, *IEEE Comput, 2017*)



Deep Learning-based Beam Searching Algorithm : Beam Quality Prediction

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021



Past Received powers with wide beams

Future Received powers with narrow beams

Deep Learning-based Beam Searching Algorithm

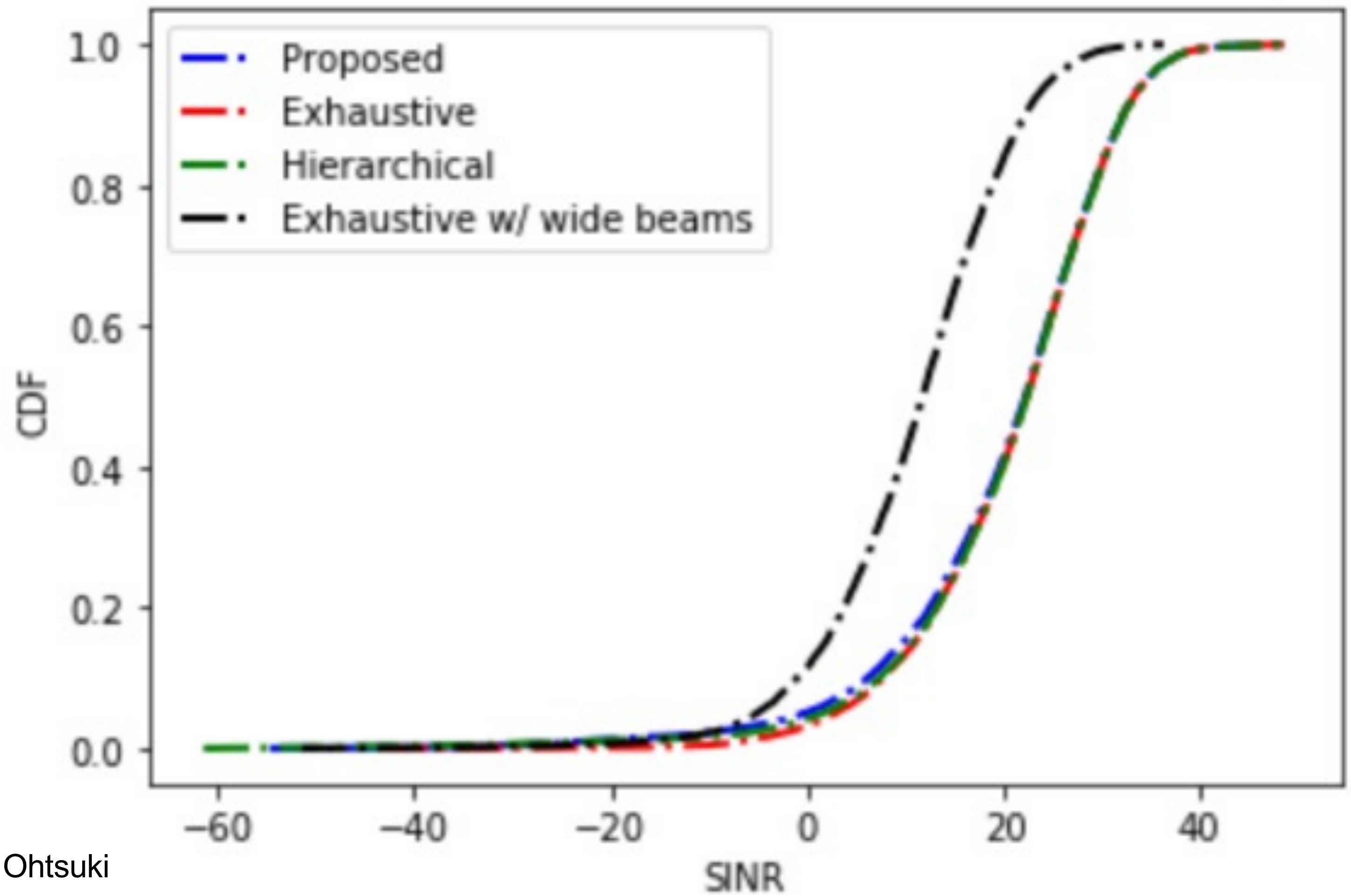
- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, "A Deep Learning-based Low Overhead Beam Selection in mmWave Communications," IEEE Trans. on Vehicular Technology, Jan. 2021

TABLE : NUMBER OF BEAM MEASUREMENTS PER COHERENCE TIME

Algorithm	Overhead
Proposed scheme	8
Exhaustive search (optimal beam select)	64
Hierarchical search	20
Exhaustive search with wide beams	16
Exhaustive search without prediction	32

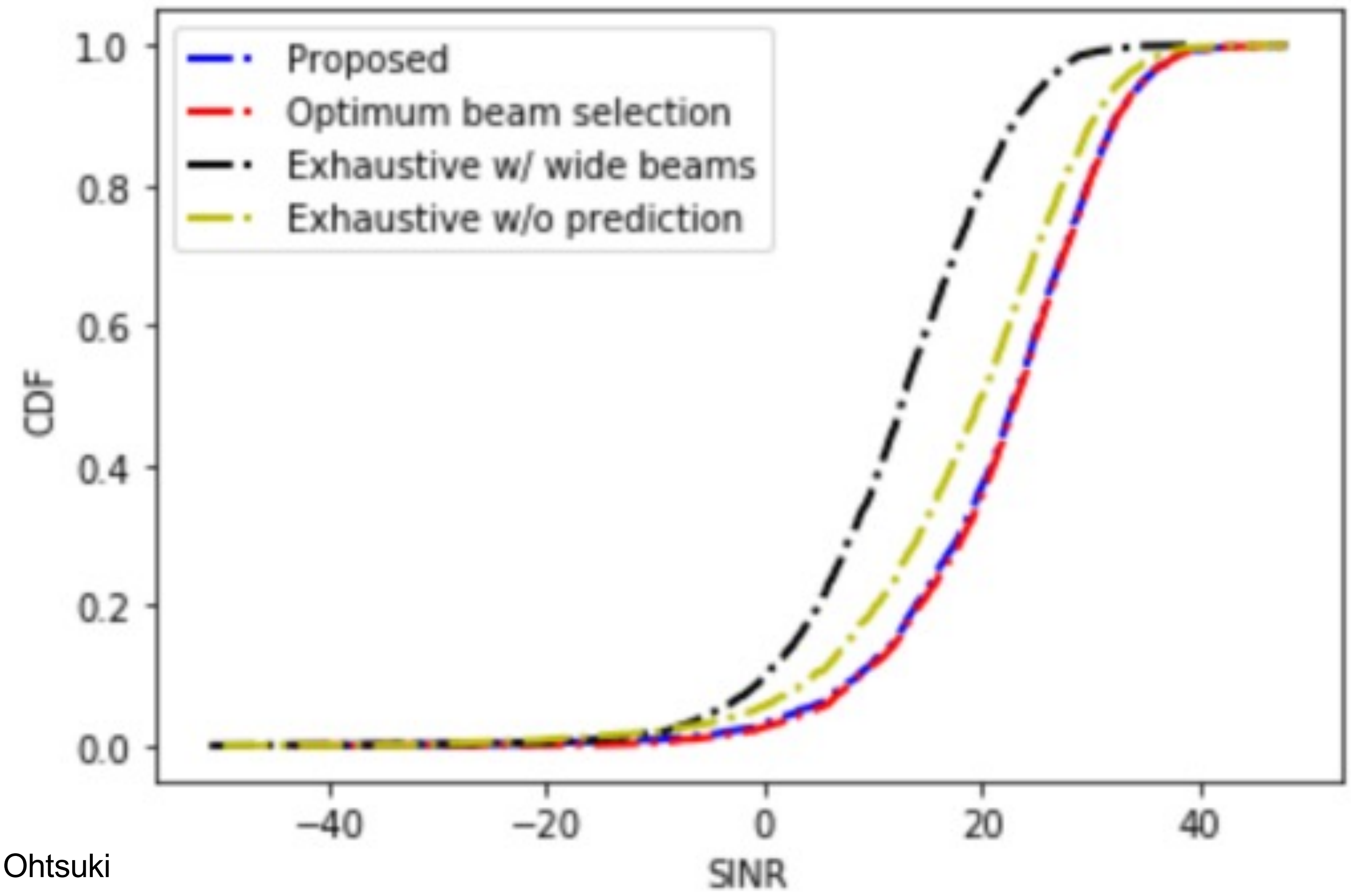
Deep Learning-based Beam Searching Algorithm : Beam Quality Estimation

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021



Deep Learning-based Beam Searching Algorithm : Beam Quality Prediction

- H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-based Low Overhead Beam Selection in mmWave Communications,” IEEE Trans. on Vehicular Technology, Jan. 2021

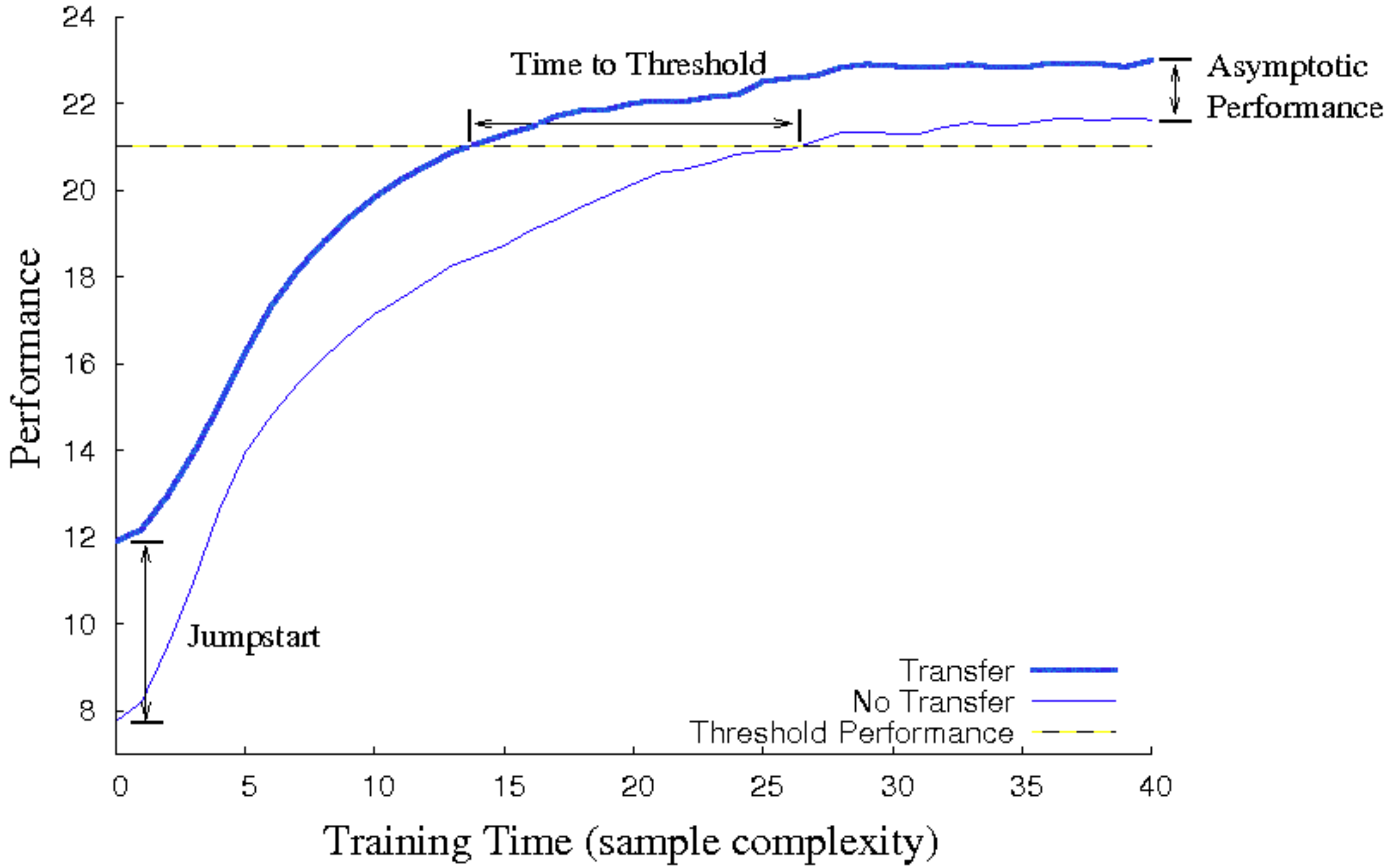


Transfer Learning : Motivation

- Traditional ML tasks assume the training/test data are drawn from the same data space and the same distribution
- Insufficient labelled data result in poor prediction performance
- Start from scratch is always time-consuming
- Transfer knowledge from other sources may help!

P.-H. (Eddy) Su and Y. Li, 2015

Transfer Learning : Motivation



Taylor et.al JMLR '09

Transfer Learning : Definition

- Ability of a system to recognize and apply knowledge and skills learned in previous domains/tasks to novel domains/tasks
- Improvement of learning in a **new** task through the *transfer of knowledge* from a **related** task that has already been learned.

Transfer Learning : Definition

Notations:

- Domain $\mathcal{D}$
  - Data space $\mathcal{X}$
  - Marginal distribution $P(X) \in \mathcal{X}$, where $X \in \mathcal{X}$

- Task $\mathcal{T}$ (Given $\mathcal{D} = \{X, P(X)\}$)
  - Label space $\mathcal{Y}$
  - Learn an $f: X \rightarrow Y$ to approach the underlying $P(Y|X)$, where $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$

P.-H. (Eddy) Su and Y. Li, 2015

Transfer Learning : Definition

Assume we have only one source S and one target T :

- Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where

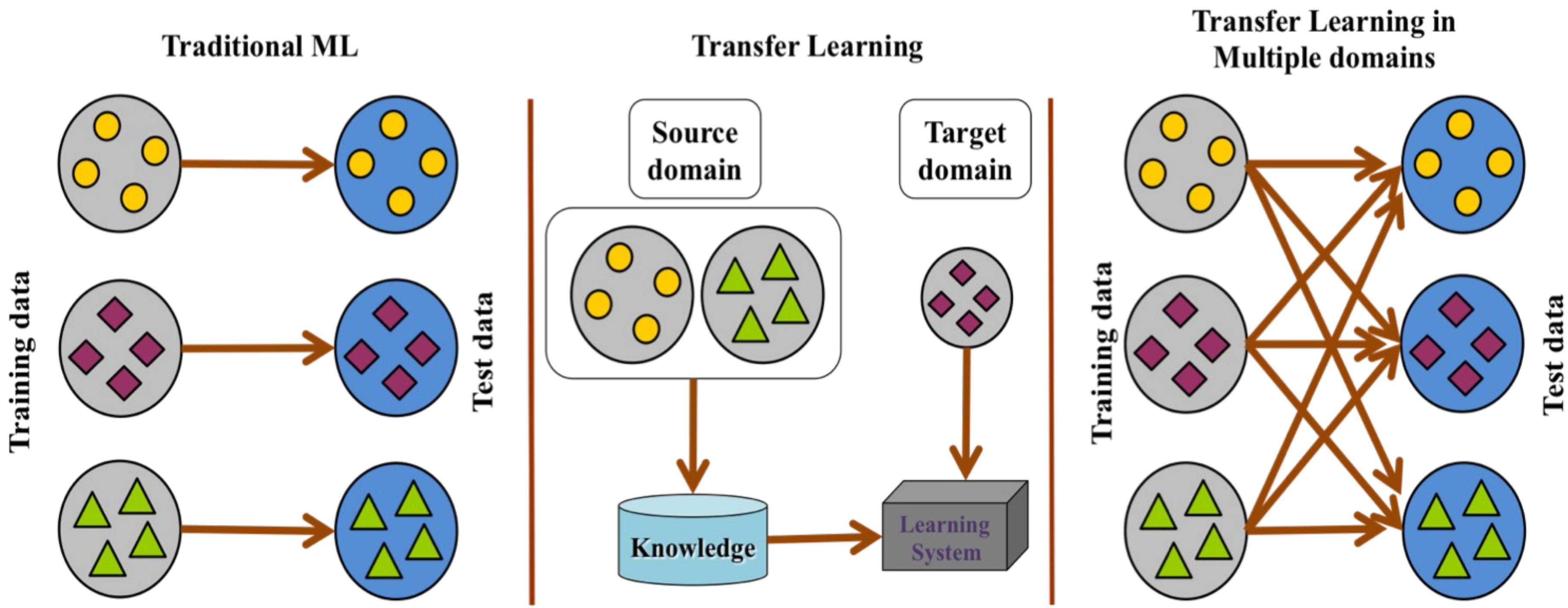
$$\mathcal{D}_S \neq \mathcal{D}_T \text{ (either } \mathcal{X}_S \neq \mathcal{X}_T \text{ or } P_S(X) \neq PT(X))$$

or

$$\mathcal{T}_S \neq \mathcal{T}_T \text{ (either } \mathcal{Y}_S \neq \mathcal{Y}_T \text{ or } P(Y_S|X_S) \neq P(Y_T|XT))$$

P.-H. (Eddy) Su and Y. Li, 2015

ML vs. TL



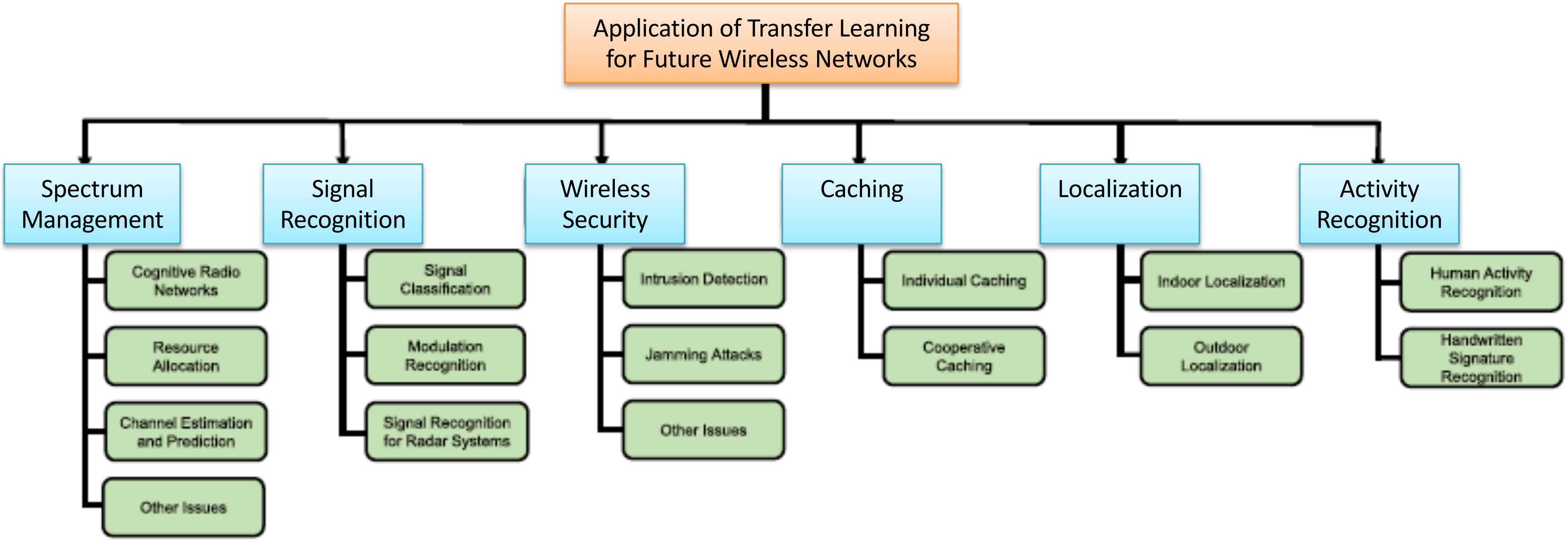
Langley '06, Yang et al. '13

Transfer Learning

1. What to transfer
    - Instance transfer : data
    - Feature transfer : features
    - Model transfer : parameters or priors of models

2. When to transfer
    - When there exists relatedness between source and target domains
        - Transferring knowledge from unrelated domain can be harmful : Negative transfer

Application of Transfer Learning for Future Wireless Networks



Nguyen et al., 2022

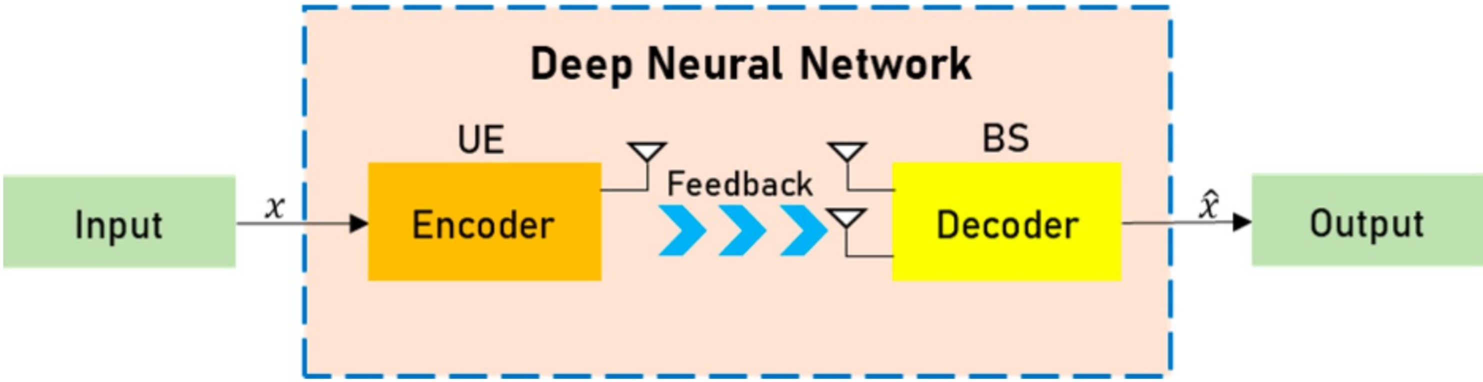
Transfer Learning : Spectrum Management

- Cognitive radio
  - Spectrum sensing
  - Channel selection
  - Interference management
  - Radio map construction
- Resource allocation
  - Channel assignment
  - Power allocation and energy efficiency
  - Resource block allocation
  - Resource utilization prediction
- Channel estimation and prediction
- Other issues

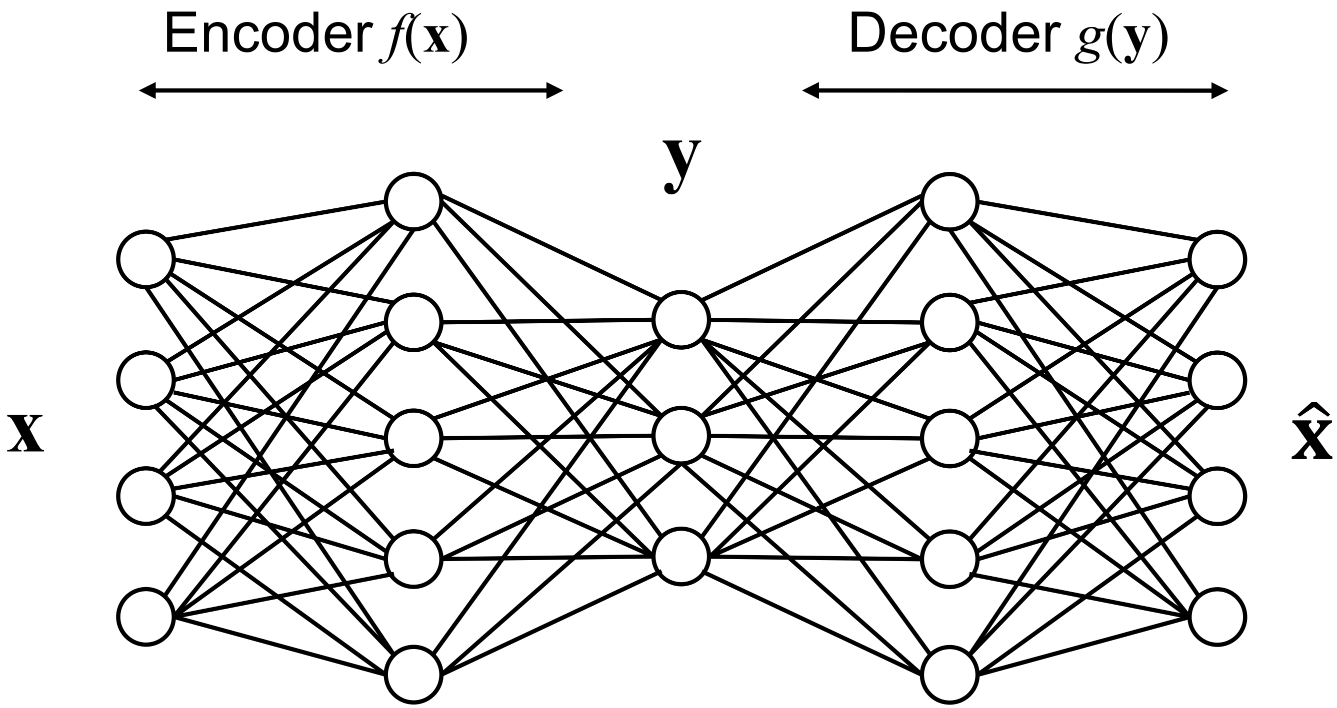
Nguyen et al., 2022

Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems

- J. Zeng et al., "Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems," IEEE Trans. Cogn. Commun. Netw., 2021



Autoencoder

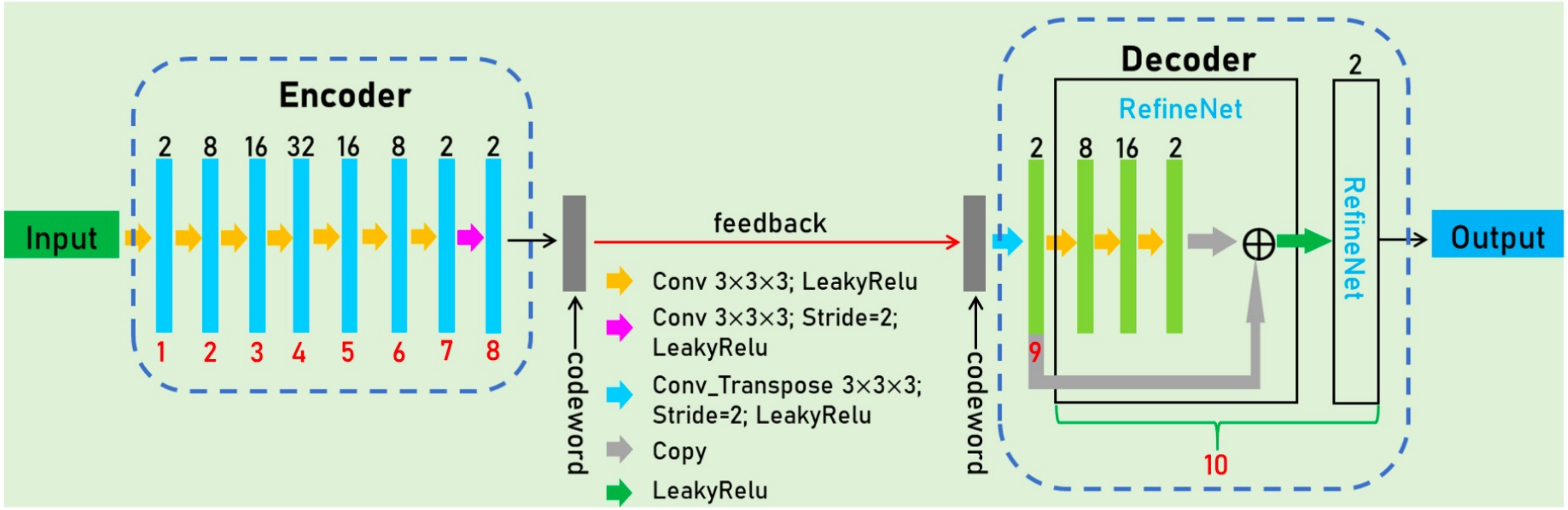


Find a useful representation $\mathbf{y} \in \mathbb{R}^r$ of $\mathbf{x} \in \mathbb{R}^n$ at intermediate layer through learning to reproduce the input at the output

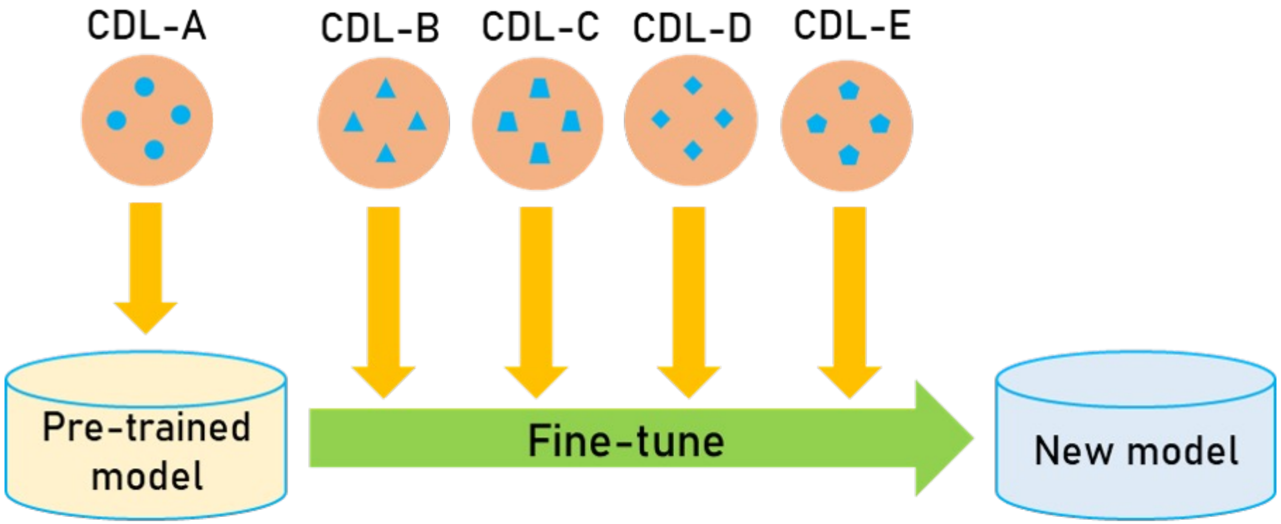
- Incomplete autoencoder: $r < n$
- Overcomplete autoencoder: $r \geq n$

Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems

- J. Zeng et al., "Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems," IEEE Trans. Cogn. Commun. Netw., 2021



Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems

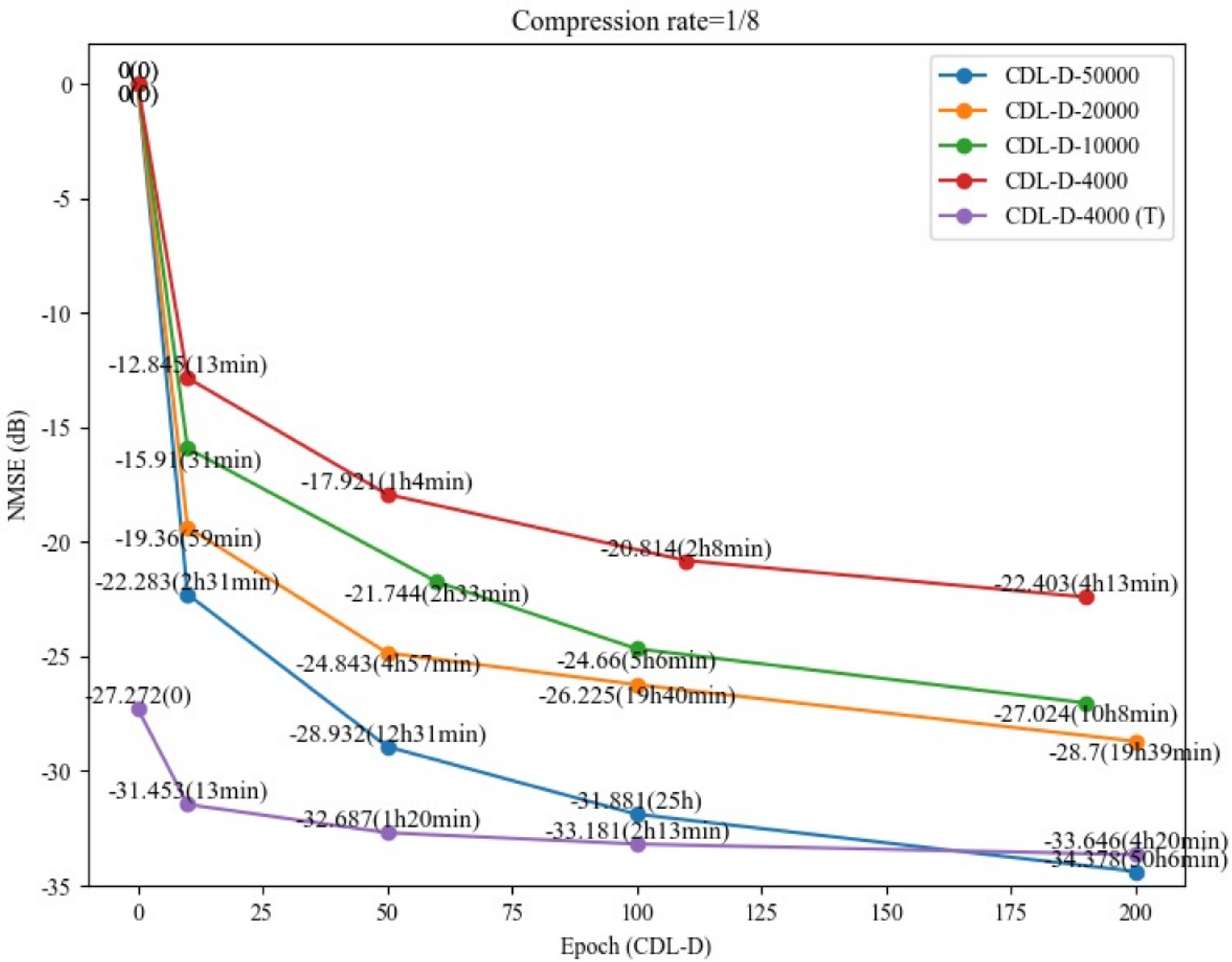


Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems

TABLE I
SIMULATION PARAMETERS.

Item	Value
Frequency of the uplink	2.0 GHz
Frequency of the downlink	2.1 GHz
The numbers of antennas of UE	2
The numbers of antennas of BS	32
The number of subcarriers	72
Subcarrier spacing	15 kHz
The number of OFDM symbols	14
The number of source data samples	50,000
The compression ratio	1/8
The number of target data samples	200, 500, 1000, 2000, 4000

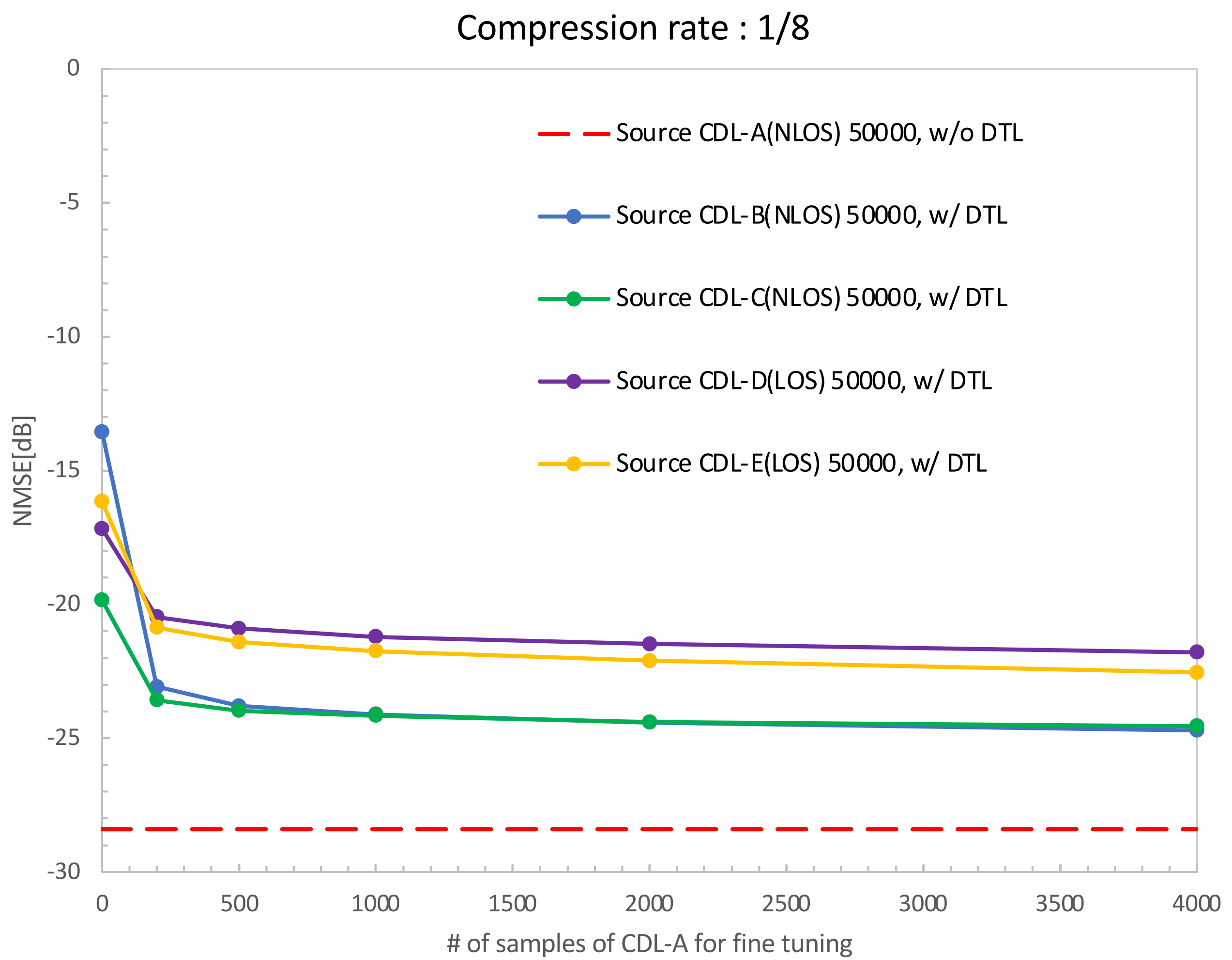
Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems



Keio University

Tomoaki Ohtsuki

Evaluation of Source Data Selection for DTL Based CSI Feedback Method in FDD Massive MIMO Systems



Meta Learning

- Insufficient labelled data result in poor prediction performance
- Start from scratch is always time-consuming
- If we have already learned many tasks, can we figure out how to learn more efficiently?
- Can we learn to learn?

- Meta learning = Learning to Learn

C. Finn and S. Levine, 2019

Model-Agnostic Meta-Learning (MAML)

Setting:

- Multiple tasks are trained during training and new tasks are trained during testing with a small amount of training data

MAML

- Model parameter : θ
- Model represented by the model parameter θ : f_θ
- Update the model parameters by stochastic gradient descent (SGD) to minimize the objective function (loss function) $\mathcal{L}_{\mathcal{T}_i}$ for task $\mathcal{T}_i$. If there are multiple sets of training data, this SGD is repeated multiple times.

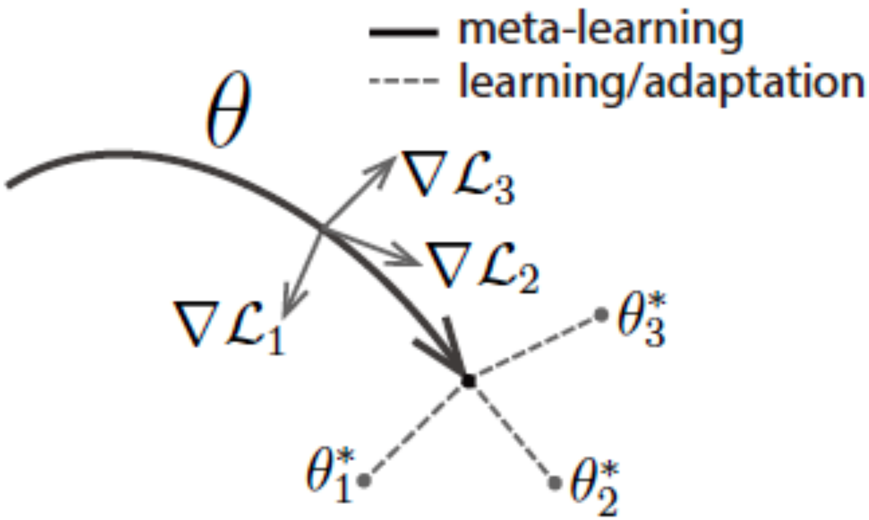
$$\theta_i' := \theta_0 - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$$

- Initial parameters are determined so that the sum of the objective function after updating becomes small.

$$\theta_0 = \arg\min_{\theta_0} \sum_i \mathcal{L}_{\mathcal{T}_i}\left(f_{\theta_i'}\right)$$

- Find a good initial value for each task that will work well after updating in SGD from the current initial value.

Model-Agnostic Meta-Learning (MAML)

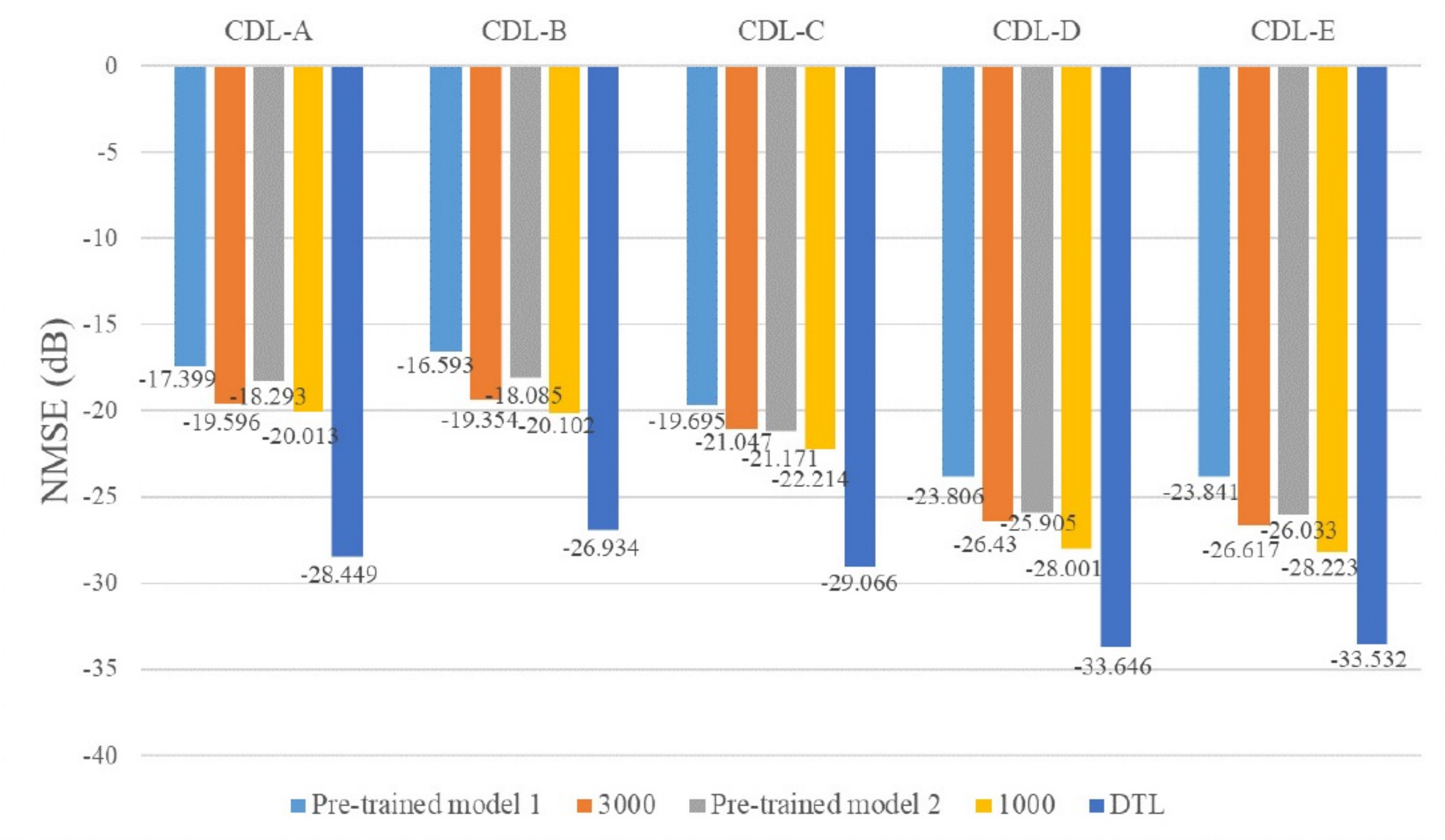


Algorithm 1 Model-Agnostic Meta-Learning

Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters
1: randomly initialize θ
2: **while** not done **do**
3: Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
4: **for all** $\mathcal{T}_i$ **do**
5: Evaluate $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ with respect to K examples
6: Compute adapted parameters with gradient descent: $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$
7: **end for**
8: Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i'})$
9: **end while**

C. Finn et al., 2017

Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems : DTL and MAML



Downlink CSI Feedback Algorithm with Deep Transfer Learning for FDD Massive MIMO Systems : DTL and MAML

Algorithm	Batch size	GPU memory usage	Training time
DTL	50	4433M	50h12min
MAML	20	8531M	15h42min

- MAML is computationally expensive due to high GPU memory usage. However, training time is short and the time cost is low.
- DTL is better when a large amount of data can be collected for each environment
- On the other hand, MAML is also good if the data collected for each environment is small, although there is some degradation of characteristics.

DL-aided Communications and Sensing

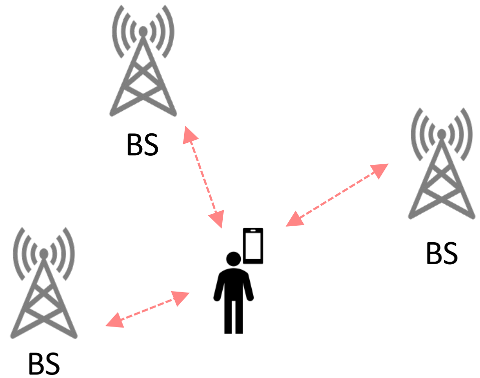
- S. Yang, M. Bouazizi, Y. Cao, and T. Ohtsuki, "Inter-User Distance Estimation Based on a New Type of Fingerprint in Massive MIMO System for COVID-19 Contact Detection," Sensors, vol. 22, issue 16, 22 pages, 2022.

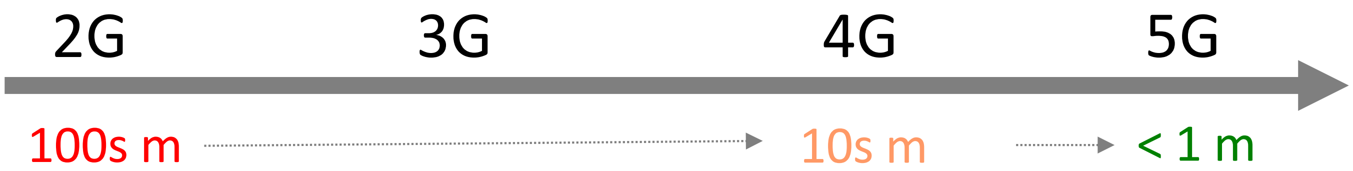
- M. Bouazizi, S. Yang, Y. Cao, and T. Ohtsuki, "A Novel Approach for Inter-User Distance Estimation in 5G mmWave Networks Using Deep Learning," APCC2021, Virtual, Oct. 2021

Background

Localization in cellular networks:

- Estimate the location of the target object
- Rely on a set of wireless signals propagated between BSs and the target.





Co-located devices identification:

- Identify devices in proximity of one another
- Estimate inter-device distance → Similar to localization

➔ Exposure to Covid-19: requires estimation of the distance between the virus carrier and the subject ➔ Accurate estimation leads to better identification of exposure

Keio University
1858
CALAMVS GLADIO FORTIOR

Localization and Co-location techniques -1/2-

[1] Z. Yang et al., ACM CSUR, 2013.
[2] A. A. Wahab et al , IEEE ITST, 2013.
[3] S. Kumar et al, ICMCN 2014.
[4] Q. Vo and P. De, IEEE CST, 2015.
[5] M. Li and Y. Liu, ICMCN, 2009.

Geometry-based techniques:

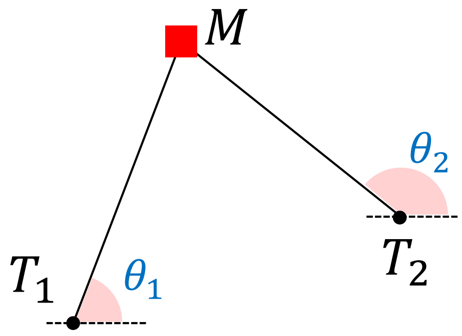
- Refer to the geometric properties of the signal propagation
- RSS-based [1], TOA-based [2], AOA/AOD-based [3]

Fingerprint-based techniques [4]:

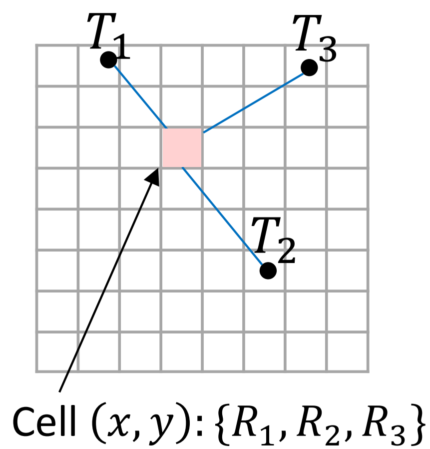
- Make location specific database of features
- Refer to the database to identify the location of users

Proximity-based techniques [5]:

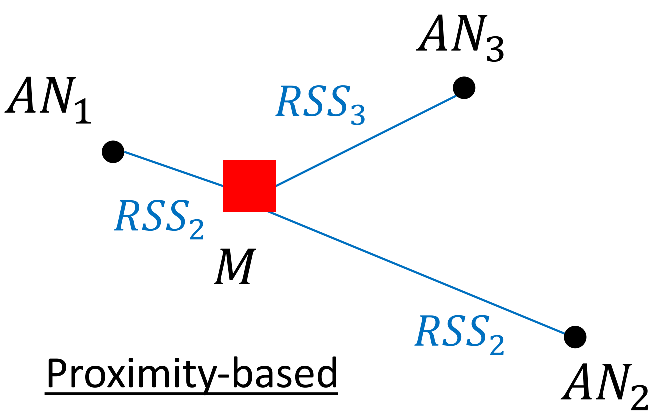
- Refer to RSS measurements of emitting devices:
  - Identify the location of UEs
  - Identify co-located UEs



AOA/AOD-based



Fingerprint-based



Proximity-based

RSS: Received Signal Strength
TOA: Time of Arrival
AOA: Angle of Arrival
AOD: Angle of Departure
LOS: Light of Sight

Co-location techniques -2/2-

BLE: Bluetooth Low Energy
UE: User Equipment

Existing Co-location techniques

- Use similar techniques to Localization:

1. Rely on Wi-Fi and Bluetooth signals for proximity-detection [6–9]:
   - In [6],[7] and [8]: Rely on RSS of iBeacons, other BLE devices, and ambient Wi-Fi signals to identify groups of people walking together: UEs experiencing the same levels of signals from the same devices → in proximity
   - In [9]: for spaces with "dense" distribution of Wi-Fi hotspots (e.g., malls), use the RSSI of the Wi-Fi signal to identify UEs in proximity

     ➔ Good clustering and vicinity detection accuracy (Over 98% accuracy)

     ➔ Mostly focus on indoor environment,

     ➔ Works for short range

     ➔ Identify UEs in vicinity rather than estimate the distance between them

2. Rely on Localization techniques in cellular networks for Co-location [10]

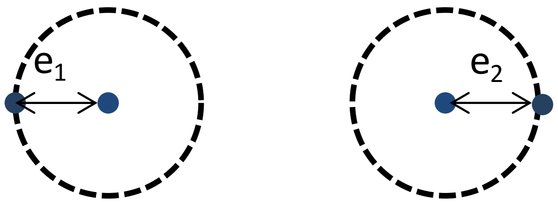
   ➔ Imprecision from the localization techniques are inevitably inherent when using these techniques for co-location

[6] M. Pedro and T. Ohtsuki, *IEEE Access*, 2016.
[7] R. Canetti et al., arXiv 2020.
[8] M. Pedro et al. *IEEE IoT*, 2017
[9] M. Dmitrienko et al., *ScienceOpen,* 2020.
[10] F. Hejazi et al., *arXiv* 2021.

Keio University
CALAMVS GLADIO FORTIOR
1858

Conventional Method

- Use location Information to estimate inter-UE distance [10]
  - Use a Deep Convolution Neural Network (DCNN) to estimate the UE location
    - DCNN: learn the mapping between angle delay profile (ADP) and user location
    - ADP: a linear transformation of channel state information (CSI), it can represent the relation between the absolute gain of AOA and delay.
  - Use the found locations to estimate the distance between users
- Limitation
  - Only minimizing the error between the estimated location and UE location:
    - Does not directly minimize the error between two different UE locations
    - Estimation error between two UE can go up to e1 + e2



● user location
● estimation location
e_1, e_2: estimation error of user location

[10] F. Hejazi et al., arXiv 2021.

Objective and Contributions

Objective:

- Improve the accuracy of co-location in cellular systems
- Target system: mmWave 5G cellular network

Contributions:

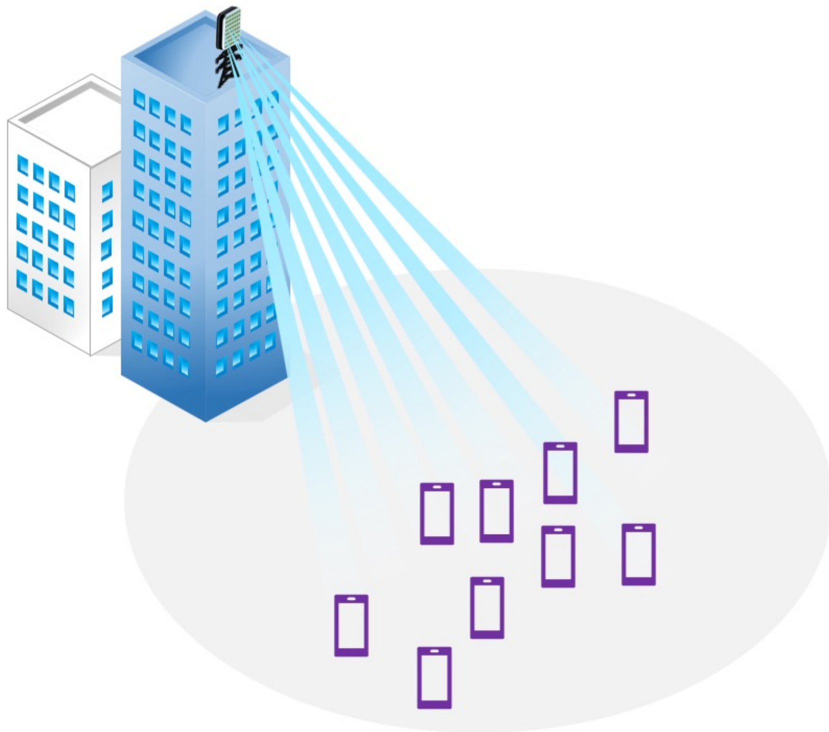
- Propose a novel fingerprint-like method for co-location → robust to changes over time in the environment, and to change in environment.
- Estimate the distance between each pair of user equipments (UEs), even apart ones → Improve the estimation accuracy of Inter-UE distance, compared to existing work
- Propose a technique to reduce the time and power consumption of the original proposed method without much loss of estimation performance.

Keio University
CALAMVS GLADIO FORTIOR
1858

System description -1/2-

System:

- A mmWave massive MIMO system
- Use uniform planar array (UPA) with $M_t = M_v \times M_h$ antennas at the BS (transmitter) and single-antenna at K UEs (receivers)

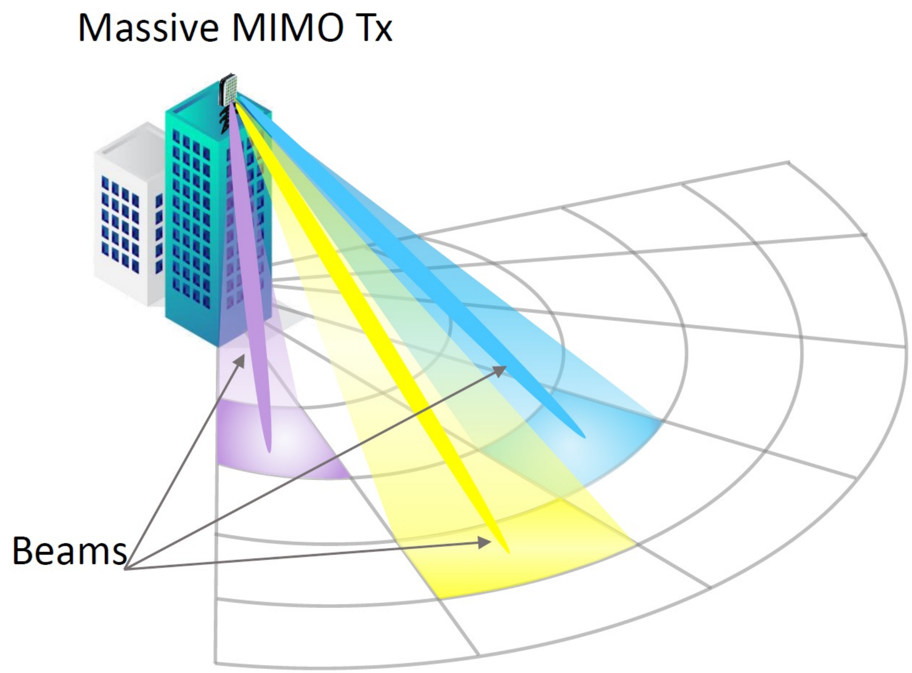


Keio University
1858
CALAMVS GLADIO FORTIOR

System description -2/2-

- Beam sweeping in consecutive time slots.

- UEs located at different locations in the coverage of the BS record measurements of the received beams.



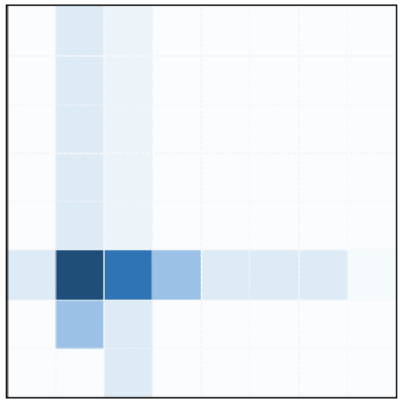
- The recorded measurements are used as a fingerprint of the users' locations

- Unlike fingerprint-based localization conventional methods, there is no need for offline fingerprint learning or building a database of features

Proposed Approach - Fingerprint Image Generation

- The BS sweeps beams in consecutive timeslots.
- We refer to the number of beams needed to "cover" the entire region as $N_b = n_b \times n_b$.
- The RSSI of each received beam at the k-th UE end is recorded as $m_k^{(x,y)}$ for $x \in 0, \cdots, n_b$ and $y \in 0, \cdots, n_b$
- For all the beams, we define the beam-based image M_k of the k-th UE as

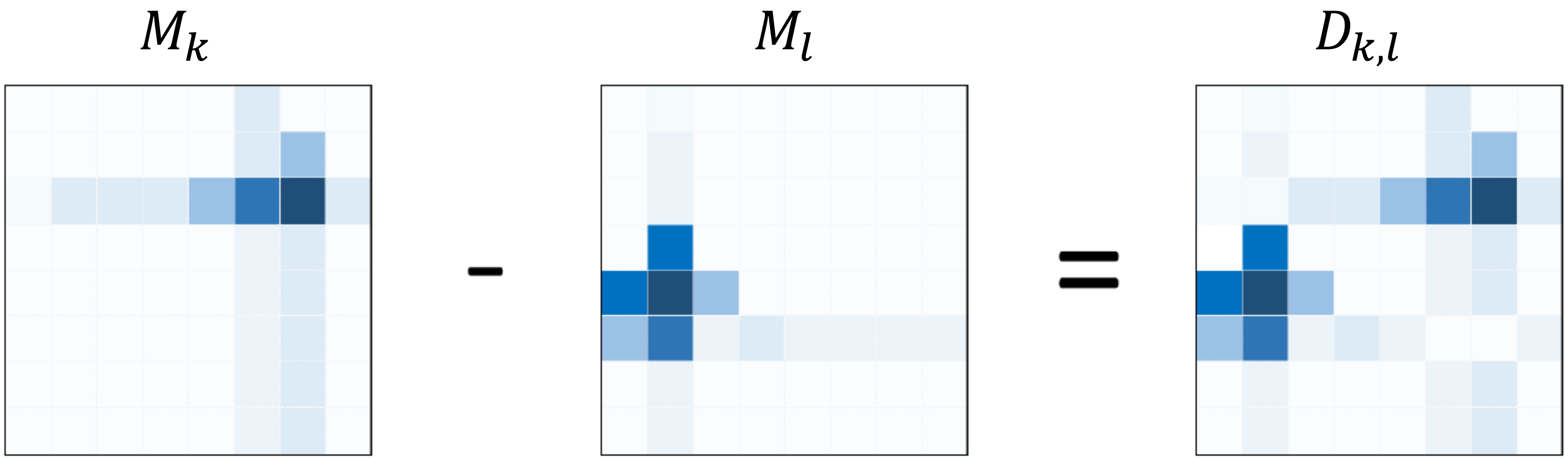
$$M_k = \begin{bmatrix} m_k^{(1,1)} & \cdots & m_k^{(1,n_b)} \\ \vdots & \ddots & \vdots \\ m_k^{(n_b,1)} & \cdots & m_k^{(n_b,n_b)} \end{bmatrix}$$



Proposed Approach - Image Difference measurement

- Given two UEs k and l, with their respective measurement matrices / images, we define the difference matrix $D_{k,l}$ as:

$$D_{k,l} = \begin{bmatrix} |m_k^{(1,1)} - m_l^{(1,1)}| & \cdots & |m_k^{(1,n_b)} - m_l^{(1,n_b)}| \\ \vdots & \ddots & \vdots \\ |m_k^{(n_b,1)} - m_l^{(n_b,1)}| & \cdots & |m_k^{(n_b,n_b)} - m_l^{(n_b,n_b)}| \end{bmatrix}$$



- D is obviously commutative ($D_{k,l} = D_{l,k}$)

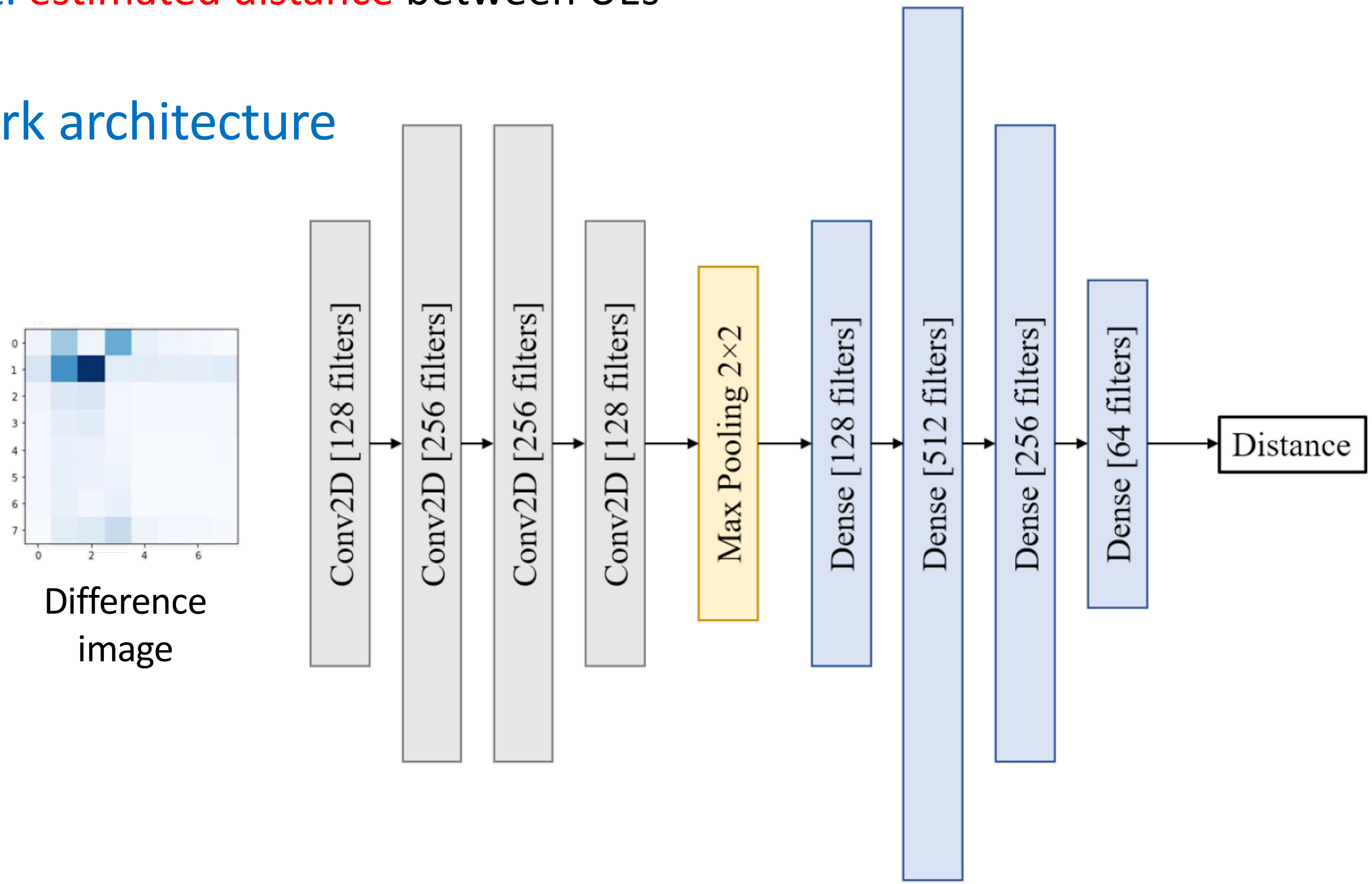
Keio University
1858
CALAMVS GLADIO FORTIOR

Proposed Approach - Inter-UE distance estimation

- Use Deep Learning-based regression:
  - Input: difference Image
  - Output: estimated distance between UEs

Total Number of Parameters: 2,444,226

Neural Network architecture



<u>Neural network used for inter-UE distance estimation</u>

Keio University

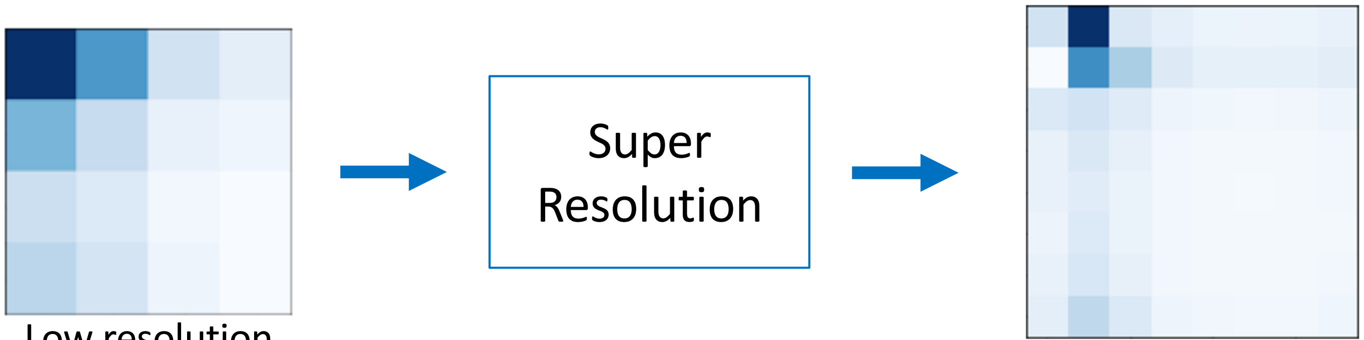
Proposed Approach - Reduce Resource consumption

Limitations

- Narrower beams = More beams needed ➔
  - ↗ Higher Accuracy
  - ↘ Power resources consumption
  - ↘ More time required for the sweeping

Proposed method:

- Use wide beams ➔ Apply Super Resolution (SR) to improve the generated fingerprint image quality



Low resolution difference image

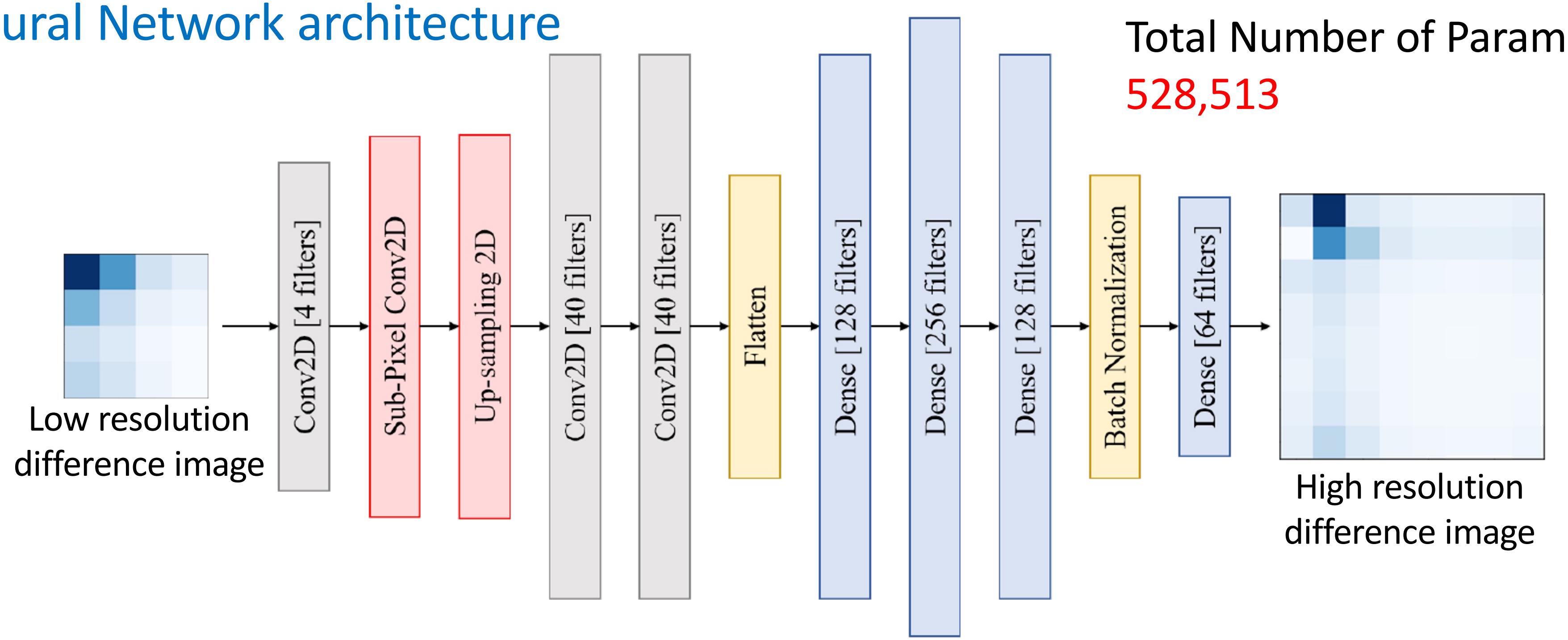
Super resolution difference image

Proposed Approach - SR Neural Network Architecture

- Use Deep Learning-based Super Resolution:
  - Input: low resolution difference image
  - Output: high resolution difference image
- Use Sub-Pixel Convolution and 2D Up-sampling
- Upscale the image ×4: in our experiment 4×4 → 8×8

Neural Network architecture

Total Number of Parameters: 528,513



<u>Neural network used for Super Resolution</u>

Keio University

Simulation Parameters

Parameter description	Value
Carrier frequency	60 GHz
# of antennas at the BS ($M_v \times M_h$)	8×8
# of DTF codebook beams N_b	8×8 / 4×4
UE spread area	60 m × 30 m
UE average distance to the BS	24.2 m
Height of BS	10 m
Total downlink power P	30 dBm
Building and ground reflection gain g	-6 dB
Noise figure F	9.5 dB

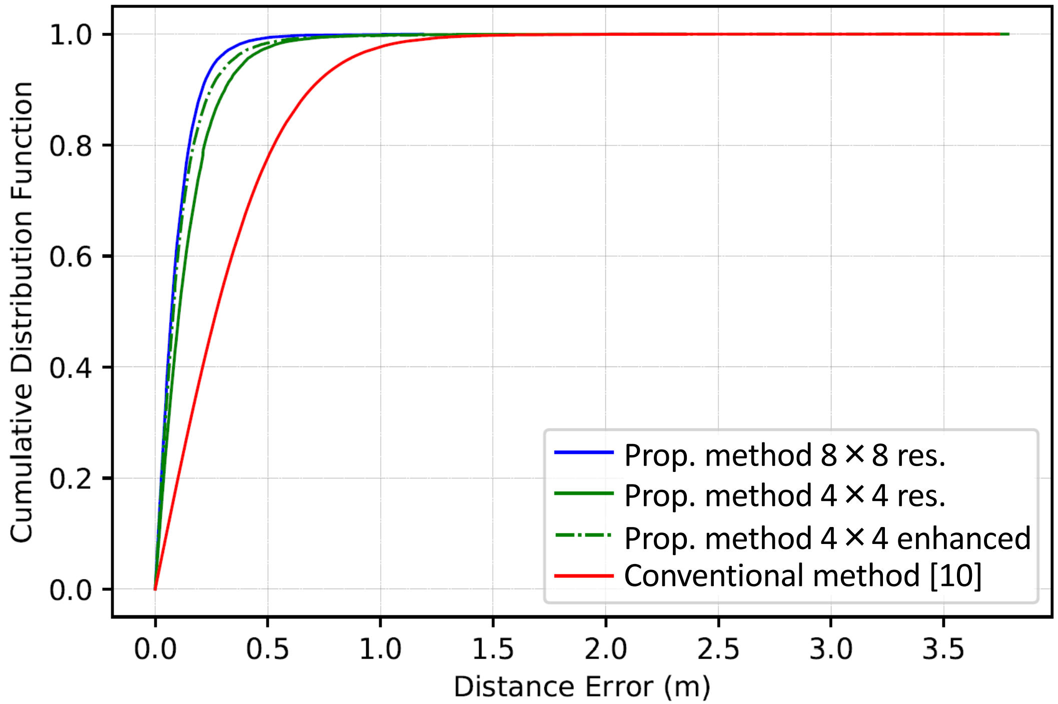
Evaluate the proposed method against [10]:

- Use a Deep Convolution Neural Network (DCNN) to estimate the UE location
- DCNN: learn the mapping between angle delay profile (ADP) and user location

.

[10] F. Hejazi et al., *arXiv* 2021.

CDF of Distance Error Over All Pairs of Users



CDF of distance error over all pairs of users

At CDF = 0.5:

- Prop. method [8×8 res.]: 0.097 m
- Prop. method [4×4 res.]: 0.160 m
- Prop. method [4×4 Super res.]: 0.101 m
- Conv. method: 0.280 m

At CDF = 0.9:

- Prop. method [8×8 res.]: 0.231 m
- Prop. method [4×4 res.]: 0.344m
- Prop. method [4×4 Super res.]: 0.254 m
- Conv. method: 0.753 m

➔ Much better accuracy than the conventional method

➔ Reduce the estimation error by 26% after applying Super Resolution to images of size 4×4

Keio University

Summary

- AI is a key enabler of 6G
  - Great potentials for addressing big data issues to achieve high requirements in complex and dynamic environments
  - Knowledge about AI becomes essential for people in communication fields