Enhancing AI-Supported Channel Estimation in MIMO Systems with Open Set Recognition

Huiwen Zhang*, Venkataramani Kumar†, Feng Ye*, Rose Qingyang Hu‡ and Yi Qian§

*Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison, WI, USA

†Department of Computer Science and Engineering, University of Puerto Rico at Mayaguez, PR

‡Department of Electrical and Computer Engineering, Utah State University, Logan, UT, USA

§Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, Omaha, NE, USA

Emails: *{hzhang2279, feng.ye}@wisc.edu, †venkataramani.kumar@upr.edu, ‡rose.hu@usu.edu, §yi.qian@unl.edu

***Abstract*—Accurate channel estimation is required for various multiple input multiple output (MIMO) implementations in the next-generation wireless communication systems. Recently, Artificial Intelligence (AI) techniques have been introduced for channel state information (CSI) processing in channel estimation because of their accuracy and relatively low complexity compared to the traditional approaches. However, these AI-supported CSI processing models are usually developed with a fixed training dataset. Therefore, the performance of such approaches cannot be guaranteed in new environments. This paper focuses on enhancing AI-supported channel estimation methods with a 2-stage open set recognition scheme. New environments are detected in the first stage by identifying different characteristics between testing and training data. In the second stage, new data is filtered and further categorized to each individual environment. Simulation results using four different environment settings demonstrate that the proposed method can greatly enhance the usability of AI-supported channel estimation.**

I. Introduction

The next-generation wireless technologies play an important role in the realization of futuristic applications due to the numerous benefits such as higher throughput, massive interconnectivity, and ultra-low latency [1]. Multiple-input multiple-output (MIMO) is one of the emerging technologies supporting the next-generation wireless networks [2]. However, for the successful realization of MIMO, it is imperative to estimate the channel state information (CSI) with higher accuracy. In a traditional frequency-division duplex (FDD) system, CSI is estimated by reconstructing the channel information feedback from user equipment (UE) based on pilot signals sent by the base station (BS) [3]. The multi-step channel estimation process introduces challenges such as higher computational complexity and time. Due to the virtue of channel reciprocity, the channel information can be derived with reduced computations in the time-division duplex (TDD) systems [3]. Regardless of the methodologies adopted by the system, it is imperative to ensure accurate computation of channel information in any environmental conditions and scenarios. In this work, we focus on enhancing Artificial Intelligence (AI) supported CSI feedback processing methods with Open Set Recognition (OSR) for an FDD system.

Recently, AI has been introduced to support CSI processing in channel estimation [4–10]. In specific, CsiNet was one of the first deep-learning approaches for CSI processing [4]. Subsequent development of CsiNet has been based on Long-Short-Term Memory (LSTM) [5], Generative Adversarial Network (GAN) [6], and transformer [7, 8]. Moreover, other approaches such as CRNet [9] and MRFNet [10] have been developed to enhance CSI feature extraction across multiple resolutions and recover features with different receptive fields for CSI processing in an FDD system. However, most of those existing works assume the same dataset for training and testing their deep-learning models for CSI processing. Therefore, the performance of those approaches cannot be guaranteed in a wireless environment different from the initial dataset.

In this paper, we propose to tackle this issue and enhance AI-supported CSI processing methods by with a 2-stage approach. First, alert when the model is deployed in unknown environments. Second, data corresponding to each individual environment should be collected and labeled autonomously. Broadly speaking, identifying unknown/unlabeled data in an unsupervised manner is closely related to the Open Set Recognition (OSR) problem [11]. For example, authors in [12] proposed a method that uses class-conditioned auto-encoders to do OSR. Authors in [13] use a few data-trained auto-encoder to classify open-set audio. However, these solutions are more commonly associated with classification and recognition. Hence, these solutions cannot be applied directly to AI-supported CSI processing methods. Nonetheless, we borrow the concept of OSR to address the two aforementioned challenges in this paper. The first challenge can be addressed by accessing the mean-squared error (MSE) value of the testing data. Generally speaking, the MSE results from an unknown environment should be higher than that during the training phase of an AI model. Hence, a drastic increase in MSE can indicate the presence of unknown environments. To address the second challenge, we proposed a method based on one-epoch transfer learning. Transfer learning enhances a pre-trained model in one domain by leveraging information from a closely related domain. Instead of enhancing an AI-assisted CSI processing model, the transfer learning process only includes 1 epoch, which is lightweight and effective in separating the data from multiple environments. Evaluation is conducted using simulated data in four different wireless environments. Compared to an existing method, the 2-stage approach proposed in this work can quickly detect unknown environments and correctly collect and label data from each individual environment.

The remainder of the paper is organized as follows. Section II describes the overview of the proposed framework. Section III presents the detailed methods of solving the problem, while Section IV elicits evaluation results. Section V concludes the paper.

II. Studied Network System and Preliminaries

The proposed work focuses on AI-supported CSI processing for an Orthogonal Frequency-Division Multiplexing (OFDM) based FDD MIMO system. Given a pilot sequence, the received signal at k^{th} subcarrier is written as

$$\mathbf{y}_k = \hat{\mathbf{h}}_k^H \mathbf{v}_k + w_k, \tag{1}$$

where $\hat{\mathbf{h}}_k^H$ is the channel frequency response, and $\mathbf{v}_k$ is the precoding vector [2]. The channel can be estimated using the Least-Square (LS) method:

$$\hat{\mathbf{H}}_{\text{LS}} = \frac{1}{X_P} Y_P, \tag{2}$$

where $\hat{\mathbf{H}}_{\text{LS}}$ refers to the estimated channel matrix using LS method, X_P, and Y_P refers to the original and received pilots respectively. Without loss of generality, an auto-encoder structure is assumed in a given AI-supported CSI processing method. As shown in Fig. 1, an encoder is deployed on the User Equipment (UE) side, and a decoder is deployed on the BS side, respectively. The encoder is to compress the estimated channel information $\hat{\mathbf{H}}_{LS}$ into codeword $\mathbf{S}$. Once sent back to the BS, codeword $\mathbf{S}$ can reconstruct the CSI by the decoder. Note that the studied system assumes practical implementation of such CSI processing methods, supporting multiple UEs from different yet new environments to the initialized model. For better illustration, the notations used in the rest of this work are summarized in Table I.

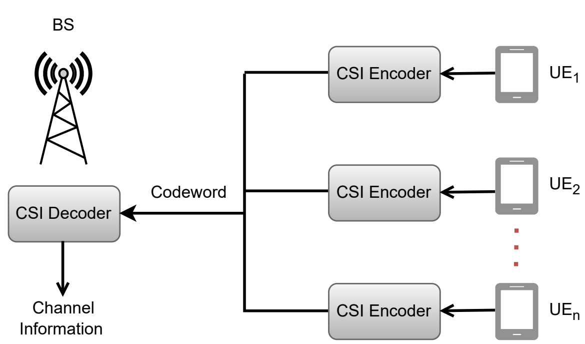


Fig. 1: Overview of studied MIMO system with AI-supported CSI processing.

However, for improved performance and usability of such AI-based CSI estimation techniques, it is imperative to understand two main considerations. First, the compatibility of the AI model with any given environment must be accessed. Second, categorizing the new or unknown data from a different environment encountered autonomously.

TABLE I: Notations used in this work.

Notation	**Remarks**
N_t	# Transmit antennas
N_r	# Receive antennas
k	Subcarrier
y_k	Received signal at the k^{th} subcarrier
f_c	Center frequency
$\hat{\mathbf{h}}_k^H$	Channel frequency response
$\hat{\mathbf{H}}_{\text{LS}}$	Estimated channel matrix using LS
X_P	Pilot sequences
Y_P	Received pilots
$\mathbf{v}_k$	Precoding vector
x_k	Transmitted symbol
X	Piece of CSI data with n elements
$\hat{X}$	Predicted CSI data
S	Codeword
N_{S1}	# of the local minima in stage 1
N_{S2}	# of the local minima in stage 2
N_{new_env}	# of the new founded environments
$f(x)$	estimated density function
n	# of data points
G	Gaussian Kernel
h	bandwidth of the KDE

III. 2-Stage Approach for CSI Processing OSR

An overview of the proposed 2-stage OSR method is depicted in Fig 2. The first stage is to identify whether the AI-supported CSI processing model is being used in unknown environments. The second stage is to further label each individual environment and collect corresponding datasets. For better illustration, it is assumed that the ground truth of CSI can be obtained and transferred to the BS, e.g., via traditional methods.

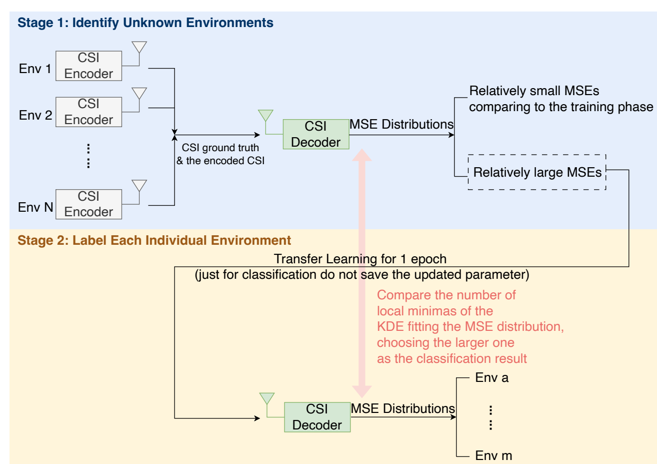


Fig. 2: Overview of the proposed 2-stage OSR method.

A. Stage 1 - Identify Unknown Environments

Given an AI-supported CSI processing model, its performance on channel reconstruction can be evaluated by the MSE metric, computed as:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \hat{X}_i)^2, \tag{3}$$

TABLE II: Settings of the four tested environments.

	Indoor1	Indoor2	Outdoor1	Outdoor2
# of Paths	3	7	6	2
Path delays	[0 3e-6 5e-6]	[0 3e-4 5e-5 7e-5 1e-6 9e-4 2e-7]	[0 2e-7 8e-8 1.2e-6 2.3e-6 3.7e-6]	[0 2.5e-5]
Average path gains	[0 -1 -2]	[0 0.2 0.5 0.1 -3 -1.2 -1]	[0 -0.9 -4.9 -8 -7.8 -23.9]	[0 -3]
Fading distributions	Rayleigh	Rayleigh	Rician	Rician
Fading technique	Filtered Gaussian noise	Sum of Sinusoids	Filtered Gaussian noise	Sum of Sinusoids

where X is a piece of CSI data with n elements and $\hat{X}$ is the predicted CSI data from the model, which has the same shape as X. If higher MSE values are observed compared to that during the training phase, the model is either under-trained or not trained with respect to the testing environment. In other words, a clear separation in MSE distributions can alert the presence of unknown environments in this stage. Technically, Kernel Density Estimation (KDE) is implemented to find the separations. KDE is a non-parametric way to estimate the probability density function (PDF) of a random variable [14]. This method is particularly useful for smoothing out the data distribution and is often employed when the underlying data distribution is unknown. KDE works by placing a continuous kernel function at each data point and then averaging these kernels to produce a smooth estimate of the density. The estimated density function in this study can be expressed as:

$$f(x) = \frac{1}{nh} \sum_{i=1}^{n} G\left(\frac{\text{MSE} - \text{MSE}_i}{h}\right). \tag{4}$$

By calculating the local minima of the KDE, the MSE distributions can be separated. However, the MSE distributions among different unknown environments may not be clearly separated. Hence, newly collected data cannot be labeled correctly according to their originating environments according to the results in stage 1.

B. Stage 2 - Label Each Individual Environment

A one-epoch transfer learning process is implemented in stage 2 to provide clear separation among different unknown environments. In particular, the one-epoch transfer learning can result in an MSE distribution that is different from that of the previous ones. It may be noted here that transfer learning is utilized as a tool to categorize the data in this work. The same KDE method is applied after the 1-epoch transfer learning to separate the MSE distributions. Note that the local minima in stage 1 (N_{S1}) and stage 2 (N_{S2}) can be vastly different. The higher count of the local minimum points in either stage can be interpreted as the number of unknown environments. Note that in stage 1, since the initial environment of the training data is included and its corresponding MSE values are much smaller, the number of the local minima is at least one with the presence of unknown environments. Therefore, when comparing the number of local minima in stage 2, the number of local minima in stage 1 should decrease by one. The total number of unknown environments is the maximum number of local minima between stage 1 and stage 2 results plus 1. The data can then be labeled according to the corresponding MSE distributions separated by the local minima points.

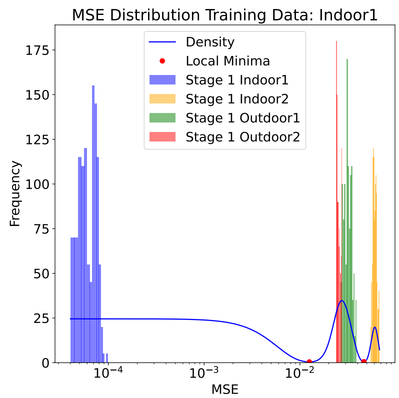
Algorithm 1 Proposed 2-stage OSR method.

Input: Multiple environments data
Output: N_{new_env}
Initialisation :
A pre-trained CSI feedback model
Stage 1
Input all data into the model
KDE fitting MSE distribution
number of local minima N_{S1}
Filter the data MSE $\leq$ the smallest minima
Stage 2
Input existing data into the model
KDE fitting MSE distribution
number of local minima N_{S2}
if $N_{S1} - 1 \neq N_{S2}$ **then**
  $N_{new\_env} = \max(N_{S1} - 1, N_{S2}) + 1$
end if
if $N_{S1} - 1 = N_{S2}$ **then**
  Cross reference between two stages
end if
return N_{new_env}

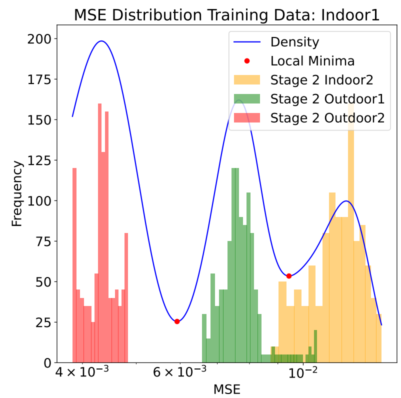
IV. Evaluation Results

A. Evaluation Settings and Dataset

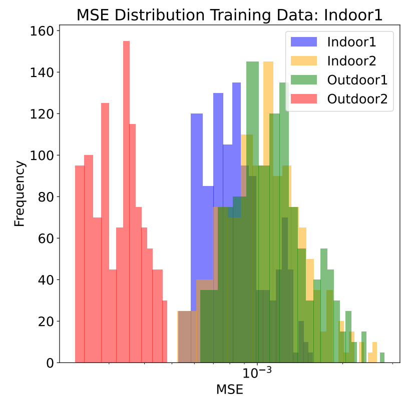
Evaluation is conducted on a synthetically generated dataset on a 32×32 MIMO system pertaining to four different environments using MATLAB. Detailed settings for the four environments are summarized in Table II. All four environment simulations have the same center frequency at 2.4 GHz and signal-to-noise ratio (SNR) at 15 dB. Frequency-independent parameters such as the number of paths, path delays, and path gains are collected from the different studied environments in a static condition. The other necessary system settings are summarized in Table III. For each environment, 55 OFDM symbols are generated with a symbol length of 217. For each environment, 50 OFDM symbols are allocated for training, and 5 OFDM symbols are designated for testing. Thus, the number of training samples for each environment is 10850, and 1085 are testing samples. Each channel response is originated as a two-dimensional complex matrix with the shape of 32×32. It is then decomposed into its real and imaginary components and forms a three-dimensional matrix with dimensions $2 \times 32 \times 32$. CsiNet is chosen as the AI-supported CSI processing method for the evaluation. CsiNet is an auto-encoder approach. In the encoder design, the first layer is a convolution layer, followed by a batch norm layer



(a) Results of stage 1 processing.

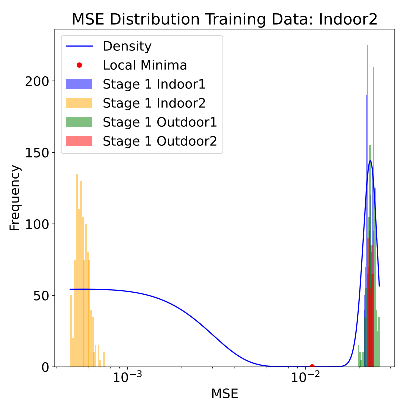


(b) Results of stage 2 processing.

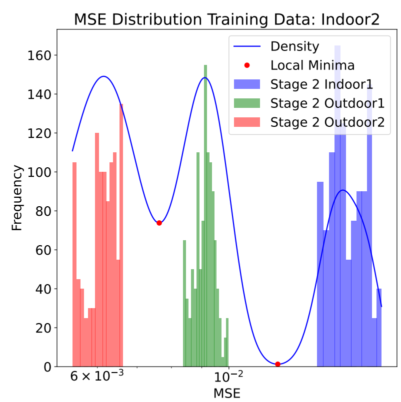


(c) Separation results from existing method.

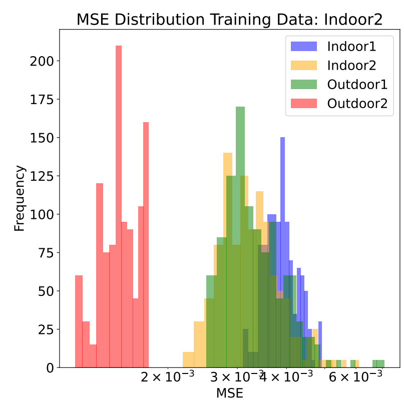
Fig. 3: The MSE distributions of results from AI model initialized using Indoor1 dataset.



(a) Results of stage 1 processing.

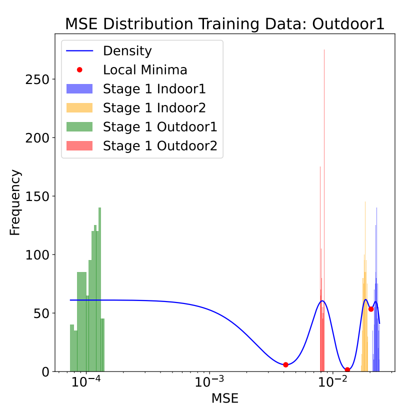


(b) Results of stage 2 processing.

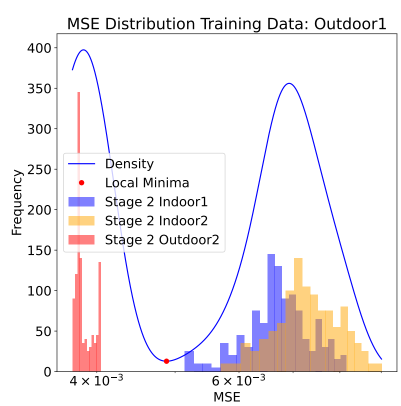


(c) Separation results from existing method.

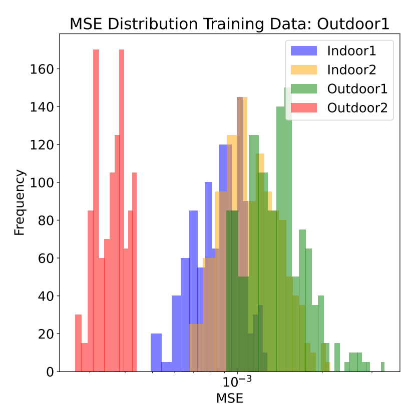
Fig. 4: The MSE distributions of results from AI model initialized using Indoor2 dataset.



(a) Results of stage 1 processing.

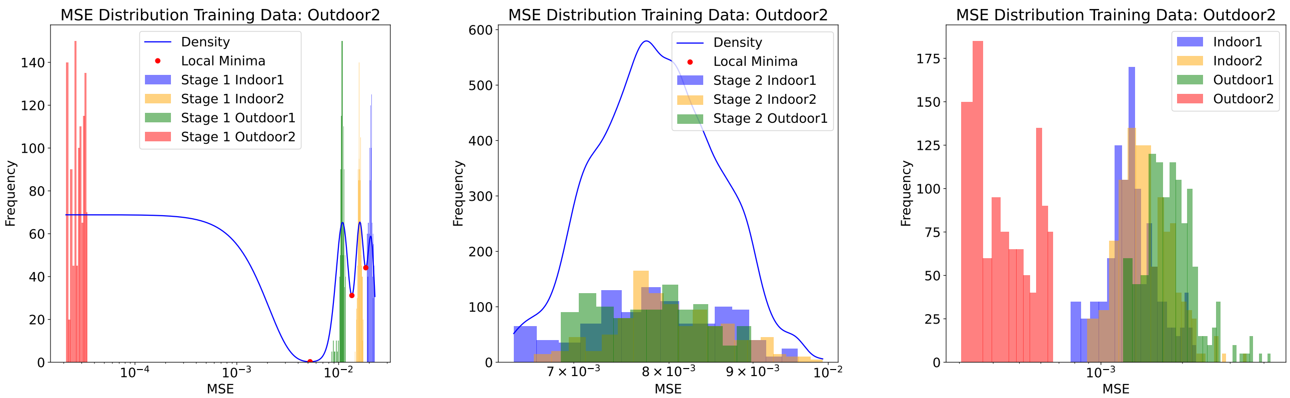


(b) Results of stage 2 processing.



(c) Separation results from existing method.

Fig. 5: The MSE distributions of results from AI model initialized using Outdoor1 dataset.



(a) Results of stage 1 processing.

(b) Results of stage 2 processing.

(c) Separation results from existing method.

Fig. 6: The MSE distributions of results from AI model initialized using Outdoor2 dataset.

TABLE III: Other settings for evaluation.

Parameter	**Settings**
N_t	32
N_r	32
f_c	2.4
Environments	Indoor1/Indoor2/Outdoor1/Outdoor2
SNR (dB)	15
Fading model	Rayleigh/Rican
# training samples	10,850 per testing scenario
# testing samples	1,085 per testing scenario

and a leaky Rectified Linear Unit (ReLu) layer. Finally, a linear layer compresses the output into a lower dimension. In the decoder design, the first layer is a fully connected linear layer, followed by two Residual Conv Units in RefineNet [15] with three convolution layers. A batch normalization layer, a convolution layer, and a leaky ReLu activation layer are used to obtain the final result. Please note that the evaluation can be extended to other methods straightforwardly. Table IV shows the performance of the CsiNet on the initial environments. These MSE values are used as benchmarks in the 2-stage approaches.

TABLE IV: MSE for initial environments (dB).

	Indoor1	**Indoor2**	**Outdoor1**	**Outdoor2**
Average MSE	-42	-32.5	-39.6	-45.5
Min MSE	-44	-34	-41.3	-46.8
Max MSE	-40.1	-31.3	-38.5	-44.7

B. Evaluation Results of Data Separation

Fig. 3(a), Fig. 4(a), Fig. 5(a) and Fig. 6(a) show the stage 1 MSE distributions, the KDE fitting curve, and the calculated local minima under the four environments, respectively. As we can see, the unknown environments are detected in stage 1 processing as local minima from the KDE curve can be found between the lower MSE values from the training environment and the higher MSE values from the testing environments. However, the numbers of local optima do not necessarily represent the number of unknown environments from stage 1 processing. The results from stage 2 processing are shown in Fig. 3(b), Fig. 4(b), Fig. 5(b) and Fig. 6(b), respectively. As we can see, the MSE distributions from unknown environments are randomized further due to the 1-epoch transfer learning. Based on Alg. 1, the corresponding N_{S1}, N_{S2} and N_{new_env} can be found and are listed in Table V. As we can see, all three unknown environments can be identified from the 2-stage approach. Moreover, data labeling can be done perfectly when using Indoor2, Outdoor1, or Outdoor2 as the initial environment. When using Indoor1 as the initial environment, data from Outdoor1 and Indoor2 cannot be separated completely. Nonetheless, the mislabeling data only include the few ones in the overlap area, as shown in Fig. 3(b).

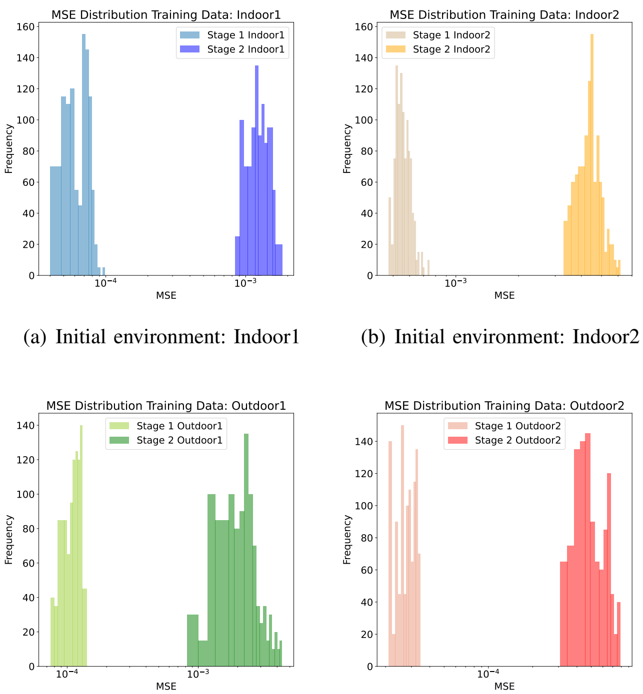
TABLE V: Detection results of unknown environments.

	Indoor1	**Indoor2**	**Outdoor1**	**Outdoor2**
N_{S1}	2	1	3	3
N_{S2}	2	2	1	0
N_{new_env}	**3**	**3**	**3**	**3**
Stage chosen for separation	2	2	1	1

An existing method intended to detect CSI data from unknown environments [16] is implemented for comparison. The existing method is based solely on transfer learning. As shown in Fig. 3(c), Fig. 4(c), Fig. 5(c) and Fig. 6(c), the existing method can barely separate the unknown environments from each other. Therefore, the data cannot be labeled correctly. Moreover, the extensive transfer learning process in the existing method is more complicated and time-consuming compared to the proposed 2-stage process with 1-epoch transfer learning.

C. More Discussions

The proposed transfer learning-based channel classification framework is significant for various reasons. To begin with, this method paves the way towards efficient categorization of channel samples from different environments. Though the principle of channel estimation based on CSiNet is the same regardless of environments and system settings, contamination of testing datasets with unknown samples results in reduced performance and robustness of the system. Therefore, it is necessary to eliminate the unfamiliar samples. And these unfamiliar samples are not useless. Categorizing them can provide data for training models to adapt to their channels. Finally, our framework achieves high accuracy in the unsupervised categorization of unfamiliar data.



(a) Initial environment: Indoor1 (b) Initial environment: Indoor2

(c) Initial environment: Outdoor1 (d) Initial environment: Outdoor2

Fig. 7: The CSI reconstruction performance of the testing AI model before and after the 2-stage process.

Please note that the purpose of the 2-stage method is to provide an autonomous approach for identifying and labeling data from unknown environments so that an AI-supported CSI processing model can be enhanced, e.g., through re-training or extensive transfer learning. Therefore, the given AI model should not be used directly for channel estimation after the 2-stage process. As shown in Fig. 7, the performance in terms of CSI reconstruction MSE degrades in all four testing scenarios after the 1-epoch transfer learning.

V. Conclusion and Future works

In this work, a novel approach based on the concept of OSR is proposed to filter out and categorize the unfamiliar CSI data from different environments based on MSE and transfer learning. The approach was validated using synthetically generated MATLAB data pertaining to four different environments. The evaluation results highlight the ability of our approach to assist the model in dealing with unknown data. In future work, we will focus on the integration of the unknown wireless environment detection as well as AI model updates for real-time CSI processing.

Acknowledgment

This project was supported partially by the U.S. National Science Foundation under the grants NSF ECCS-2336234, ECCS-2139508, and ECCS-2139520.

References

[1] N. Hassan, K.-L. A. Yau, and C. Wu, "Edge computing in 5g: A review," *IEEE Access*, vol. 7, pp. 127 276–127 289, 2019.

[2] D. Mercado-Perez, V. Kumar, F. Ye, R. Q. Hu, and Y. Qian, "An evaluation platform for channel estimation in mimo systems," in *NAECON 2023-IEEE National Aerospace and Electronics Conference*. IEEE, 2023, pp. 244–248.

[3] B. Xiang, D. Hu, and J. Wu, "Deep learning-based downlink channel estimation for fdd massive mimo systems," *IEEE Wireless Communications Letters*, vol. 12, no. 4, pp. 699–702, 2023.

[4] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep Learning for Massive MIMO CSI Feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, Oct. 2018, conference Name: IEEE Wireless Communications Letters.

[5] Q. Li, A. Zhang, P. Liu, J. Li, and C. Li, "A Novel CSI Feedback Approach for Massive MIMO Using LSTM-Attention CNN," *IEEE Access*, vol. 8, pp. 7295–7302, 2020, conference Name: IEEE Access.

[6] B. Tolba, M. Elsabrouty, M. G. Abdu-Aguye, H. Gacanin, and H. M. Kasem, "Massive MIMO CSI Feedback Based on Generative Adversarial Network," *IEEE Communications Letters*, vol. 24, no. 12, pp. 2805–2808, Dec. 2020, conference Name: IEEE Communications Letters.

[7] X. Bi, S. Li, C. Yu, and Y. Zhang, "A Novel Approach Using Convolutional Transformer for Massive MIMO CSI Feedback," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 1017–1021, May 2022, conference Name: IEEE Wireless Communications Letters.

[8] Y. Cui, A. Guo, and C. Song, "TransNet: Full Attention Network for CSI Feedback in FDD Massive MIMO System," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 903–907, May 2022, conference Name: IEEE Wireless Communications Letters.

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

[10] Z. Hu, J. Guo, G. Liu, H. Zheng, and J. Xue, "Mrfnet: A deep learning-based csi feedback approach of massive mimo systems," *IEEE Communications Letters*, vol. 25, no. 10, pp. 3310–3314, 2021.

[11] C. Geng, S.-J. Huang, and S. Chen, "Recent Advances in Open Set Recognition: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3614–3631, Oct. 2021, conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[12] P. Oza and V. M. Patel, "C2ae: Class conditioned auto-encoder for open-set recognition," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2302–2311.

[13] J. Naranjo-Alcazar, S. Perez-Castanos, P. Zuccarello, F. Antonacci, and M. Cobos, "Open Set Audio Classification Using Autoencoders Trained on Few Data," *Sensors*, vol. 20, no. 13, p. 3741, Jan. 2020, number: 13 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1424-8220/20/13/3741

[14] Y.-C. Chen, "A tutorial on kernel density estimation and recent advances," *Biostatistics & Epidemiology*, vol. 1, no. 1, pp. 161–187, 2017.

[15] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path Refinement Networks for High-Resolution Semantic Segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Jul. 2017, pp. 5168–5177.

[16] J. Zeng, Z. He, J. Sun, B. Adebisi, H. Gacanin, G. Gui, and F. Adachi, "Deep transfer learning for 5g massive mimo downlink csi feedback," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, 2021, pp. 1–5.