Learn to Adapt to New Environments From Past Experience and Few Pilot Blocks

Ouya Wang, Jiabao Gao, and Geoffrey Ye Li, *Fellow, IEEE*

***Abstract*—In recent years, deep learning has been widely applied in communications and achieved remarkable performance improvement. Most of the existing works are based on data-driven deep learning, which requires a significant amount of training data for the communication model to adapt to new environments and results in huge computing resources for collecting data and retraining the model. In this paper, we will significantly reduce the required amount of training data for new environments by leveraging the learning experience from the known environments. Therefore, we introduce few-shot learning to enable the communication model to generalize to new environments, which is realized by an attention-based method. With the attention network embedded into the deep learning-based communication model, environments with different power delay profiles can be learnt together in the training process, which is called the learning experience. By exploiting the learning experience, the communication model only requires few pilot blocks to perform well in the new environment. Through an example of deep-learning-based channel estimation, we demonstrate that this novel design method achieves better performance than the existing data-driven approach designed for few-shot learning.**

***Index Terms*—Channel estimation, deep learning, few-shot learning, power delay profile, attention mechanism.**

I. Introduction

Deep learning (DL) can address the intricate correlation among variables, especially those that are difficult to accurately describe with mathematical models [1], which allows us to design wireless communication systems without requiring expert knowledge. Therefore, it has been used to develop communication systems and received widespread attention for its effectiveness [2]. For the data-driven method in [3], a deep neural network (DNN) is adopted to replace the channel estimator and the signal detector in the orthogonal frequency division multiplexing (OFDM) receiver. The end-to-end design in [2] and [4] use two DNNs representing the transmitter and receiver, respectively. Recently, the spirit of the end-to-end model has been extended to semantic communication [5]. These DL-based wireless systems have demonstrated impressive performance in the additive white Gaussian noise (AWGN) channel [3] and frequency-selective channels in [2]. However, only single communication environment is considered in [2] and [3]. It is a challenge for the DL-based communication system to be adapted to new environments with different power delay profiles (PDPs) or distortions.

DL has been highly successful in data-intensive applications but is often hampered by a small available training dataset [6]. The DL-based channel estimation (CE) is often purely data-driven. For any particular channel propagation environment, a large number of pilot blocks or labelled data are required for the training, which is usually performed offline.

Using a small dataset to train a DNN with a large number of parameters can easily lead to over-fitting [1]. Few-shot learning (FSL) has been proposed to tackle that issue. Using prior knowledge, it is possible to quickly generalize a model to a new task with only a few available samples [6]. Meta-learning has been the most common framework for FSL in recent years. In [7], common initialization parameters that enable fast training on any channel have been found using the meta-learning approach. From [7], significant training speed improvement and an efficient communication model can be obtained only through one iteration of gradient descent. Transfer learning (TL) can also be a solution for FSL tasks. Existing applications in wireless communications include deep TL-based signal detection for backscatter communication networks [8] and TL-enabled convolutional neural network (CNN) for 5G industrial edge networks in [9]. Accretionary learning [10] is designed to accumulate learned knowledge and acquire new knowledge. The idea of accretionary learning can be applied to achieve the goal of FSL, where knowledge is learned and accumulated independently during offline training. For accretionary learning, few new parameters are trained and combined with learned knowledge to acquire new knowledge online. In [11], an online training system, called SwitchNet, can capture the features of the new propagation environment. In SwitchNet, multiple DNNs are pre-trained, each for a specific propagation environment. Only a small dataset is required to linear combine outputs of those DNNs in online adaption. However, the propagation environments tested for this method are limited in [11]. In addition to meta-learning and accretionary learning that can be potentially used to realize FSL, model-driven methods also require less data for training due to fewer trainable parameters. In [12], the adaptivity of the channel estimator is

Manuscript received 2 September 2022; revised 8 November 2022; accepted 4 December 2022. Date of publication 13 December 2022; date of current version 11 April 2023. The associate editor coordinating the review of this article and approving it for publication was Y. Pei. *(Corresponding author: Ouya Wang.)*

Ouya Wang and Geoffrey Ye Li are with the Department of Electrical and Electronic Engineering, Faculty of Engineering, Imperial College London, SW7 2BX London, U.K. (e-mail: ouya.wang20@imperial.ac.uk; geoffrey.li@imperial.ac.uk).

Jiabao Gao is with the Institute of Information and Communication Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: gao_jiabao@zju.edu.cn).

Digital Object Identifier 10.1109/TCCN.2022.3228536

enhanced by designing a hypernet to generate parameters for the model-driven based wideband mmwave system. However, for the data-driven CE, the parameter set is enormous, which is challenging for the hypernet to generate all parameters.

In this paper, we focus on a data-driven CE system that can be quickly adapted to a new environment. In Section II, we introduce the existing DL-based approach for CE and briefly review the attention mechanism. Section III formulates the problem and introduces the mechanism for attention generators. In Section IV, we present the FSL method. In Section V, we compare our FSL method with other related ones. Section VI concludes the paper and discusses the potential of developing FSL with some other techniques. The main contributions of this paper are as follows:

- We propose to use the attention mechanism for the CE model to realize FSL, where attention networks generate weights for each feature vector under multiple rules for dynamic adjustment. To the best of our knowledge, this is the first work to introduce the attention mechanism in wireless communications to realize FSL.
- We design a task-attention model to enhance the generalization ability for various distributed training data and improve testing performance in the new environment. By using few pilot blocks in the new environment, the task-attention network adds the knowledge of the new environment to feature maps by generating attention vectors in the channel domain.[1]
- We introduce the cross-attention mechanism to find the correlation between the support and the query blocks in the spatial domain, which leads to a higher estimation accuracy in initialization blocks. This is the first work applying the cross-attention mechanism to wireless communication problems.
- The cross-attention model embedded initialization network is proposed to produce the input of the CE backbone. The query block is firstly initialized according to the efficient feature embedding from support blocks and then sent into the CE backbone. We develop a method that takes the most advantage of the support blocks to improve CE for the query block.

II. Related Works

In this section, we briefly review recent advances in DL-based CE and introduce the working mechanism of SwitchNet, which is closely related to our work. Furthermore, the attention mechanism and its applications in wireless communication are described briefly.

A. Deep Learning in CE

DL has emerged as an effective tool for CE in wireless systems. A new DL-based receiver design, ComNet, has been proposed for an OFDM system to deal with frequency-selective fading channels [13], where two cascaded DNNs are utilized for the CE and signal detection (SD), respectively. For ComNet, the estimated channel is used to recover the transmitted data. With the expert knowledge embedded, ComNet is more predictable and explainable than the fully-connected DNN-based receiver in [3], which treats the joint channel estimator and signal detector as a black box. Due to the strong fitting ability of DNN and the end-to-end training mechanism, the SD network can still recover the transmitted signal even if the output of the CE network is far away from the real channel. Therefore, the output of CE can be regarded as a feature representation rather than accurate instantaneous channel coefficients.

A DNN-based approach for channel sensing and downlink hybrid beamforming is proposed in [14], which is generalized for any number of users. The multi-user cascaded CE is formulated as a denoising process in [15]. CNN with the deep residual framework estimates channel coefficients from the noisy pilot-based observations.

Recently, SwitchNet has been proposed to provide a more accurate CE and enable the system to adapt quickly to the new environment. The architecture of the SwitchNet is shown in Figure 1. It consists of CE and SD and is similar to the conventional receiver. In [11], the structure of the CE network is designed for online adaption. The CE network consists of least-square (LS) CE and five CE SubNets, which are all implemented by neural networks. The CE outputs are linearly combined with parameters $\boldsymbol{\alpha}$. CE SubNet 0 performs the basic CE. Multiple CE SubNets are used as compensating networks for CE SubNet 0 to adapt propagating environments in the training set. Each compensation network aims at a specific propagation environment. During online adaption, each compensation network is governed by a trainable parameter α_i that controls its contribution to fit the new environment. Since there are only few trainable variables during testing for the new environment, only a small batch of samples is required for the adaption. Denote $\mathbf{W}_i$ and $\boldsymbol{\theta}_i$ as the multiplicative parameter weights and bias, respectively. The estimated channel can be expressed as,

$$\boldsymbol{h}_{est} = \left(\sum_{i=1}^{M} \alpha_i \boldsymbol{W}_i + \alpha_0 \boldsymbol{I}\right)(\boldsymbol{W}_0 \boldsymbol{h}_{ls} + \boldsymbol{\theta}_0) + \sum_{i=1}^{M} \alpha_i \boldsymbol{\theta}_i, \quad (1)$$

where $\boldsymbol{h}_{ls}$ is channel coefficients calculated through LS. After more accurate channel coefficients $\mathbf{h}_{est}$ are estimated, the signal detection (SD) network is employed to recover the transmitted data. In order to compare the CE performance with our proposed method, only the CE part of SwitchNet will be used sometime. The above is the offline process. The online adaption aims at learning a combination of the CE SubNets using the support blocks, where true channel coefficients are known for the SwitchNet support blocks. The online fine-tuning process is to learn a set of $\boldsymbol{\alpha}$ to minimize the mean-squared error between the true channel coefficients and the output of CE SubNet, $\mathbf{h}_{est}$. Such an online adaption mechanism allows the model to be adapted to different propagation environments and makes the system more robust than conventional DL-based communication systems.

[1] Channel domain mentioned in this paper is a term of the convolutional neural network, which represents a dimension of the feature map, rather than the communication model.

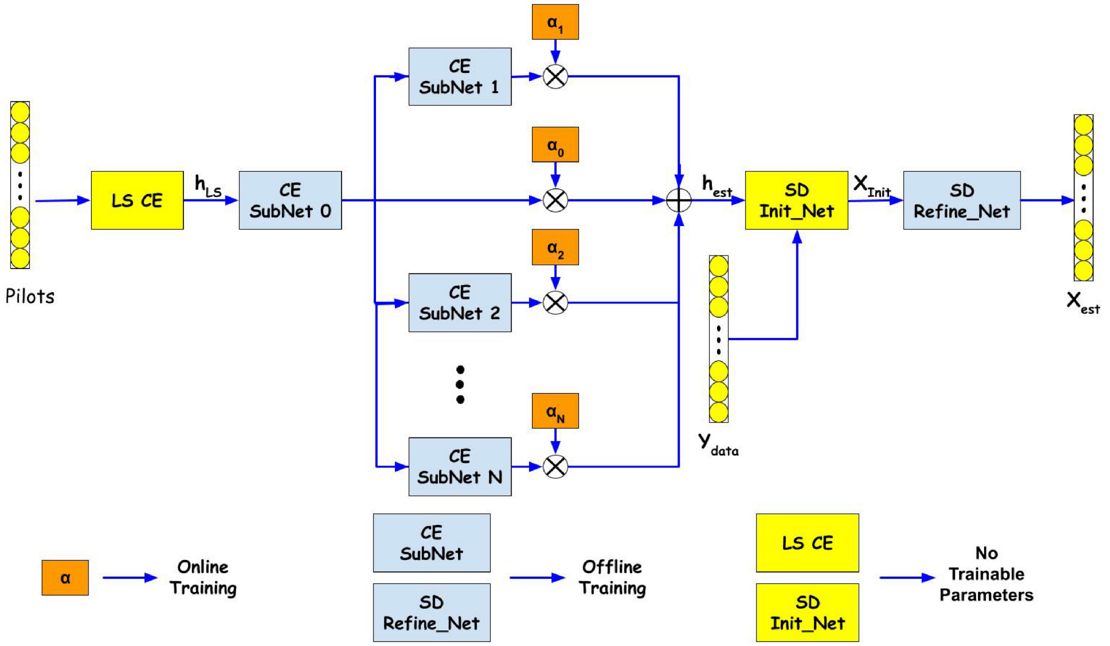


Fig. 1. The structure of the attention mechanism embedded into the end-to-end network. The figure only demonstrates the attention network embedded into the first two convolutional layers for the whole system.

B. Attention Mechanism

Convolutional Neural Networks (CNN) performs classification or regression tasks by generating feature maps for samples. Traditional CNN often treats the extracted features equally, limiting the generalization of a model in some cases. When the category of testing samples is highly separate data, like instantaneous channel coefficients from different environments, the importance of these extracted features is different. For example, the features for coping with channel distribution within a specific angular region might have limited even with negative impacts on estimating channels from another region or environment. Therefore, the attention mechanism is employed to generate weights for each extracted feature vector so that more critical features have more significant contributions [16].

The theory of attention in DL has been first proposed in [17], which adopts recurrent neural network (RNN) and reinforcement learning (RL) to obtain attention in the spatial domain. Then attention mechanism becomes popular with the introduction of squeeze-and-excitation network (SENet) [18] and multi-head attention [19]. SENet is designed to recalibrate channels by using attention weights adaptively. It advances computer vision. Our proposed attention approach is developed based on this theory. We use self-attention to represent the working mechanism of SENet subsequently.

Attention mechanism has been employed to help the DL-based communication system, such as channel state information (CSI) compression [20], channel compression [21] and CE [16], [22]. It can enhance the estimation accuracy for channels with highly separate distributions in [16]. Therefore, it has the potential to improve the learning capacity of the DL model by accumulating more experience learning from datasets with different distributions.

III. Problem Formulation

In this section, we will introduce our FSL problem scenario and provide necessary information about cross attention and task attention in FSL application.

A. Impact of PDPs

The mobile communication channel in our problem scenario is time-varying multipath propagation, which leads to serving dispersion of the transmitted signal. As a multipath propagation channel, the channel impulse response (CIR) at time t can be written as,

$$\boldsymbol{h}(t,\tau)=\sum_{l=1}^{L}A_l(t)e^{-j\phi_l(t)}\delta(\tau-\tau_l), \tag{2}$$

where L is the number of resolvable paths. A_l, ϕ_l and τ_l, represent attenuation, phase shift, and delay in the l^{th} path. The PDP indicates the distribution of transmitted power over various paths in propagation [23]. The channel PDP is calculated from the spatial average of $|\boldsymbol{h}(t,\tau)|^2$ over a local area and represents small-scale multipath channel statistics. The mean power of each multipath component depends on the propagation delay τ_l and is defined by a PDP $P(\tau_l)$ [24]. Furthermore, we employ $\alpha_l(t)$ to describe the fading characteristic of the channel. Therefore, we can use PDP $P(\tau_l)$ and fading function $\alpha_l(t)$ to express the path attenuation.

The received signal from the multipath channel can be written as,

$$y(t)=\sum_{l=1}^{L}\sqrt{P(\tau_l)}\alpha_l(t)e^{-j\phi_l(t)}x(t-\tau_l)+n(t), \tag{3}$$

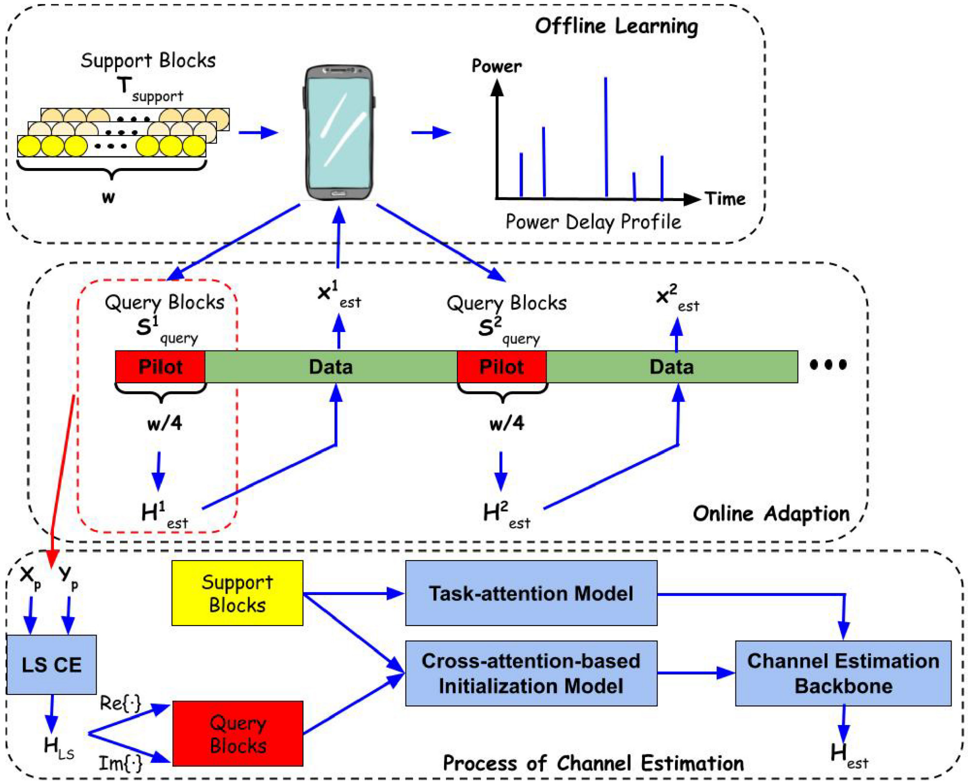


Fig. 2. An overview of how the communication system adapts to the new environment with few pilot blocks.

where $x(t)$ refers to the transmitted signal, $n(t)$ represents AWGN, and $\alpha_l(t)$, with $E|\alpha_l(t)|^2 = 1$, models the time-variant fluctuation of the path attenuation. Channels with the same PDP share the same $\sqrt{P(\tau_l)}$ in path attenuation.

This paper defines the propagation environment as an area shared with the same PDP. We consider high Doppler spread in our simulation, where the coherence time is short and instantaneous channel coefficients vary rapidly. However, instantaneous channels from the same propagation environment share the characteristics of PDP.

B. Problem Scenario

We assume that the channel estimator is designed and trained offline for several propagation environments. Then we investigate how we can use the previous experience together with a few pilot blocks to estimate the channel in a new environment. In the beginning, all symbols in the blocks are pilots in the new environment, which are referred to as support blocks $\mathcal{T}_{support}$. We can learn the features of the new environment well from these pilot blocks, which can be done when mobile devices are on standby. Then the pilot portion in each block is decreased so that transmission becomes more efficient. Such blocks with fewer pilots are called query blocks $\mathcal{S}_{query}$. Due to high Doppler spread, $\mathcal{T}_{support}$ and $\mathcal{S}_{query}$ may differ more significantly compared with slow time-varying channel cases.

As shown in Figure 2, each support block, $\mathcal{T}_{support}$, includes whole w pilots in an OFDM block while the query block $\mathcal{S}_{query}$ only contains $\frac{w}{4}$ pilots. The communication system will learn environmental features offline from $\mathcal{T}_{support}$. In the online adaption, these features are employed to enhance the CE from $\mathcal{S}_{query}$. The communication system can estimate channel $\mathbf{H}^i_{est}$ from the i^{th} query block and then recover the transmitted data $\mathbf{x}^i_{est}$ in the i^{th} data block. The accuracy of CE plays a decisive role in transmitted data detection. Therefore, we develop an FSL approach to quickly adapt the communication system to the new environment. With this FSL approach, we can still achieve good CE results for query blocks in the new environment with the guidance of few support blocks.

Support blocks and query blocks in our problem are different from those in conventional FSL problems. In the conventional setting, the support dataset comes with true labels or values, while the query dataset is unlabelled. In our cases, the support blocks, $\mathcal{T}_{support}$, have no corresponding true values since it is hard to obtain the accurate channel coefficients in real communication systems due to various inherent uncertainties in the wireless channels. We only use channel coefficients estimated by LS from the pilot blocks. Support blocks $\mathcal{T}_{support}$ have more pilots compared with the query blocks, which means the channel estimated from $\mathcal{T}_{support}$ has higher accuracy. Therefore, $\mathcal{T}_{support}$ contains more channel information and more accurate environment features can be extracted compared to that included in $\mathcal{S}_{query}$. Since instantaneous channels from the same propagation environment share many features, the environment features extracted from $\mathcal{T}_{support}$ can be employed to enhance the CE for $\mathcal{S}_{query}$. Since instantaneous channels from the same propagation environment share many features, the environment features extracted from $\mathcal{T}_{support}$ can be employed to enhance the CE for $\mathcal{S}_{query}$.

The symbol-spaced multipath channel is described by complex random variable $\{h(n)\}_{n=0}^{N-1}$. while $x(n)$, $y(n)$, and $\omega(n)$ represent discrete transmitted, received pilot signals and AWGN. N refers to the length of pilot blocks, which equals w in Figure 2. After performing the discrete Fourier transform (DFT), the received pilot signal in the frequency domain can be written as,

$$Y_p(k) = X_p(k)H(k) + W(k),$$

where $Y_p(k)$, $X_p(k)$, $H(k)$ and $W(k)$ are the DFT of $y(n)$, $x(n)$, $h(n)$ and $\omega(n)$, respectively. The support and query blocks are estimated through LS from pilot blocks, which can be described below.

$$\begin{aligned}\mathcal{T}_{support} &= [H_{LS}(0), H_{LS}(1), \ldots, H_{LS}(N-1)]^T \\ &= \left[\frac{Y_p(0)}{X_p(0)}, \frac{Y_p(1)}{X_p(1)}, \ldots, \frac{Y_p(N-1)}{X_p(N-1)}\right]^T \end{aligned} \tag{4}$$

Since $\mathcal{S}_{query}$ only contains $\frac{w}{4}$ pilots, interpolation is then performed to obtain the estimated channel impulse response in all OFDM symbols. Two estimated channel coefficients from adjacent pilot symbols are employed for CE on the data between them.

We demonstrate the process of estimating channel $\mathbf{H}_{est}$ from the query block in Figure 2. First, we use pilots to estimate $\mathbf{H}_{LS}$ through LS. Then we get real-valued vector $\mathcal{S}_{query}$ that consists of the real and imaginary parts of $\mathbf{H}_{LS}$. $\mathcal{S}_{query}$ is sent to the CE network to generate accurate estimated channel $\mathbf{H}_{est}$. The process of CE is as follows: the task-attention model learns the environment features from $\mathcal{T}_{support}$ offline and generates attention vectors to provide dynamic adjustment during the estimation process by the CE backbone. The cross-attention-based initialization model employs $\mathcal{T}_{support}$ to initialize $\mathcal{S}_{query}$ through feature embedding and initialized $\mathcal{S}_{query}$ is the input of the CE backbone. The detailed working mechanism will be introduced subsequently.

The essence of our FSL problem is to use a small number of pilot blocks with more information ($\mathcal{T}_{support}$) to guide blocks with less channel information ($\mathcal{S}_{query}$) for efficient CE in new environments. With only a small number of support blocks, the estimation accuracy of query blocks can be close to the testing accuracy boundary, which is obtained by testing the query blocks with the model trained with sufficient pilot blocks of the same environment. The approach to achieve the goal of FSL is to find out useful features through a small number of support blocks and then use these features to improve the accuracy of the channel estimated from the query block.

The attention mechanism is employed to realize the approach. The task attention uses global environment features extracted from all support blocks to give dynamic adjustment in the process of estimating channels from $\mathcal{S}_{query}$. In comparison, the cross-attention uses local feature correlation between each support and query block to enhance channel feature embedding. For each $\mathcal{S}_{query}$, we will select different channel features from $\mathcal{T}_{support}$ and embed them in $\mathcal{S}_{query}$ to improve CE. This process is called CE guidance. Such guidance depends on the individual pilot block, while the adjustment from task attention is only related to the environment. Both attention approaches have meta-learner to guarantee effectiveness when deployed to new environments. These two attention mechanisms will be introduced subsequently.

C. Cross-Attention

Conventional attention models (e.g., SENet) are not effective for FSL since they usually find out important features of the test samples only based on the prior of the training dataset and result in poor generalization performance to unseen samples. Furthermore, a large number of samples are required to learn the attention mechanism for any specific class. Instead of using each sample's own feature map to draw attention, the cross-attention model (CAM), proposed in [25], employs semantic relevance between support and query set features to generate attention maps in pairs. The query set contains the unlabeled samples from unseen classes, while the support set has few labelled samples from the corresponding classes. With such an approach, only few labelled samples are required to highlight the important regions so that more discriminative features can be extracted.

The CAM is illustrated in Figure 3 (a) in our problem scenario. The class feature map, $\mathbf{P} \in \mathbb{R}^{n\times w\times c}$, is extracted from the support blocks and the testing feature map, $\mathbf{Q} \in \mathbb{R}^{w\times c}$, is extracted from the query block, where c and w refer to the number of channels and the width of the feature map generated from one block, n represents the number of support blocks. For each testing feature map, there are n class feature maps to help extract attention vectors. CAM will produce cross attention weights $\mathbf{A}^p$ ($\mathbf{A}^q$) in the spatial domain for the feature map $\mathbf{P}$ ($\mathbf{Q}$). The testing feature map computes the cross-attention with each class feature map. A total of n pairs of attention maps will be generated. The testing feature map, $\mathbf{Q}$, is duplicated n times in order to weight with different attention maps.

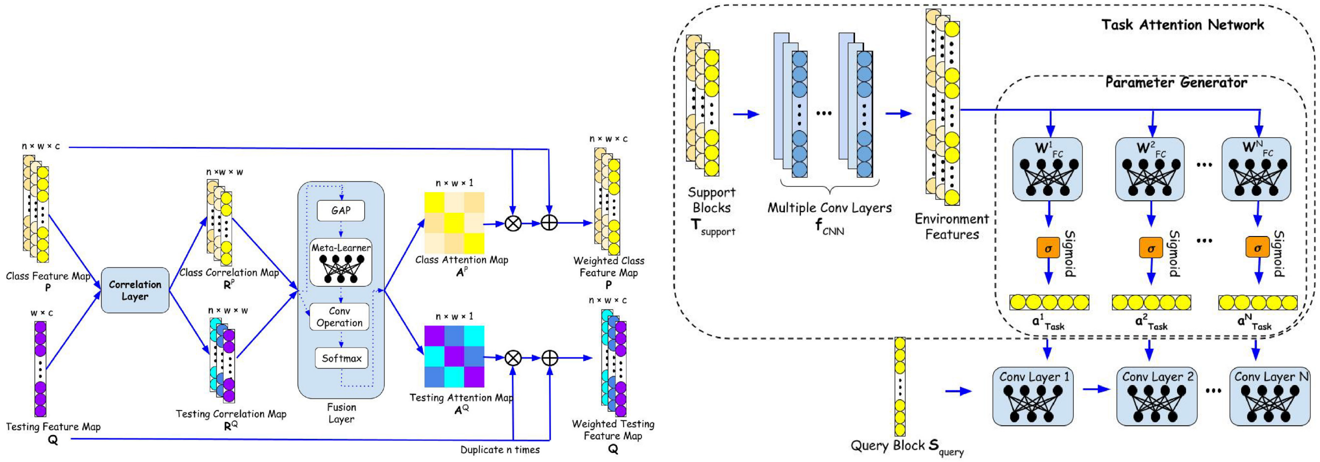
First, the correlation map between feature maps $\mathbf{P}$ and $\mathbf{Q}$ is calculated for the generation of the cross-attention map. Such a correlation map $\mathbf{R}$ refers to the semantic relevance between $\mathbf{p}_i$ and $\mathbf{q}_i$ with cosine distance, where $\mathbf{p}_i$ and $\mathbf{q}_i \in \mathbb{R}^c$ are the i^{th} spatial position in $\mathbf{P}$ and $\mathbf{Q}$, respectively. The correlation layer computes the semantic relevance between $\mathbf{p}_i$ and $\mathbf{q}_j$ to get the correlation map $\boldsymbol{R} \in \mathbb{R}^{nw\times w}$ as

$$\boldsymbol{R}_{ij} = \left(\frac{\boldsymbol{p}_i}{\|\boldsymbol{p}_i\|_2}\right)^T \left(\frac{\boldsymbol{q}_j}{\|\boldsymbol{q}_j\|_2}\right), i = 1, \ldots, nw, j = 1, \ldots, w. \tag{5}$$

There are two correlation maps, $\mathbf{R}^P = [\mathbf{r}_1^p, \mathbf{r}_2^p, \ldots \mathbf{r}_{nw}^p]^T$ and $\mathbf{R}^Q = [\mathbf{r}_1^q, \mathbf{r}_2^q, \ldots \mathbf{r}_w^q]$, where $\mathbf{r}_i^p \in \mathbb{R}^w$ denotes the correlation between the local support block's feature vector $\mathbf{p}_i$ and all query blocks feature vectors $\{\mathbf{q}_j\}_{j=1}^{w}$, and $\mathbf{r}_j^q \in \mathbb{R}^{nw}$ is the correlation between the query block feature vector, $\mathbf{q}_j$, and all support block feature vectors $\{\mathbf{p}_i\}_{i=1}^{nw}$. The correlation vector $\mathbf{r}_i^p$ is formulated as:

$$\boldsymbol{r}_i^p = \boldsymbol{p}_i^T \boldsymbol{Q}^T, i = 1, \ldots, nw. \tag{6}$$

Then, a fusion layer is employed to generate attention maps based on $\mathbf{R}^p$ and $\mathbf{R}^q$. The fusion layer applies global average pooling (GAP) to the correlation map and sends it into a meta-leaner to generate kernel $\mathbf{w}_{kernel}$ in the spatial domain. The convolutional operation is taken between the correlation map



(a) The structure of the cross-attention network

(b) The structure of the task-attention network

Fig. 3. The structure of two attention networks.

$\mathbf{R}^p$ and kernel $\mathbf{w}_{kernel}$, which aims at fusing each local correlation vector $\{\mathbf{r}_i^p\}_{i=1}^{nw}$ of $\mathbf{R}^p$ into an attention value. Finally, softmax is employed to normalize the attention value. The attention value at the i^{th} place can be expressed as

$$\boldsymbol{A}_i^p = \frac{\exp\Big(\mathbf{W}_2(\sigma\mathbf{W}_1(GAP(\mathbf{R}^p)))^T \boldsymbol{r}_i^p/\tau\Big)}{\sum_{j=1}^{nw}\exp\Big(\mathbf{W}_2(\sigma\mathbf{W}_1(GAP(\mathbf{R}^p)))^T \boldsymbol{r}_j^p/\tau\Big)} \tag{7}$$

where $\mathbf{A}_i^p$ represents the attention value at the i^{th} position, τ is the temperature hyperparameter in softmax function, $\mathbf{W}_1$ and $\mathbf{W}_2$ are parameters of meta-learner, and σ represents to the RELU function. The meta-learner generates the kernel, $\mathbf{w}_{kernel}$, that aggregates the correlations between two feature maps $\mathbf{P}$ and $\mathbf{Q}$ in order to draw attention to the target objects. The testing attention map, $\mathbf{A}^q$, can be achieved in a similar way. Finally, we weight the initial feature map, $\mathbf{P}$ and $\mathbf{Q}$, elementwisely by $1 + \mathbf{A}^p$ and $1 + \mathbf{A}^q$ to form a more discriminative feature maps.

D. Task-Attention

The idea of task-attention is borrowed from the task-aware feature embedding network (TAFE-Net) in [26], which is proposed to handle FSL in image classification. TAFE-Net consists of a meta-learner and a prediction network backbone to do classification for unlabelled images. The meta-learner module focuses on extracting feature embedding from few labelled images for the particular task and generating task-specific weights for each layer in the prediction network. TAFE-Net has shown promising results for data efficiency improvement for FSL. In our experiment settings, we estimate channel from query block, $\mathcal{S}_{query}$, in CE backbone, which acts as a prediction model in TAFE-Net. By following the working mechanism of the meta-learner in TAFE-Net, we develop a task-attention model (TAM) to improve the estimation accuracy.

In Figure 3 (b), the TAM employs an extra CNN to extract features of the support blocks. Then the extracted features are sent to a single layer perceptron to generate attention vectors $\boldsymbol{a}_{task}$ for the CE backbone. The 2D convolutional layers are employed for the CNN since the input shape of the TAM is $n \times w \times k$, where k equals two since each complex channel coefficient is split into two real values. The TAM in Figure 3 (b) can be formulated in the following,

$$\boldsymbol{a}_{task}^i = \frac{1}{1 + \exp\big(\boldsymbol{W}_{FC}^i\big(\boldsymbol{f}_{CNN}\big(\mathcal{T}_{support}\big)\big)\big)}, \tag{8}$$

where $\boldsymbol{f}_{CNN}$ represents the multi-convolutional-layer network function, $\boldsymbol{W}_{FC}^i$ is the weights of the parameter generator with single fully-connected (FC) layer. The sigmoid function is utilized to limit the range of the parameters. The output attention vector for the i^{th} backbone layer output is represented as $\boldsymbol{a}_{task}^i$, as shown in Figure 3 (b).

IV. FSL FOR CE

Instead of using a large amount of data, we propose an FSL approach to allow the channel estimator to adapt to the new environment with only few pilot blocks. In this section, we are going to present the attention-based CE method in detail, where the overview of the algorithm and the working mechanism of each attention model will be demonstrated.

A. Structure of FSL-Based CE

As shown in Figure 4, the FSL-based channel estimator consists of three parts: the CE backbone, the CAM-based initialization network, and the TAM. The CE backbone estimates channel coefficients from the query blocks. CNN is known for being good at exploiting correlation in the input data and is widely employed in CE applications in the frequency domain [27] and the angular domain [16]. A multi-layer one-dimensional (1D) CNN is used for the CE network due to the shape of the input query blocks, as shown in Figure 4. We design the initialization network to initialize the query blocks before sending them to the CE backbone. The initialized query blocks, $\mathbf{s}_{initial}$, are used as the input of the CE backbone. The initialization network allows support blocks to provide guidance for the query block and CAM is embedded

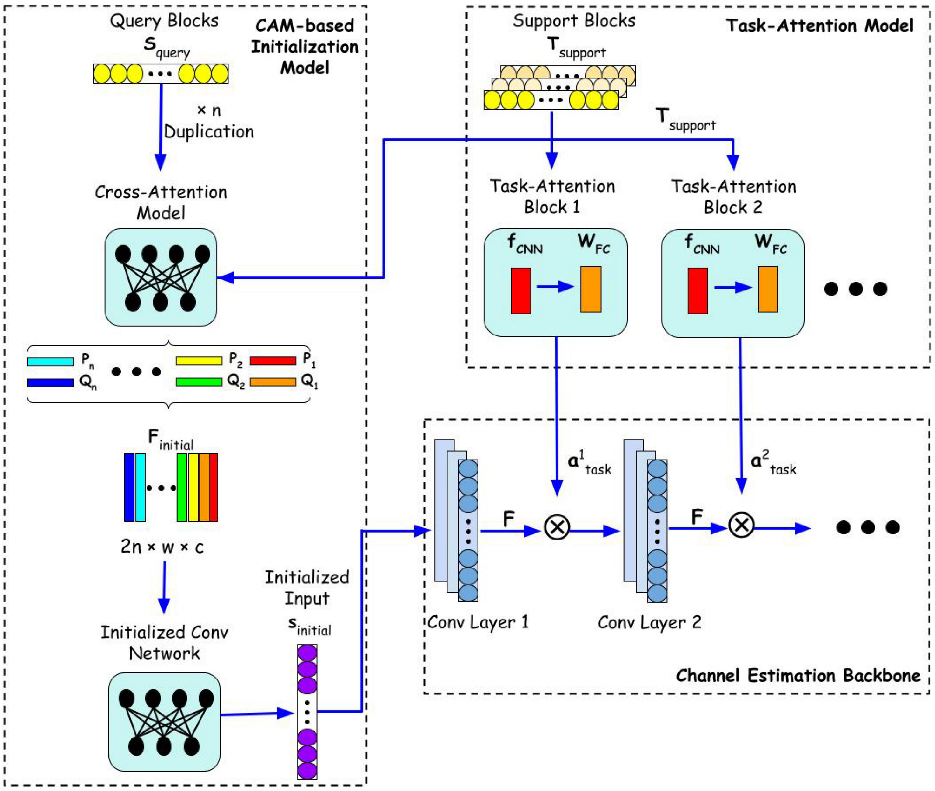


Fig. 4. The structure of the attention mechanism embedded into the CE backbone.

to enhance such guidance. The TAM helps the backbone to improve robustness. It selects important features in the channel domain, while the CAM helps select features for the initialization model in the spatial domain. With the cooperation of the attention mechanism and the initialization model, the CE backbone can be fast adapted to the new environment.

We should emphasize that the above CE backbone is trained offline, corresponding to certain channel environments, such as indoor or outdoor channels. When deploying the CE backbone online, its parameters, θ_{CE}, are fixed. If it is used in a new environment, the mismatch will cause significant performance degradation, which will be addressed through FSL.

The detailed algorithm is demonstrated in Algorithm 1, where $\boldsymbol{f}_{CAM}$, $\boldsymbol{f}_{CIN}$, $\boldsymbol{f}_{TAM}$, and $\boldsymbol{f}_{CE}$ refer to the functions of the CAM, convolutional initialization network (CIN), TAM and CE backbone. From Algorithm 1, the CE backbone has N+2 layers. Except for the last layer, which directly generates the CE result, the output of each other layer is multiplied by the weights, $\boldsymbol{a}_{task}$, generated by TAM. The loss function is considered in two aspects, the output of the CE backbone and the CAM-based initialization network output, both of which derive the error based on the true channel coefficients $\boldsymbol{s}_{true}$.

Algorithm 1 Training the Attention-Based FSL for CE System

```
Require: Learning rate $\eta$
Require: Initial Parameters $\boldsymbol{\theta}_1 = [\boldsymbol{\theta}_{CAM}, \boldsymbol{\theta}_{CIN}]$
Require: Initial Parameters $\boldsymbol{\theta}_2 = [\boldsymbol{\theta}_{TAM}, \boldsymbol{\theta}_{CE}]$
  $\boldsymbol{g} \leftarrow 0$
  $\boldsymbol{\theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2]$
  for $i = 0, 1, \ldots$ do                          ▷ Loops of samples
      Sample a batch of $\mathbf{t}$ from $\mathcal{T}_{support}$
      Sample a batch of $\boldsymbol{s}$ and $\boldsymbol{s}_{true}$ from $\mathcal{S}_{query}$
      $\boldsymbol{F}_{initial} \leftarrow \boldsymbol{f}_{CAM}(\boldsymbol{s}, \boldsymbol{t}, \boldsymbol{\theta}_{CAM})$
      $\boldsymbol{s}_{initial} \leftarrow \boldsymbol{f}_{CIN}(\boldsymbol{F}_{initial}, \boldsymbol{\theta}_{CIN})$
      $\boldsymbol{a}_{task} \leftarrow \boldsymbol{f}_{TAM}(\boldsymbol{t}, \boldsymbol{\theta}^0_{TAM})$
      $\boldsymbol{F} \leftarrow \boldsymbol{f}_{CE}(\boldsymbol{s}_{initial}, \boldsymbol{\theta}^0_{CE})$
      $\boldsymbol{F} \leftarrow \boldsymbol{F} \odot \boldsymbol{a}$
      for $j = 1, 2, \ldots, N$ do ▷ Loops of CE backbone layers
          $\boldsymbol{a}_{task} \leftarrow \boldsymbol{f}_{TAM}(\boldsymbol{t}, \boldsymbol{\theta}^j_{TAM})$
          $\boldsymbol{F} \leftarrow \boldsymbol{f}_{CE}(\boldsymbol{F}, \boldsymbol{\theta}^j_{CE})$
          $\boldsymbol{F} \leftarrow \boldsymbol{F} \odot \boldsymbol{a}_{task}$
      end for
      $\boldsymbol{g} \leftarrow \bigtriangledown\mathcal{L}_{\boldsymbol{\theta}_1}(\boldsymbol{s}_{initial}, \boldsymbol{s}_{true}) +$
            $\bigtriangledown\mathcal{L}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{CE}(\boldsymbol{F}, \boldsymbol{\theta}^{N+1}_{CE}), \boldsymbol{s}_{true})$
      $\boldsymbol{\theta} = \boldsymbol{\theta} - \eta\boldsymbol{g}$
  end for
```

B. TAM-Embedded CE

The support blocks from the same propagation environments are employed as the input of TAM. The generated parameter vector, $\mathbf{a}_{task}$, selects important feature vectors in the channel domain. When encountering a new environment, TAM attempts to predict the PDP based on $\mathcal{T}_{support}$, which is the most representative feature of each environment. The attention vector, $\mathbf{a}_{task}$, is generated based on the predicted PDP. By performing channel-wise multiplication with the original feature map, the backbone can have a better generalization ability to the new environment.

The TAM plays another essential role in multi-environments adaption. Similar to the self-attention mechanism, the TAM-based method ensures the entire deep learning model to become robust to the data with different distributions. The difference is that self-attention generates weights based on local features (e.g., features of the sample itself), while TAM uses global features (e.g., PDP of the environment) to select important features. The original CE backbone cannot converge when the training dataset contains channel coefficients from different environments. However, with TAM embedded, the system can converge to an optimal minimum with data from all environments.

It is challenging to analyze in detail how the TAM learns the weight vectors, $\mathbf{a}_{task}$, to improve the robustness. But we can try to explain the role of the attention mechanism in an implicit way. For the conventional neural network, all data with different distributions are processed by the same weights of the DNN and there is no dynamic adjustment to adapt to a specific distribution. Such a philosophy of DNN design limits the diversity of trainable data distributions. The operation of TAM is similar to the concept of "divide and conquer", where generating weight vectors for feature maps can be treated as a dynamic adjustment. With such an adjustment, the model can be adapted to the training data with various distributions.

Furthermore, the mechanism of this dynamic adjustment can be learnt by the attention network in the training process of adapting to various environments. Compared with TAM, self-attention has poor performance for blocks in the new environment. Because attention is generated based on the feature map of the block itself, self-attention can only find the basic features with the prior of previous training classes and lacks generalization ability to new environments. In contrast, we use TAM to enhance the generalization ability to the new environment. TAM performs as a meta-learner, which aims to learn to make the backbone generalized to various tasks. After being trained by a large number of pilot blocks from different environments, TAM can find out the environment features based on the experience when it encounters pilot blocks of the new environment. The environment-specific attention helps the backbone to be generalized to the new environment.

C. CAM-Based Initialization Network

One of the limitations of the TAM-embedded CE backbone is that the attention generated by TAM is independent of query blocks $\mathcal{S}_{query}$. In other words, for the same propagation environment, different query instantaneous channels with various feature maps will have the same weights if support blocks $\mathcal{T}_{support}$ are the same, which results in attention maps becoming less efficient. Furthermore, TAM cannot introduce extra channel features in the support blocks to the query blocks since TAM only works on re-weighting existing features from the query blocks. Due to more pilots in each support block, we intend to develop a model to allow $\mathcal{S}_{query}$ to learn additional features, which are contained in those positions without pilots. This learning efficiency should be further improved by exploiting the correlation between support blocks $\mathcal{T}_{support}$ and query blocks $\mathcal{S}_{query}$.

As shown in Figure 4, we use a CAM-based initialization network to realize that goal. Both $\mathcal{T}_{support}$ and $\mathcal{S}_{query}$ are sent into a pre-trained feature extractor. Then feature maps are used as input of CAM to generate weighted feature maps in the spatial domain. Each pair of feature maps, $\mathbf{P}_i$ and $\mathbf{Q}_i \in \mathbb{R}^{w \times c}$, are concatenated to form a high-dimensional feature map $\mathbf{F}_{initial}$. There are n pairs of $\mathbf{P}_i$ and $\mathbf{Q}_i$ in total, where n refers to the number of support blocks as mentioned before. The new feature map, $\mathbf{F}_{initial} \in \mathbb{R}^{2n \times w \times c}$, is processed by the CNN. With the convolution operation, the channel features at the positions without any pilots in the query block can be learned with the help of support blocks in $\mathcal{T}_{support}$. The output of the CNN is treated as the initialized query blocks, $\mathbf{s}_{initial}$, and sent into the CE backbone for further processing.

The function of the CAM-based initialization network is to use $\mathcal{T}_{support}$ to guide CE for $\mathcal{S}_{query}$. CAM focuses on the correlation established by cross-attention so that query blocks with less channel information can learn the rich channel information contained in support blocks more efficiently. The meta-learner in CAM contributes to the new environment adaption and its working mechanism is the same as TAM mentioned above. Through dynamic adjustment in the channel domain based on the global feature and information enhancement in the spatial domain based on each pilot block, a better estimation accuracy can be achieved with the cooperation of the CAM initialization network and the TAM.

V. Experiments

In this section, the proposed attention-based CE system is trained and tested under different propagation scenarios. We first introduce the WINNER channel model employed for simulating the propagation effect in different environments. Then we describe the details of the experiment settings. We implement the SwitchNet [11] and test it using the same training and testing set to compare with our proposed attention-based method. We show that the attention-based CE system outperforms SwitchNet. Furthermore, we explore how the CAM-based initialization network and cross-attention mechanism help improve the testing performance. The testing accuracy boundary of $\frac{w}{4}$-pilot case is also considered, where each testing environment has sufficient training data for the CE backbone to learn from instead of only few shots available. With the help of the w-pilot block in $\mathcal{T}_{support}$, our proposed method is able to get closer to the testing accuracy boundary or even exceed it.

A. WINNER Channel Model

All instantaneous channel coefficients in this experiment are generated by the WINNER channel model (WCM) [28], which has been adapted to various mobile communication scenarios from a local area to a wide area. WCM uses spatial and temporal parameters obtained from the measured CIR to characterize different environments. The measured CIR for each propagation environment is analyzed and processed to get the environment-specific parameters [28], which can be used to simulate the propagation effect for the specific environment. There are twelve different propagation scenarios[2] that WCM can emulate and we choose five of them as the training set while another two are used for testing. In the simulation setting, the carrier frequency is 5.25GHz, and both line-of-sight (LOS) and None-line-of-sight (NLOS) are considered. The PDP varies according to different environments, which leads to different lengths of instantaneous channel coefficients. In order to facilitate the subsequent training of the DL-based CE model, we unify the number of all channel coefficients to 72 and use zero padding for those channels shorter than 72. Since channel coefficients are complex numbers, we split each one into two real numbers. Therefore, the number of real coefficients for each channel is 72 × 2.

B. Experiment Settings

The propagation system layout is configured as a 300-by-300 (meters) map. We consider single-input-single-output (SISO) in the system. The training set for five different propagation scenarios with maximum delay are listed: indoor office with a maximum delay of 175 ns, indoor-to-outdoor with a maximum delay of 305 ns, indoor hotspot with a maximum delay of 405 ns, outdoor-to-indoor (urban) macro-cell with a maximum delay of 535 ns, and urban macro-cell with a maximum delay of 615 ns. It is evident that the average power of each channel tape varies for different environments or the same environment but with different transmitters and receivers positions. We generate 500 PDPs for each propagation scenario for the training dataset by changing the propagation conditions and user positions. Each PDP contains 500 different instantaneous channel coefficients. A significant number of PDPs are required for training the meta-learner in TAM and CAM to learn as many environmental features as possible. When new environmental samples appear, the meta-learner can learn new features by combining the learned features, which guarantees the new task adaption of the backbone.

The testing set includes channels in rural macro-cell with a maximum delay of 420 ns and moving networks with a maximum delay of 210 ns. Compared with propagation scenarios in the trainingset, the moving network scenario considers a higher Doppler spread. And the rural macro-cell has a much lower building intensity. Hence, the LOS condition is more likely to exist in the coverage area. Each propagation scenario has five PDPs. The channel output can be formulated as $\mathbf{y} = \mathbf{h} \circledast \mathbf{x} + \mathbf{n}$, where $\mathbf{n}$ and $\mathbf{x}$ are channel noise and channel input, respectively, $\circledast$ refers to the linear convolution, and $\mathbf{h}$ denotes the CIR. The input of the attention-based CE system is the query blocks with $\frac{w}{4}$ pilots and the support blocks contain w pilots. The number of support blocks required for the new environment[3] will be explored in the following part. All experiment results are obtained by taking the same 500 samples of each environment for testing and then averaging.

C. Baseline

The proposed attention-based CE system is compared with two baselines, SwitchNet and testing accuracy boundary. For SwitchNet, $\frac{w}{4}$ pilots are employed as the input for the LS CE block, which has the same number of pilots as the CE backbone. The testing environments described in [11] are limited for the SwitchNet, while part of the online testing is implemented with similar propagation conditions but with different max delays. We explore the ability of SwitchNet to adapt to highly separate channels in this part. Furthermore, we will measure the distance between the test performance of the proposed attention-based system and the testing accuracy boundary. We will see whether the testing performance of our proposed method can approach or even exceed the testing accuracy boundary. We also test the effectiveness of each part of the attention-based system. First, we explore how the CE backbone performs with only TAM embedded. We will see, compared with SwitchNet, whether the TAM can help the CE backbone to perform better in the new environment through the dynamic adjustment from the meta-learner. Then the CAM-based initialization network is added to check whether the system can consistently improve its generalization to the new environment.

D. End-to-End Model Training

The number of training samples for each environment is 1.25×10^6 and the training batch size is 128. The whole attention-based CE system is trained in an end-to-end manner. Each training sample includes two categories of data: a frame of $\frac{w}{4}$ pilots block and multiple frames of w pilots blocks. Such multiple frames of w-pilot blocks refer to support set $\mathcal{T}_{support}$ in FSL sampling, which is randomly sampled from the training set and belongs to the same average PDP. The loss function and the optimizer used for training are binary cross-entropy and Adam. The model is trained with SNR = 20 dB, while the value of SNR varies from 5 dB to 25 dB during testing.

The detailed network layout for our attention-based CE is introduced in Table I. TAM Main processes the support blocks and the attention vectors are generated through TAM Meta. CAM Main is a deep residual network to introduce extra channel features in the support blocks to the query blocks and the cross-attention vectors are generated through CAM Meta. Table II lists the training parameters for the simulation.

E. Experimental Results

Figure 5 demonstrates how the attention mechanism can help the model to improve the CE accuracy in joint training of multiple environments. As mentioned in the previous section, the TAM-based method ensures the DL model to become robust to the data with different distributions. We select two environments from the training set and test the generalization ability under the condition of whether the TAM is applied for CE. Figure 5 shows that the TAM can enhance the adaption for multiple environments, especially in high SNR scenarios.

Figure 7 shows the testing accuracy for the rural macro-cell channel versus the number of support blocks. From Figure 7, when there is only one block in $\mathcal{T}_{support}$, the performance

[2] One propagation scenario contains multiple different propagation environments. One PDP represents one environment. By changing the positions of transceivers and propagation conditions in the same scenario, we can obtain various environments.

[3] All new environments are from the propagation scenario that is not included in the training set.

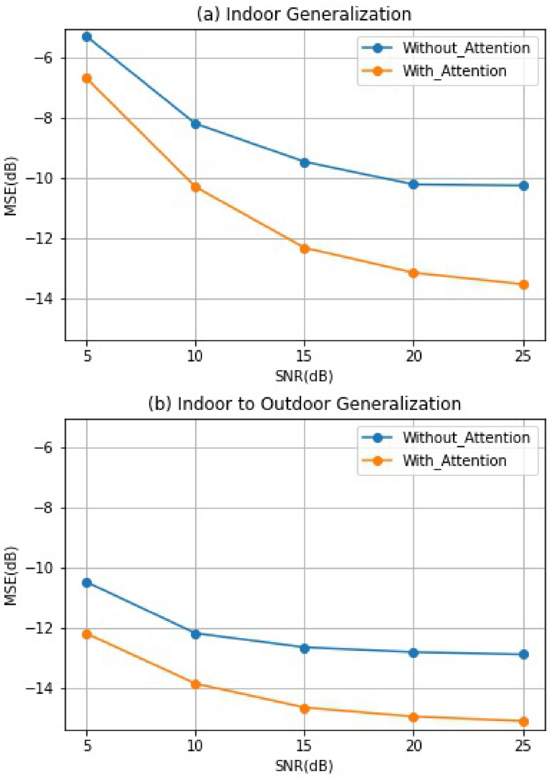


Fig. 5. The MSE between true and estimated channel for known environments.

TABLE I
NETWORK LAYOUTS FOR ATTENTION-BASED CE

	Layer	Output Channel	Activation Functions
CE Backbone	Input	2	None
	Conv	320	Linear
	Conv	320	Linear
	Conv	128	Linear
	Conv	2	Linear
TAM Main	Input	2	None
	Conv	64	Relu
	Conv	32	Relu
	Conv	16	Relu
	Conv	4	Relu
	Flatten	4608	Relu
	FC	128	Relu
TAM Meta	FC	320	Sigmoid
	FC	320	Sigmoid
	FC	128	Sigmoid
CAM Main	Input	32	None
	Conv	128	Relu
	Conv	128	Relu
	Conv	128	Relu
	Conv	64	Relu
	Conv	64	Relu
	Conv	64	Relu
	Conv	32	Linear
	Conv	2	Linear
CAM Meta	Conv	16	Relu
	Conv	72	Linear

TABLE II
TRAINING PARAMETERS FOR SIMULATION

Parameter	Value
Loss Function	MSE
Batch Size	256
Epoch	500
Initial learning rate	0.001
Optimizer	Adam

improvement of CE is the most significant compared with no block since it is easy to recognize some basic channel features from the one-shot pilot block, such as channel taps with high power in the specific environment. As the number of support blocks increases, the performance improves since more implicit features are learnt. The elbow point appears when the number is 16. After the number of blocks exceeds 16, the accuracy improvement is very limited by continuing to increase the blocks. Therefore, the subsequent experimental results are all tested under the condition that the number of blocks contained in $\mathcal{T}_{support}$ is 16.

During training, TAM allows the CE backbone to be adapted to data in various distributions. Without the TAM, the model has a bad generalization performance using training samples from different environments. Figure 6 compares mean-squared error (MSE) for different design methods. From Figure 6, the model's generalization ability to the new distributed data is enhanced using the task-attention mechanism. The TAM can help the CE backbone to achieve a lower MSE for the new environments compared with SwitchNet. The degree of freedom for the dynamic adjustment by TAM is much higher than SwitchNet. The attention network generates multiple parameter vectors while SwitchNet can only rely on the five parameters for the new environment adaption. Furthermore, SwitchNet is trained separately for the datasets with different distributions while the attention-based approach uses joint training of these datasets, which allows the model to learn additional information in joint learning from different environments, such as high-level features for the meta-learner training. Therefore attention-based mechanism outperforms SwitchNet in generalization to the new environment. In addition, SwitchNet requires online training steps to be adaptive to the new environment. Our proposed attention-based method is directly generalized to the new environment without fine-tuning steps for online adaption.

The CAM-based initialization network positively affects the TAM-embedded model's generalization to the new environment from Figure 6. Additional channel information in support blocks can be effectively learned by the query block in the initialization network. Furthermore, We demonstrate that CAM can enhance the system's testing performance. In Figure 6, the FSL model without CAM-embedded is also considered and it has worse performance compared with the CAM-embedded case. The CAM provides attention in different dimensions from TAM and it is proved in [29] that combining channel and spatial attention leads to consistent improvements for CNN-based models. Figure 6 shows that with the initialization network embedded, the CE system becomes more robust for low SNR. For the high SNR scenario, the initialization network can improve the testing accuracy and is close to the boundary in most cases. With the help of CAM, our proposed model can exceed the $\frac{w}{4}$-pilots testing boundary.

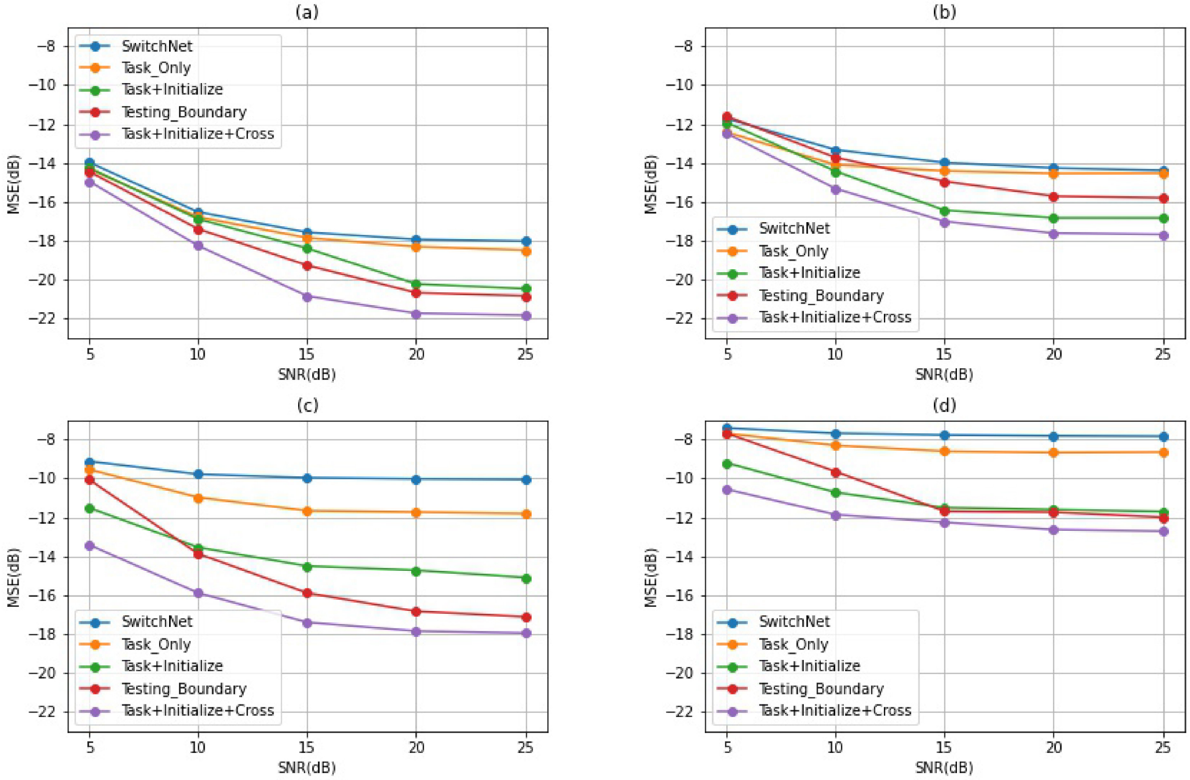


Fig. 6. MSE between true and estimated channels for rural macro-cell at different positions (a) (b), and for rural moving network at different positions (c) (d).

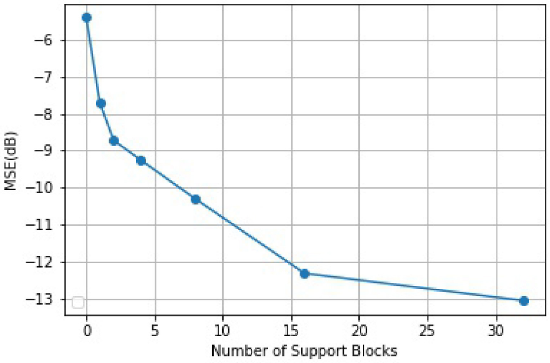


Fig. 7. The MSE between true and estimated rural channel versus the number of support blocks.

We should emphasize that such MSE performance with support blocks is not due to the high similarity between the test environment and environments in the training sets since we carefully set up the experiments to avoid such circumstances. We employed CNN to do classification for all training and testing channel coefficients, whose label is the environment to which the channel belongs. The classification accuracy is over 95%, which indicates that these environments are highly separated and have features that distinguish them from other environments. In addition, we prove that if query blocks $\mathcal{S}_{query}$ and support blocks $\mathcal{T}_{support}$ belong to different environments, which is called 'mismatch'. Two mismatch cases are considered. One case is that these two different environments are obtained from different scenarios. The other is in the same propagation scenario but with different transceiver positions. Both cases lead to varying PDPs for $\mathcal{S}_{query}$ and $\mathcal{T}_{support}$ and result in significant degrading for the testing accuracy. Only the features of the same PDP can give the most accurate CE in this environment, indicating that environmental feature similarities between different PDPs are not high.

Figure 8 demonstrates that mismatch leads to a significant degrading of the testing accuracy. For the same propagation scenarios, PDPs in different positions share more similar features compared with different scenarios. Therefore, the mismatch for different positions has a better performance than in different scenarios. However, since most channel features are different, such as delay profiles and the number of taps, the gap between mismatch and match performance is enormous.

VI. Conclusion and Further Directions

We have realized FSL for new unseen propagation environments in the DL-based data-driven CE model by exploiting an attention-based mechanism. With few pilot blocks sampled from the new environment, the global features of the new environment and correlations between the support and the query blocks can be quickly extracted. Environment-specific and block-specific attention is generated to allow the model to be fast-adapted to new environments. The proposed mechanism outperforms the existing FSL method in the data-driven scenario.

In this article, we have used CE as an example to realize our novel idea: learn to adapt to new environments using

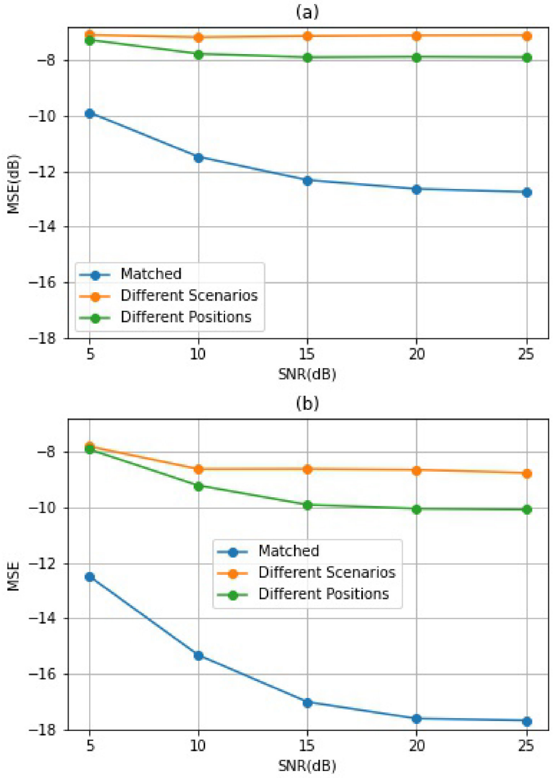


Fig. 8. MSE between true and estimated channels in new environments under match and mismatch cases for rural macro-cell (a) and rural moving network (b).

past experience. The same spirit can be utilized in a large group of communication systems and networks, such as end-to-end systems and signal detection, localization, and resource allocation.

For future research, it is desired to investigate the model-driven DL-based wireless communication model for FSL. The model-driven DL-based approaches combine communication domain knowledge with DL models [30]. Compared with the data-driven method, the model-driven method only contains a small number of parameters that need to be trained, which means the demand for the amount of training data is not so significant. The challenge is to find the most critical parameters affecting performance in the specific environment and figure out which environment features are the most closely associated with these parameters. Therefore, the model-driven approach has great potential to deal with FSL problems.

Another interesting research topic is to apply the graph neural network (GNN) to deal with FSL problems in wireless communications. GNN can be designed to capture the dependence of graphs through information transformation between nodes in the graph. Its working mechanism is equivalent to distributed optimization algorithms. GNN has the potential to perform FSL due to its fewer parameters and high computation efficiency owing to its distributed structure. Furthermore, we can employ GNN to enhance the model-driven method, such as the belief propagation (BP) algorithm. GNN can be used to construct the factor graph and extract features from each iteration in parameter updating through the algorithm so that optimum parameters can be learned.

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[2] H. Ye, G. Y. Li, and B.-H. Juang, "Deep learning based end-to-end wireless communication systems without pilots," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 3, pp. 702–714, Sep. 2021.

[3] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[4] H. Ye, L. Liang, G. Y. Li, and B.-H. Juang, "Deep learning-based end-to-end wireless communication systems with conditional GANs as unknown channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3133–3143, May 2020.

[5] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2663–2675, Apr. 2021.

[6] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, no. 3, pp. 1–34, 2020.

[7] S. Park, O. Simeone, and J. Kang, "Meta-learning to communicate: Fast end-to-end training for fading channels," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process (ICASSP)*, 2020, pp. 5075–5079.

[8] C. Liu, Z. Wei, D. W. K. Ng, J. Yuan, and Y.-C. Liang, "Deep transfer learning for signal detection in ambient backscatter communications," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1624–1638, Mar. 2021.

[9] B. Yang et al., "A joint energy and latency framework for transfer learning over 5G industrial edge networks," *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 531–541, Jan. 2022.

[10] X. Wei, B.-H. F. Juang, O. Wang, S. Zhou, and G. Y. Li, "Accretionary learning with deep neural networks," 2021, *arXiv:2111.10857*.

[11] P. Jiang et al., "AI-aided online adaptive OFDM receiver: Design and experimental results," *IEEE Trans. Wireless Commun.*, vol. 20, no. 11, pp. 7655–7668, Nov. 2021.

[12] W. Jin, H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Adaptive channel estimation based on model-driven deep learning for wideband mmWave systems," 2021, *arXiv:2104.13656*.

[13] X. Gao, S. Jin, C.-K. Wen, and G. Y. Li, "ComNet: Combination of deep learning and expert knowledge in OFDM receivers," *IEEE Wireless Commun. Lett.*, vol. 22, no. 12, pp. 2627–2630, Dec. 2018.

[14] K. M. Attiah, F. Sohrabi, and W. Yu, "Deep learning for channel sensing and hybrid precoding in TDD massive MIMO OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10839–10853, Dec. 2022.

[15] C. Liu, X. Liu, D. W. K. Ng, and J. Yuan, "Deep residual learning for channel estimation in intelligent reflecting surface-assisted multi-user communications," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 898–912, Feb. 2022.

[16] J. Gao, M. Hu, C. Zhong, G. Y. Li, and Z. Zhang, "An attention-aided deep learning framework for massive MIMO channel estimation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 3, pp. 1823–1835, Mar. 2022.

[17] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. Int. Conf. Adv. Neural Inf. Process Syst.*, vol. 27, 2014, pp. 2204–2212.

[18] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.

[19] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Adv. Neural Inf. Process Syst.*, vol. 30, 2017, pp. 6000–6010.

[20] Q. Cai, C. Dong, and K. Niu, "Attention model for massive MIMO CSI compression feedback and recovery," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2019, pp. 1–5.

[21] J. Xu, B. Ai, W. Chen, A. Yang, P. Sun, and M. Rodrigues, "Wireless image transmission using deep source channel coding with attention modules," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2315–2328, Apr. 2022.

[22] D. Luan and J. Thompson, "Attention based neural networks for wireless channel estimation," 2022, *arXiv:2204.13465*.

[23] R. Saha, "Power delay profile and channel classification in cellular mobile communications/ a handbook on cellular mobile communication laboratory a MATLAB-based approach," ResearchGate, Berlin, Germany, Rep. 6, 2016.

[24] T. S. Rappaport et al., *Wireless Communications: Principles and Practice*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[25] R. Hou, H. Chang, B. Ma, S. Shan, and X. Chen, "Cross attention network for few-shot classification," in *Proc. Int. Conf. Adv. Neural Inf. Process Syst.*, vol. 32, 2019, pp. 4003–4014.

[26] X. Wang, F. Yu, R. Wang, T. Darrell, and J. E. Gonzalez, "TAFE-net: Task-aware feature embeddings for low shot learning," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1831–1840.

[27] P. Jiang, C.-K. Wen, S. Jin, and G. Y. Li, "Dual CNN-based channel estimation for MIMO-OFDM systems," *IEEE Trans. Commun.*, vol. 69, no. 9, pp. 5859–5872, Sep. 2021.

[28] Y. D. J. Bultitude and T. Rautiainen, "IST-4-027756 WINNER II D1. 1.2 V1. 2 WINNER II channel models," EBITG, TUI, UOULU, CU/CRC, NOKIA, Espoo, Finland, Rep., 2007.

[29] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[30] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven deep learning for physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 77–83, Oct. 2019.

Ouya (Tracy) Wang received the B.Eng. degree in electrical and electronic engineering from the University of Manchester in 2020, and the M.Sc. degree in applied machine learning from Imperial College London in 2021, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Electronic Engineering. His research interests include accretionary learning and deep learning with application in signal processing.

Jiabao Gao received the B.S. degree in information engineering from Zhejiang University, Hangzhou, China, in 2019, where he is currently pursuing the Ph.D. degree with the Zhejiang Provincial Key Laboratory of Information Processing, Communication and Networking. He is currently a visiting student with the Department of Electrical and Electronic Engineering, Imperial College London, U.K. His current research interests are compressive sensing and deep learning based millimeter wave and THz massive MIMO communications.

Geoffrey Ye Li (Fellow, IEEE) is currently a Chair Professor with Imperial College London, U.K. Before joining Imperial in 2020, he was a Professor with the Georgia Institute of Technology, USA, for 20 years and a Principal Technical Staff Member with AT&T Labs-Research, New Jersey, USA, for five years. His general research interests include statistical signal processing and machine learning for wireless communications. In the related areas, he has published over 600 journal and conference papers in addition to over 40 granted patents and several books. His publications have been cited over 57,000 times with an H-index over 110 and he has been recognized as a Highly Cited Researcher, by Thomson Reuters, almost every year.

Dr. Li was awarded an IEEE Fellow and an IET Fellow for his contributions to signal processing for wireless communications. He won several prestigious awards from IEEE Signal Processing, Vehicular Technology, and Communications Societies, including IEEE ComSoc Edwin Howard Armstrong Achievement Award in 2019. He also received the 2015 Distinguished ECE Faculty Achievement Award from Georgia Tech. He has been involved in editorial activities for over 20 technical journals, including the founding Editor-in-Chief of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS Special Series on ML in Communications and Networking. He has organized and chaired many international conferences, including the Technical Program Vice-Chair of the IEEE ICC'03, and the General Co-Chair of the IEEE GlobalSIP'14, the IEEE VTC'19 Fall, the IEEE SPAWC'20, and the IEEE VTC'22 Fall.