Towards Scalable Automatic Modulation Classification via Meta-Learning

Jungik Jang[†], Jisung Pyo[†], Young-Il Yoon[‡], Sang Yong Seo[‡], Eun Jae Lee[‡], Gyeong Hun Jung[‡], Jaehyuk Choi[†]
[†]School of Computing, Gachon University, Seongnam-si 13120, Republic of Korea
E-mail: {jji4449, p990301, jchoi}@gachon.ac.kr
[‡]Research and Development Center, LIG Nex1, Seongnam-si 13488, Republic of Korea
E-mail: {yougil.yoon, champ.seo, eunjae.lee, gyeonghun.jung}@lignex1.com

MILCOM 2023 - 2023 IEEE Military Communications Conference (MILCOM) | 979-8-3503-2181-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/MILCOM58377.2023.10356268

***Abstract*—Driven by recent technological breakthroughs in deep learning (DL), many recent automatic modulation classification (AMC) methods utilize deep networks to classify the type of modulation in the incoming signal at the receiver. However, existing DL-based approaches suffer from limited scalability, especially for unseen modulations or input signals from new environments not used in training the DL model, thus not ready for real-world systems such as software defined radio devices. In this paper, we introduce a scalable AMC scheme that provides flexibility for new modulations and adaptability to input signals with diverse configurations. We propose a meta-learning framework based on few-shot learning (FSL) to acquire general knowledge and a learning method for AMC tasks. This approach allows the model to recognize new unseen modulations by learning with only a very small number of samples, without requiring the entire model to be retrained. Additionally, we enhance the scalability of the classifier by leveraging a transformer-based encoder, enabling efficient processing of input signals with varying configurations. Extensive evaluations demonstrate that the proposed AMC method outperforms existing techniques across all signal-to-noise ratios (SNRs) on RadioML2018.01A dataset.**

***Index Terms*—Automatic modulation classification, few-shot learning, meta-learning, Transformer, unseen dataset.**

I. Introduction

Accurate classification of modulation types in received signals is a key element of the wireless communication system. Radio signal recognition and automatic modulation classification (AMC) techniques play a crucial role in recognizing modulation types for a wide range of military and civilian services, including dynamic spectrum access, jamming detection, surveillance, and spectrum coexistence.

However, the design of a highly accurate AMC scheme is challenging in the modern wireless communication environment since various heterogeneous communication systems coexist in a complex and non-cooperative manner. It is even more challenging to perform the AMC task in cognitive radio (CR) networks and Software Defined Radio (SDR) systems, which offer the flexibility to utilize various wireless communication services across a broad frequency range. In CR and SDR environments, dynamic spectrum sensing and access is performed over a wide frequency band in a non-cooperative manner. This often leads to unreliable reception and incomplete reception of signals. It is important to note that the AMC in these environments should be able to identify modulation types even when the received samples do not contain the entire packet information and may only have partial information in the middle or tail [1].

Existing DL-based AMC methods are not yet suitable for real-world deployment due to the limited scalability, especially for unseen modulations or input signals with different configurations not seen during training. In non-cooperative and complex real-world communication environments, the received inputs often differ from the features used in the model training phase, leading to significant classification errors. Note that the performance of most DL-based AMC methods heavily relies on a large amount of training data. For instance, most DL-based AMC methods use fixed frame lengths as inputs to models and do not consider the scenarios with variable input sizes [2]. Fig. 1 shows the classification accuracy of ResNet-based and CNN-based methods [2], [3] for different input frame lengths. We can observe that the shorter the length of the input frame, the worse the classification performance where models were trained only with fixed length frames of 1024 samples. In particular, it is noteworthy that both models exhibit an accuracy drop below 80% when the length is 256. Unfortunately, it is nearly impossible to collect sufficient labeled training datasets in advance for numerous combinations of the target classes, such as different frame length, and different signal-to-noise ratios (SNRs), to sustain classification accuracy. Besides, when introducing unseen modulation, prior solutions have to collect a large number of samples and re-train the model. Therefore, it is essential to develop a more intelligent and scalable AMC technique that can adapt to new unseen modulations and recognize input signals with complex combinations of temporal and spatial features.

In this paper, we present a scalable AMC scheme that offers flexibility for new unseen modulations and adaptability to input signals with varying configurations. Our proposed framework is built upon two key components: (i) a meta-learning framework utilizing few-shot learning (FSL), and (ii) a feature extractor based on a Transformer architecture [5]. In our proposed meta-learning framework, we first train the base encoder on a source dataset for a given set of modulations and

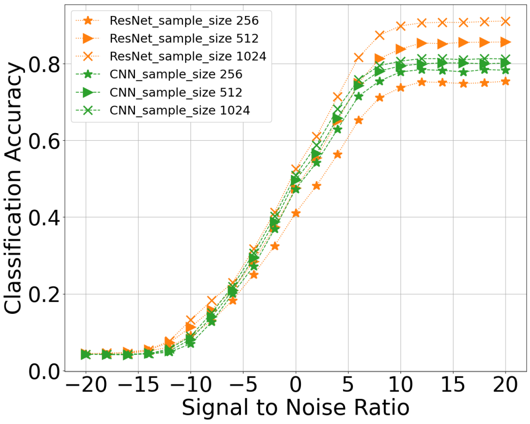

Fig. 1. Impact of input frames' lengths on the classification accuracy of ResNet-based and CNN-based method [2], [3] under different SNR values based on the RadioML2018.01A [4].

then adapt the trained encoder to the new target modulations using only a small number of newly collected samples. This approach effectively resolves the challenges associated with data collection and re-training overhead when dealing with new unseen modulations. Furthermore, to reduce the training overhead of the feature extractor for input signals with diverse configurations, we leverage a Transformer-based encoder [5] in designing the feature extractor for the proposed AMC method. Through extensive evaluations on the RadioML2018.01A dataset [4], we demonstrate that the proposed method consistently outperforms existing techniques.

II. Proposed Method

In this section, we first present the proposed meta-learning framework for AMC task and then explain its details including meta-training and meta-testing processes.

A. Overview

Fig. 2 illustrates the architecture of our proposed meta-learning system. Our system consists of two main modules: a meta-learning module and a meta-testing module. The meta-learning module uses source datasets for given modulation classes (we will denote these modulations as *seen modulations*) and trains the modulation classifier, in particular Transformer-based encoder f_θ, where θ represents the trainable parameters. Unlike traditional supervised learning methods, our meta-learning approach acquires meta-knowledge that facilitates faster learning on new tasks even with limited samples (Section II-B). During the meta-training phase, the encoder f_θ learns general meta-knowledge to extract appropriate feature vectors for AMC tasks, where meta-knowledge represents the underlying essence or commonality among multiple tasks [6]. To do this, we employ the methodology of learning the metric space using prototypes of each class introduced in ProtoNet [7]. Once trained, meta-testing module uses the encoder f_θ for new unseen modulations with fewer collected samples (Section II-C).

Fig. 3 illustrates the architecture of the encoder and its operational sequence. To enhance the scalability of the model with respect to the dataset size, we employed the Transformer-based feature extractor proposed by Dosovitskiy et al. in [5]. This approach induces performance improvement of the model through simple hyperparameter manipulations when dealing with extensive datasets. Each module has an input layer of $2 \times N$ size that takes IQ components of a signal data as input. The IQ components are divided into fixed-size patches, and each patch is linearly embedded after adding position information embeddings.

TABLE I
Details of Proposed Model Hyperparameters

Layers	Hidden size dim	MLP size	Heads	Patch size
8	36	32	9	2x16

Table I summarizes the hyperparameters applied to the encoder of proposed system. The hyperparameters were determined through empirical experiments to achieve optimal performance.

B. Meta-Training

Meta-training module trains the base encoder f_θ with meta-knowledge for the ACM task. Learning at this phase proceeds on an episodic basis. Each episode ϵ is composed of (i) a support set (training set) for prototype generation and (ii) a query set (validation set) for modulation prediction and parameter updating. To generate the support set and query set for each episode, we randomly choose k categories from the source dataset and randomly select n instances from each category. Here, k represents the total number of classes in the support set, which is also referred to as k-way, and n represents the number of data samples for each class (way), known as n-shot. Here, the classes of the modulation data used in training are considered as seen modulations. The N annotated data used as input, denoted as $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, have a frame size of $1 \times 2 \times 1024$ ($C \times H \times W$), which is provided in RadioML2018.01A [4] dataset. The corresponding class labels are represented as $y_i = \{1, \ldots, K\}$.

Within a single episode, the support set and query set data are divided and tokenized at the patch level, as mentioned earlier, and fed into the encoder. For the support set, the prototype c_l is generated using the average of the extracted feature vectors (embedded support points) from the annotated data set S_l belonging to class l.

$$c_l = \frac{1}{|S_l|} \sum_{(x_i, y_i) \in S_l} f_\theta(x_i) \tag{1}$$

The feature vectors extracted from the query set are classified using the generated prototypes based on a distance function d, which can be a method like Euclidean distance or cosine similarity. In ProtoNet [7], Euclidean distance was initially used and demonstrated excellent performance. Therefore, we

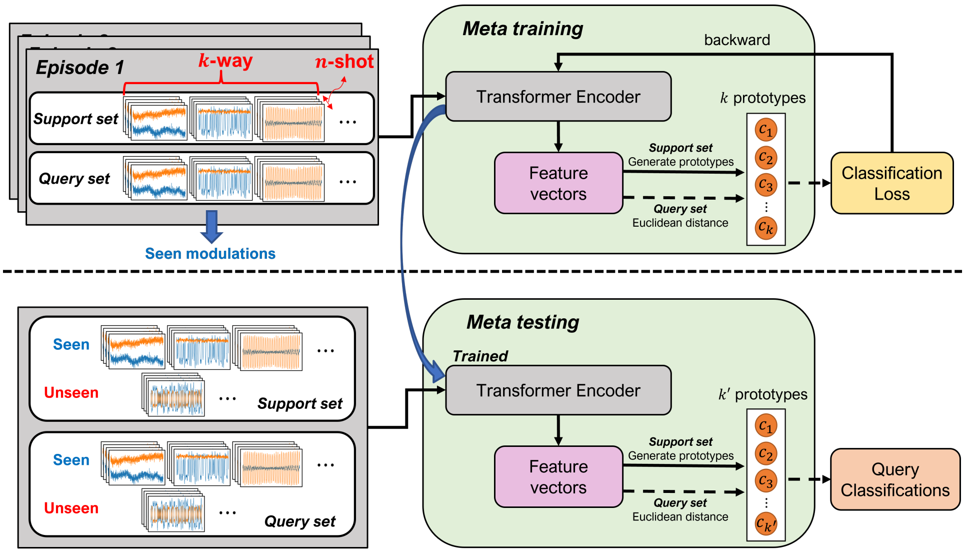

Fig. 2. Overview of our meta-learning system for AMC: meta-training with source datasets for given target modulations and meta-testing process with unseen modulations limited size datasets

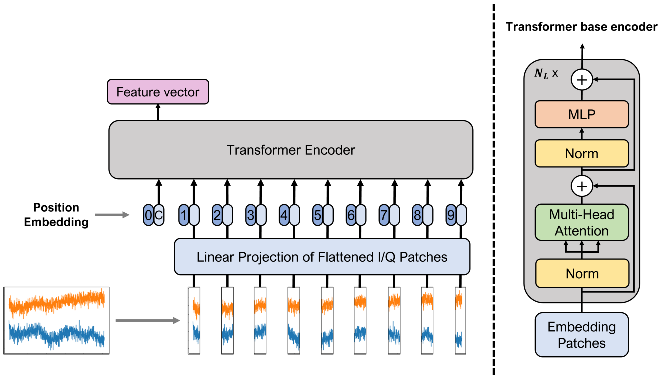

Fig. 3. Transformer-based encoder f_θ to extract feature vectors of I/Q signals. We employed ViT [5]'s encoder structure.

chose to use Euclidean distance as the distance metric. Based on softmax over the distances between the query point x and the prototypes in the embedding space, we generate a distribution over classes. The equation for this distribution is as follows:

$$p_\theta(y = l|x) = \frac{exp(-d(f_\theta(x), c_l))}{\sum_{l'} exp(-d(f_\theta(x), c_{l'}))} \tag{2}$$

As each episode progresses, the parameters θ of f_θ are updated using the Adam optimizer [8] to minimize the negative log probability of the actual class k, as defined in Eq. II-B. L(θ) $= -logp_\theta(y = k|x)$ The algorithm 1 illustrates the meta-training process for an episode.

C. Meta-Testing

Meta-testing module utilizes the base encoder f_θ trained through the meta-training phase. In the meta-testing (or testing) phase, both the support set and the query set consist of unseen modulations, allowing us to evaluate the model's adaptation to a new domain and assess its generalization capability. Note that the query set in the meta-testing phase is no longer used to adjust the parameters θ and is only used for testing [6].

As we will discuss in Section III, we consider applying our method to SDR platform scenarios. For example, an operational SDR equipment needs to be upgraded in order to recognize new modulations that are not included in the

Algorithm 1 Process of meta training. $k \le K$ is the number of classes per episode, E is the selected k classes for episode, N_S is the number of support sample per class, N_Q is the number of query sample per class, $\hat{m}$ is the bias-corrected moving average of the gradients, $\hat{v}$ is the bias-corrected moving average of the squared gradients, α is the learning rate and ϵ is a small value used for numerical stability. RANDOM UNIFORM(S, N) denotes uniform and random selection of N values from the S set.

```
Input: Training set $S_{train} = \{(x_1, y_1), ..., (x_N, y_N)\}$
Output: Trained base encoder $f_\theta$
for $l$ in {1,...,k} do
    $S_{support} \leftarrow$ RANDOM UNIFORM$(S_{E_l}, N_S)$
    $S_{query} \leftarrow$ RANDOM UNIFORM$(S_{E_l} \backslash S_{support}, N_Q)$
    $c_l \leftarrow \frac{1}{N_S} \sum_{(x_i,y_i) \in S_l} f_\theta(x_i)$
end for
$L \leftarrow 0$ {Initialize loss $L$}
for $l$ in {1,...,k} do
    for $(x, y)$ in $S_{query}$ do
        $\theta \leftarrow \theta - \frac{\alpha}{\sqrt{\hat{v}}+\epsilon}\hat{m}$
    end for
end for
```

current model's training. For this purpose, the meta-testing module can include the datasets for seen modulations used in the training phase, as well as new unseen modulations. The results of these tests can be found in Section III-C. A commonly used setting for support sets in most FSL-based approaches is 5-shot, meaning that the support set consists of five data samples. Similar to the meta-trainig phase, meta-testing module generates k' prototypes using the trained f_θ, where k' denotes the number of target classes for meta-testing. The query set used for inference is classified based on the Eudclidean distance between the embedding vectors and the prototypes. In our experiments, we investigated the impact of the k' value, the results of which can also be found in section III-C.

III. Performance Evaluation

In this section, we evaluate the performance of our proposed system through a series of extensive experiments. These experiments include comparing meta-learning and supervised learning approaches (Section III-B), evaluating the few-shot learning capability of our method on new *Unseen* modulations (Section III-C), and examining the scalability of our method for different input data sizes (Section III-D).

The training dataset ratio p_{train} is set to 0.8, and N_{epoch} is set to 50. The optimizer used is Adam [8], with an initial learning rate α of 0.001. A scheduler with a step size of 10 and γ of 0.9 is employed. The experiments were conducted on an Ubuntu 20.04 system with an Intel(R) i9-9900KF processor and GeForce RTX 2080 Ti 11GB GPU.

A. Dataset

We conducted our experiments using the RadioML2018.01A [4] dataset widely in the field of AMC research. This dataset comprises a total of 24 modulations, including analog modulations such as AM-DSB-WC, AM-DSB-SC, AM-SSB-WC, AM-SSB-SC, FM, and digital modulations such as OOK, 4ASK, 8ASK, BPSK, QPSK, 8PSK, 16PSK, 32PSK, 16APSK, 32APSK, 64APSK, 128APSK, 16QAM, 32QAM, 64QAM, 128QAM, 256QAM, GMSK, and OQPSK. Each frame consists 1024 samples for the IQ components. The dataset consist of 4096 frames for each modulation-SNR combination, resulting in a total of 2.5 million frames. The SNR range spans from -20 dB to 30 dB with a step size of 2 dB.

B. Comparing Meta-Learning and Supervised Learning

First, we conducted an evaluation to compare the classification accuracy of meta-learning models including our method and supervised learning models for the 24 modulations. The supervised learning models used in the experiments include ResNet [4] based and CNN [2] based models, where we will denote them as ResNet and CNN, respectively. For the meta-learning models, we employed ProtoNet [7] along with our proposed model. These four models were trained using four different SNR ranges: [0, 20] dB, [0, 10] dB, [-10, 20] dB, and [-20, 20] dB. Their performance was evaluated in terms of accuracy, with a step size of 2 dB, across the entire range of [-20 to 20] dB. This experiment is designed to assess the degree of dependency on training data and evaluate scalability. We conducted training using partial data instead of the entire dataset and evaluated the performance within the [-20, 20] dB range.

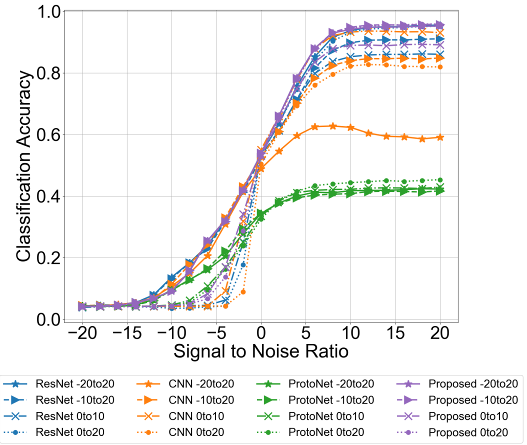

Fig. 4. Performance comparison between meta-learning (our proposed method and [7]) and supervised learning (ResNet [4] and CNN [2]) models for all 24 modulations.

Fig. 4 shows the results of the evaluation. We observed significant variations in the performance of the ResNet and CNN models depending on the size of the training data. In particular, it is noteworthy that both models performed worse when trained with training data set containing the low SNR ranges of [-20, -10] dB. The CNN model's performance exhibited a significant drop in this scenario.

In contrast, the meta-learning-based ProtoNet and our proposed technique were relatively unaffected by the size of the training data. However, ProtoNet performed better only in the high SNR range and had lower accuracy overall. On the other hand, the proposed model maintained high overall performance regardless of the range of training data, and achieved the highest accuracy of 95.76% when trained with partial dataset in the SNR range of [-10, 20] dB. The meta-learning approach demonstrated high performance and high scalability in AMC classification compared to traditional supervised learning methods.

We also compared the complexity of the four models in Table II. Despite its higher computational complexity compared to the other three models, the proposed model demonstrated the best performance across all 24 modulations.

C. Unseen Modulation

Next, we evaluate the adaptation performance of our proposed method to new modulation types. As mentioned earlier, one of the advantages of meta-learning is its ability to quickly adapt the model to new unseen classes. We conducted experiments where the proposed model was trained on 12

TABLE II
COMPLEXITY COMPARISON OF DIFFERENT MODELS

Model	FLOPs	Memory	Speed	Params
ResNet [4]	**0.026G**	**5.32MB**	0.004s	0.17M
CNN [2]	0.038G	16.69MB	0.005s	0.04M
ProtoNet [7]	0.014G	7.06MB	**0.002s**	**0.02M**
Proposed	0.046G	15.98MB	0.005s	0.72M

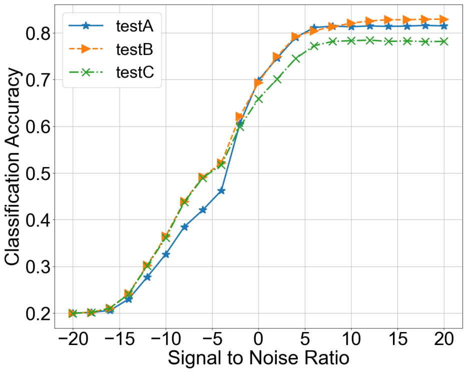

Fig. 5. Performance comparison for three test cases in Table III.

randomly chosen modulations (denoted as *Seen* modulations) out of the total 24 modulations. We then randomly selected 5 modulations out of the remaining 12 modulations as *Unseen* modulations for testing. We divided the test cases into three categories as indicated in Table III. For each test case, we conducted 100 test iterations (i.e., selecting five *Unseen* modulations randomly for each iteration) to calculate the average accuracy. The default value for ”shot” was set to 5-shots.

TABLE III
12 MODULATIONS USED FOR TRAINING BY TEST CASE

Test Case	Modulations
A	’8ASK’, ’BPSK’, ’32PSK’, ’16APSK’, ’64APSK’, ’128APSK’, ’128QAM’, ’AM-SSB-WC’, ’AM-SSB-SC’, ’AM-DSB-SC’, ’GMSK’, ’OQPSK’
B	‘BPSK’, ‘8PSK’, ’32PSK’, ’32APSK’, ’64APSK’, ’128APSK’, ’64QAM’, ’AM-SSB-WC’, ’AM-DSB-WC’, ’FM’, ’GMSK’
C	’8ASK’, ’BPSK’, ’QPSK’, ’16PSK’, ’32PSK’, ’32APSK’, ’32QAM’, ’128QAM’, ’AM-SSB-WC’, ’AM-DSB-WC’, ’FM’, ’GMSK’

Fig. 5 depicts the accuracy results for the three test cases, illustrating an average accuracy of approximately 80% in the high SNR region for the five randomly selected *Unseen* modulations. The variation in accuracy among the test cases is influenced by the complexity of the modulations employed during the training phase, as more complex modulations tend to exhibit better performance during the inference phase. For the subsequent experiments, we used the Test B category in Table III.

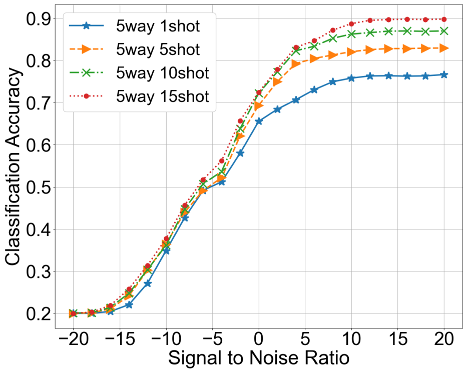

Fig. 6. Impact of number of shots on classification performance for 5-way (five *Unseen* modulations/classes) with 1, 5, 10, and 15 different shots.

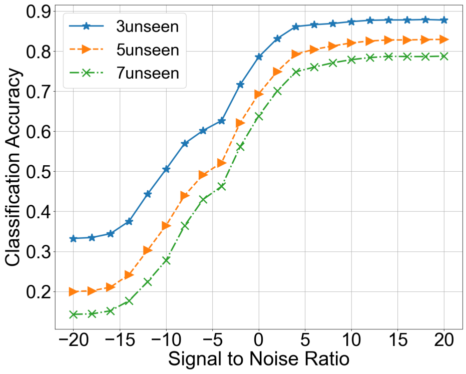

Fig. 7. Performance evaluation for different number of ways, i.e., 3, 5, and 7 *Unseen* modulations, with a fixed 5-shot learning.

Fig. 6 shows the results of an experiment that examined the impact of shots on each class (way) of the Support Set during the meta-testing phase. For the 5-way classification, we used the $\{1, 5, 10, 15\}$ shots. The results indicate that that the accuracy increases as the number of shots increases. With 15 shots, our method achieved 90% accuracy on the *Unseen* modulations, demonstrating its ability to quickly acquire general knowledge about a new domain even with a few dataset.

Fig. 7 shows the classification performance results for three different numbers of *Unseen* modulations, using a fixed 5-shot. We conducted tests with *Unseen* modulations consisting of 3, 5, and 7 classes. The results indicate that with a decrease in the number of *Unseen* modulations, there is an improvement in the differentiation of prototypes within the embedding space, resulting in higher performance. Particularly, a significant increase in classification accuracy is observed for lower SNRs as the number of *Unseen* modulations decreases.

Fig. 8 represents an experiment tailored for the SDR platform scenarios. It evaluates the performance by incorporating both *Seen* modulations used in the training phase and *Unseen* modulations. Despite the inherent challenges posed by the 12-way and 13-way configurations in the context of meta-learning, our method achieved a high performance level,

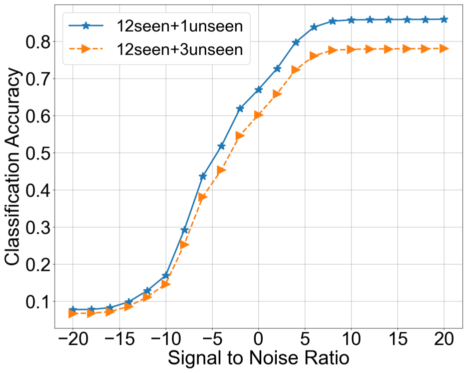

Fig. 8. Performance evaluation using both the 12 *Seen* modulations used during training and additional *Unseen* modulations in the test phase.

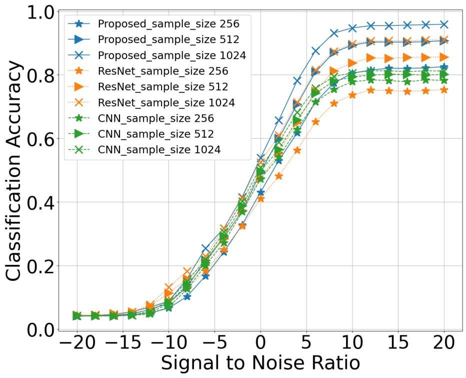

Fig. 9. The impact of varying input frame lengths on the classification accuracy in the RadioML2018.01A dataset [4].

surpassing 80% accuracy with 5-shot learning. We expect that performance can be further improved by investigating the hyperparameters and by leveraging more powerful computing environments. These possibilities will be explored further in our future work.

D. Input Size Scalability

In real-world scenarios, modulation classification may be required for signals with incomplete reception or varying lengths. In many existing AMC methods, however, the input frame size was often neglected in both model design and evaluation. Fig. 1 clearly shows that previous studies that achieved high performance are not suitable for such scenarios. Therefore, we conducted additional experiments to assess the scalability of our proposed model for frame lengths smaller than the given 2×1024 frames. Specifically, we experimented with frame lengths of $2 \times \{256, 512, 1024\}$ to evaluate the model's performance and generalizability.

Fig. 9 presents the evaluation results of the proposed model using smaller input frames compared to the existing models. All models were trained using 2×1024 frames. Thanks to the self-attention mechanism of the Transformer, the proposed model effectively captures the interactions between sample patches within the same frame. Consequently, achieving a performance of 80% or higher even with a smaller input frame size, surpassing the performance of other models evaluated with 2×1024 frames.

IV. Conclusion

In this work, we presented a scalable Automatic Modulation Classification (AMC) scheme that offers flexibility for handling new modulations and adaptability to diverse input signal configurations. By leveraging a meta-learning framework based on few-shot learning (FSL), we enabled the model to acquire general knowledge and efficiently recognize new unseen modulations using a small number of samples, without the need for complete retraining. Furthermore, we enhanced the scalability of the classifier by leveraging a Transformer-based encoder, enabling effective processing of signals with varying configurations. Through extensive evaluations on the widely used RadioML2018.01A dataset, we demonstrated the effectiveness of our proposed AMC method over existing techniques in all SNR ranges.

For future research, we will study on optimizing hyperparameters, such as number of ways and shots, in our proposed meta-learning technique. Additionally, we plan to explore the applicability of this technique in other domains and fields.

Acknowledgement

This work was supported by the Korea Research Institute for Defense Technology Planning and Advancement (KRIT) Grant funded by the Defence Acquisition Program Administration (DAPA) under Grant KRIT-CT-22-002.

References

[1] Shilian Zheng, Peihan Qi, Shichuan Chen, and Xiaoniu Yang. Fusion methods for cnn-based automatic modulation classification. *IEEE Access*, 7:66496–66504, 2019.

[2] Seung-Hwan Kim, Jae-Woo Kim, Williams-Paul Nwadiugwu, and Dong-Seong Kim. Deep learning-based robust automatic modulation classification for cognitive radio networks. *IEEE access*, 9:92386–92393, 2021.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Timothy James O'Shea, Tamoghna Roy, and T Charles Clancy. Over-the-air deep learning based radio signal classification. *IEEE Journal of Selected Topics in Signal Processing*, 12(1):168–179, 2018.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[6] Xiaoyang Hao, Zhixi Feng, Shuyuan Yang, Min Wang, and Licheng Jiao. Automatic modulation classification via meta-learning. *IEEE Internet of Things Journal*, 2023.

[7] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.