IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 December 2023, accepted 5 January 2024, date of publication 11 January 2024, date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3352634

RESEARCH ARTICLE

Meta-Transformer: A Meta-Learning Framework for Scalable Automatic Modulation Classification

JUNGIK JANG[1], (Student Member, IEEE), JISUNG PYO[1], YOUNG-IL YOON[2], AND JAEHYUK CHOI[1], (Member, IEEE)

[1]School of Computing, Gachon University, Seongnam-si 13120, Republic of Korea
[2]Research and Development Center, LIG Nex1, Seongnam 13488, Republic of Korea

Corresponding author: Jaehyuk Choi (jchoi@gachon.ac.kr)

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

ABSTRACT Recent advances in deep learning (DL) have led many contemporary automatic modulation classification (AMC) techniques to use deep networks in classifying the modulation type of incoming signals at the receiver. However, current DL-based methods face scalability challenges, particularly when encountering unseen modulations or input signals from environments not present during model training, making them less suitable for real-world applications like 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 the Meta-Transformer, a meta-learning framework based on few-shot learning (FSL) to acquire general knowledge and a learning method for AMC tasks. This approach empowers the model to identify new unseen modulations using only a very small number of samples, eliminating the need for complete model retraining. Furthermore, we enhance the scalability of the classifier by leveraging main-sub transformer-based encoders, enabling efficient processing of input signals with diverse setups. Extensive evaluations demonstrate that the proposed AMC method outperforms existing techniques across all signal-to-noise ratios (SNRs) on RadioML2018.01A. The source code and pre-trained models are released at https://github.com/cheeseBG/meta-transformer-amc.

INDEX TERMS Automatic modulation classification, few-shot learning, meta-learning, transformer, unseen dataset.

I. INTRODUCTION

Accurate classification of modulation types in incoming signals is a key element of the wireless communication system. Automatic modulation classification (AMC) and radio signal recognition methods play a crucial role in recognizing modulation types for various military and civilian services, such as dynamic spectrum access, jamming detection, surveillance, and spectrum coexistence. Typically, the preamble of a received signal carries details about its modulation scheme, enabling the receiver to determine the modulation type and pass it through to the appropriate demodulation process [1].

However, the design of a highly precise AMC scheme is challenging in the modern wireless communication environment since numerous heterogeneous communication systems coexist in a complex and non-cooperative manner. Performing the AMC task is particularly challenging in Cognitive Radio (CR) networks and Software Defined Radio (SDR) systems, as they provide the flexibility to employ various wireless communication services over a wide frequency range. In CR and SDR environments, dynamic spectrum sensing and access are performed over a wide frequency band in a non-cooperative manner. This often leads to inconsistent and partial signal reception. It is important to

The associate editor coordinating the review of this manuscript and approving it for publication was Angelo Trotta.

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 [2].

AMC methods can be categorized into two primary types: (i) *likelihood-based* (LB) and (ii) *feature-based* (FB) approaches [3]. LB approaches achieve high classification accuracy by harnessing prior knowledge about the target modulations [2]. Nonetheless, with an increase in the number of target modulations, LB approaches encounter difficulties, including elevated computational complexity and even mathematical intractability [4].

In recent years, FB approaches have extensively incorporated deep learning (DL) into AMC, attracting attention due to their outstanding classification performance, even when dealing with numerous target modulations [1], [2], [5], [6], [7], [8], [9], [10], [11]. DL-based AMC methods learn valid classification rules from a substantial amount of complex modulation data [12] and achieve high accuracy in modulation classification thanks to recent breakthroughs in deep learning techniques.

However, current DL-based AMC methods still face challenges in terms of real-world deployment, primarily due to their limited scalability, particularly when dealing with unseen modulations or input signals with different configurations not seen during the training process. In non-cooperative and complex real-world communication environments, received inputs frequently deviate from the features employed during the model training phase, resulting in substantial classification errors. Note that the performance of most DL-based AMC methods heavily depends on the availability of a substantial volume of training data. For instance, most DL-based AMC methods utilize fixed frame lengths as inputs to their models and do not account for scenarios with variable input sizes [10]. Accordingly, they may not work properly for short input frames.

Fig. 1 illustrates the classification accuracy of ResNet-based and CNN-based methods [10], [13] across various input frame lengths. These methods utilize an input frame length of 1024, which corresponds to the frame length of the training dataset. Testing for variable input lengths was performed by undersampling to lengths of {512, 256, 128, 64}, where shorter inputs were duplicated and concatenated in the preprocessing step to reach the default length of 1024. When the model was exclusively trained with a fixed-length frame of 1024 samples, we observed a deterioration in classification performance as the input frame length decreased. Unfortunately, it is nearly impossible to collect sufficient labeled training datasets in advance for numerous combinations of the target classes, such as varying frame lengths and signal-to-noise ratios (SNRs), to maintain classification accuracy. Furthermore, when introducing previously unseen modulations, existing solutions necessitate the collection of a substantial number of samples and a subsequent retraining of the model. In this paper, the term "unseen" refers to the data belonging to classes that the model did not encounter during the training phase. Therefore, it is essential to devise a more intelligent and scalable AMC technique capable of adapting to new unseen modulations and recognizing input signals with complex combinations of temporal and spatial features.

TABLE 1. Abbreviations and meanings.

Abbreviation	Meaning
DL	Deep Learning
AMC	Automatic Modulation Classification
FSL	Few-Shot Learning
SNR	Signal-to-Noise Ratio
CR	Cognitive Radio
SDR	Software Defined Radio
LB	Likelihood-Based
FB	Feature-Based
IQ	In-phase and Quadrature
OOK	On-Off Keying
ASK	Amplitude Shift Keying
PSK	Phase Shift Keying
APSK	Amplitude Phase Shift Keying
BPSK	Binary Phase Shift Keying
QPSK	Quadrature Phase Shift Keying
QAM	Quadrature Amplitude Modulation
AM	Amplitude Modulation
AM-SSB-WC	AM Single-Sideband With Carrier
AM-SSB-SC	AM Single-Sideband Suppressed Carrier
AM-DSB-WC	AM Double Side Band With Carrier
AM-DSB-SC	AM Double Side Band Suppressed Carrier
FM	Frequency Modulation
GMSK	Gaussian low-pass-filtered Minimum Shift Keying
OQPSK	Offset Quadrature Phase Shift Keying

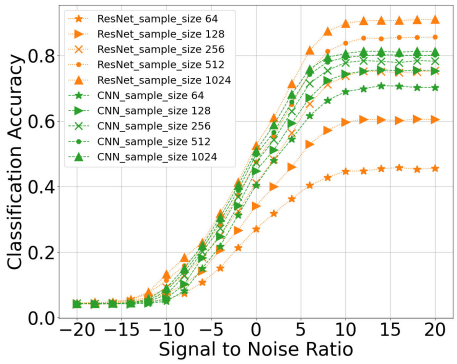


FIGURE 1. Impact of input frames' lengths on the classification accuracy of ResNet-based and CNN-based methods [10], [13] under different SNR values based on the RadioML2018.01A [5].

In this paper, we introduce Meta-Transformer, a scalable AMC scheme that provides flexibility for new unseen modulations and adaptability to input signals with diverse configurations. Our proposed framework comprises three key components: (i) a meta-learning framework that employs few-shot learning (FSL), (ii) a feature extractor built upon a Transformer architecture [14], and (iii) a main-sub model architecture to ensure scalability for input frame sizes. Within our proposed meta-learning framework, we initially train the main-sub model on a source dataset, targeting a specific set of modulations. Subsequently, we adapt the trained encoders to

new target modulations using only a small number of newly gathered samples.

This approach effectively mitigates the issues related to data collection and the overhead of retraining when addressing new unseen modulations. Moreover, to enhance the scalability and performance of the model, we leverage a Transformer-based encoder [14] in the design of the feature extractor for our proposed AMC method.

The noteworthy point is the operation of the Vision Transformer [14]. ViT divides an image into patches and tokenizes them for processing. The size of these patches plays a critical role in determining the receptive field. When configuring larger patch sizes, the model can capture a broader context initially, but it might miss finer details. Conversely, opting for smaller patch sizes allows the model to capture more intricate details but concurrently increases the risk of overfitting, as it might focus excessively on localized information. Hence, there exists a trade-off relationship to consider when determining the optimal patch size. Through extensive evaluations on the RadioML2018.01A dataset [5], we find the appropriate patch size for the AMC task and demonstrate that the proposed method consistently outperforms existing techniques across all signal-to-noise ratios (SNRs).

In real-world scenarios where AMC technology is applied, the input frame length is likely to vary, unlike the fixed length of 1024 frames used in the dataset employed in this study. This demands a solution for variable input signal lengths, and when using a single encoder, performance significantly degrades if the input length during testing is shorter than that used during training (Fig. 1). In fact, using multiple encoders ensures higher performance for diverse input lengths. However, there is a trade-off as the model becomes heavier with an increasing number of encoders, requiring consideration of computational constraints. Therefore, this paper proposes a method of employing two encoders.

The remainder of the paper is structured as follows: Section II summarizes related research work. Section III describes the overview of the proposed meta-learning based AMC scheme and its details. Section IV presents the evaluation results, and Section V concludes this paper.

II. RELATED WORK

Related work can be categorized into two main groups: (i) deep learning-based approaches focused on enhancing modulation recognition and classification performance, and (ii) studies utilizing few-shot learning techniques for AMC.

O'Shea et al. [5] utilized a 1D CNN based on ResNet [13] to extract features from the in-phase and quadrature-phase (IQ) components of the signal. They conducted experiments using the RadioML2018.01A dataset and achieved high accuracy across 24 modulations, demonstrating the effectiveness of CNN-based models for AMC. Subsequent studies using CNNs have made efforts to enhance performance in AMC using the RadioML2018.01A dataset. Kim et al. [10] proposed a CNN model that employed frame replication to expand it into a size of 4×1024 facilitating meaningful feature extraction, and utilized average pooling to reduce computational complexity. Huynh-The et al. [6], [11] proposed MCNet, which demonstrates efficient computational complexity based on 1D CNN, and RanNet, which shows high performance using the residual-attention structure. Additionally, various other CNN-based studies [7], [8], [9] have been conducted.

Recently, methodologies employing few-shot learning techniques have emerged to tackle the constraints of DL in AMC, as discussed in Section I. Zhou et al. [15] introduced AMCRN, an architecture based on CNN that assesses feature similarity between test data and annotated few-shot data. Zhang et al. [16] proposed the Attention Relation Network, which incorporates channel and spatial attention to enable modulation pattern recognition even with few-shot samples. Hao et al. [17] proposed M-MFOR, a meta-learning system leveraging few-shot learning. Their work demonstrates the ability to achieve high accuracy, even on modulation datasets with distribution bias, by effectively generalizing the meta-knowledge learned through meta-learning.

In contrast to prior research employing few-shot learning approaches with CNN-based models, to the best of our knowledge, we have pioneered the integration of a Transformer-based encoder into the meta-learning-based AMC. Our approach efficiently learns inter-sample relationships through the self-attention mechanism during the training phase. This enables rapid adaptation and achieves excellent performance even with limited data for unseen modulations during the testing phase.

III. PROPOSED METHOD

This section begins with an introduction to the Meta-Transformer, our proposed meta-learning framework designed for the AMC task. In this section, we first introduce the Meta-Transformer, which is the proposed meta-learning framework for the AMC task, and then explain its specifics, covering both the meta-training and meta-testing processes. The objective of our work is to overcome the aforementioned limitations of supervised learning-based DL approaches and ensure scalability for unseen modulations or input signals with varying configurations not seen during the training phase.

A. META-TRANSFORMER

Fig. 2 illustrates the architecture of Meta-Transformer. Our system consists of two main modules: (i) a meta-training module and (ii) a meta-testing module.

- **Meta-Training Module**: The module utilizes a source dataset for specific modulation classes, referred to as *seen* modulations. It trains the modulation classifier, namely main-sub Transformer-based encoders f_θ and $f_{\theta'}$, where θ and θ' represent the trainable parameters. Unlike traditional supervised learning methods,

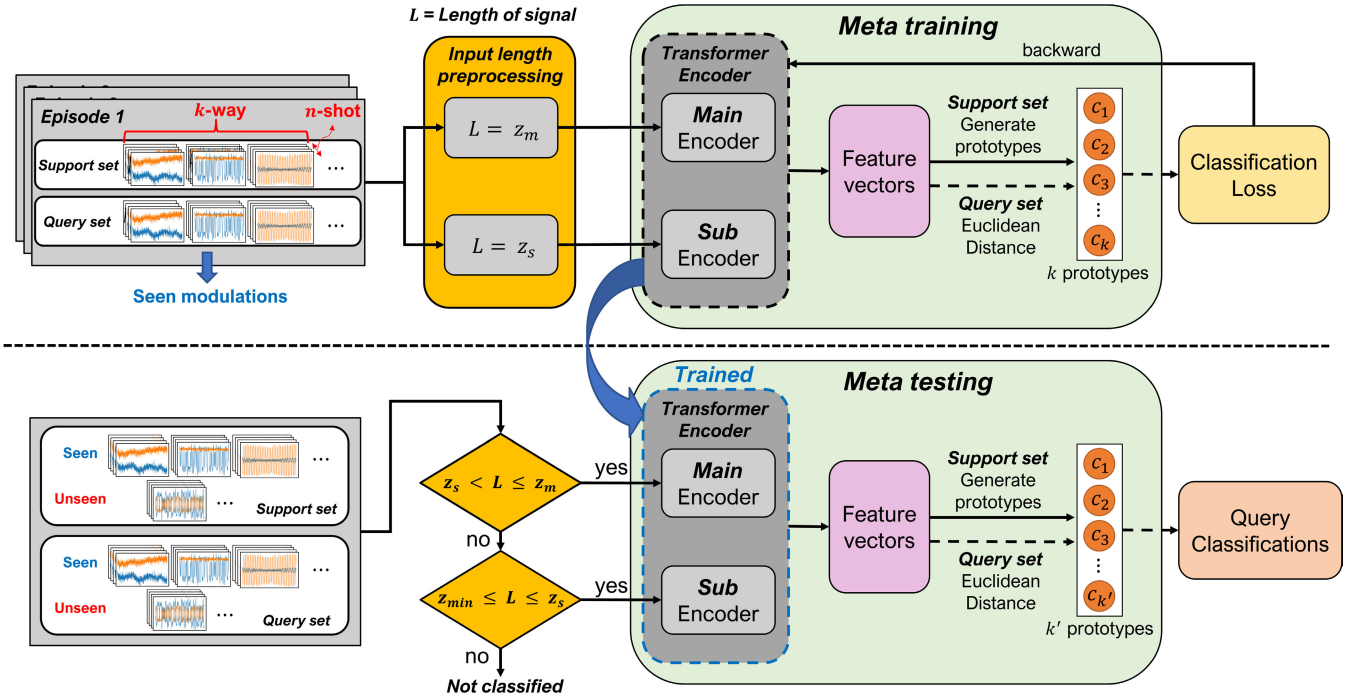


FIGURE 2. Overview of Meta-Transformer: meta-training with source datasets for given target modulations and meta-testing process with unseen modulations limited size datasets.

TABLE 2. Notations and meanings.

Notation	Meaning
f_θ	Main encoder
$f_{\theta'}$	Sub encoder
θ	Main encoder's parameters
θ'	Sub encoder's parameters
z_m	Main encoder input frame length
z_s	Sub encoder input frame length
z_{min}	Minimum input length
p	Number of patch
s	Patch length
ϵ	Episode (composed of support set and query set)
N_ϵ	Total number of episodes
p_{train}	Training data ratio
N_S	Number of support set
N_Q	Number of query set
N_{epoch}	training epoch
S	Annotated data
y_i	Class label set
c_l	Prototype for class l
d	Distance function
L	Loss function
α	Learning rate
γ	Learning scheduler parameter

our meta-learning approach acquires meta-knowledge, enabling quicker adaptation to new tasks even with limited samples (Section III-B). During the meta-training phase, the main encoder is trained with an input frame size $2 \times z_m$, while the sub-encoder is trained with $2 \times z_s$. The variables z_m and z_s denote the length of the frame, playing a crucial role in determining the performance of both the main and sub encoders. A detailed explanation of the variables is covered in Section III-C.

- **Meta-Testing Module**: Once trained, the meta-testing module uses the encoders $f_\theta, f_{\theta'}$ for new unseen modulations with fewer collected samples (Section III-C).

The main encoder f_θ and the sub encoder $f_{\theta'}$ learn general meta-knowledge to extract appropriate feature vectors for AMC tasks, where meta-knowledge represents the underlying essence or commonality among multiple tasks [17]. To achieve this, we utilize the methodology of learning the metric space by using prototypes of each class, as introduced in ProtoNet [18].

Fig. 3 depicts the architecture of the encoder and its operational sequence. To handle various input signal configurations, we employ a feature extractor based on the Transformer architecture proposed by Dosovitskiy et al. [14]. The Transformer-based encoder has a modular architecture, allowing each layer to work independently. Communication between layers is facilitated through attention mechanisms [14], providing flexibility in adjusting the model's size and complexity. In this setup, each module has an input layer of $2 \times N$, which takes IQ components of signal data as input. These components are split into p patches of size $2 \times s$, with each patch undergoing linear embedding after the addition of position information embeddings.

Table 3 summarizes the hyperparameters used for the main and sub encoders within Meta-Transformer. Since the sub encoder is trained with a smaller input frame size compared to

TABLE 3. Details of proposed model hyperparameters.

Model	Layers	Hidden D	MLP	Heads	Input	Patch
main	8	36	32	9	$2 \times z_m$	$2 \times s$
sub	8	108	32	9	$2 \times z_s$	$2 \times s$

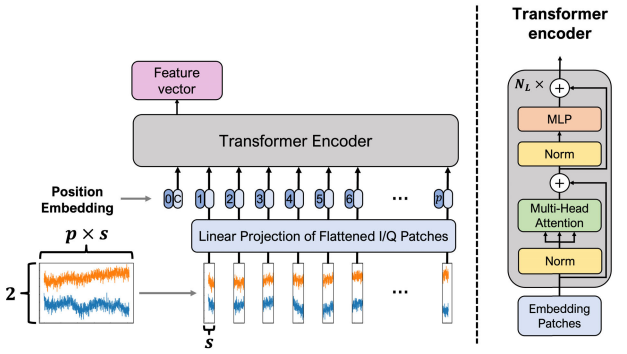


FIGURE 3. Transformer-based encoder f_θ to extract feature vectors of I/Q signals. We employed ViT [14]'s encoder structure.

the main encoder, we adjusted the hidden size dimension D to a larger value to optimize performance. The input length z and patch length s are pivotal hyperparameters affecting signal data manipulation and overall model performance. We conducted experiments focused on determining the most suitable values for these parameters, elaborated upon in Section IV. Additionally, we determined the remaining hyperparameters through empirical experiments to achieve optimal model performance.

B. META-TRAINING

The meta-training module trains two encoders, denoted as f_θ and $f_{\theta'}$, incorporating meta-knowledge for the ACM task. Both encoders follow an identical training approach, which will be explained hereafter explained with respect to f_θ. Training occurs episodically in this phase. Each episode, labeled as ϵ, comprises two parts: (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 first randomly choose k categories from the source dataset. Within each selected category, we then randomly pick n instances. Here, k represents the total number of classes within the support set, often referred to as k-way, and n represents the number of data samples for each class (way), known as n-shot. The total number of episodes, denoted as N_ϵ, can be determined using the following equation:

$$N_\epsilon = \frac{p_{train} * N}{N_S + N_Q} * N_{epoch}, \tag{1}$$

where N represents the total number of data, p_{train} is the ratio of the training dataset, N_S is the number of support sets, N_Q is the number of query sets, and N_{epoch} is the number of training epochs. Here, the modulation classes used in training are regarded 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 $1 \times 2 \times z$ ($C \times H \times W$), which is provided in the RadioML2018.01A [5] dataset. The corresponding class labels are represented as $y_i = \{1, \ldots, K\}$.

In each episode, the signal undergoes preprocessing based on experimentally derived values for z_m and z_s. Following this, the support set and query set data are segmented into patches as previously described and then fed into the encoder. For the support set, the prototype c_l is created by averaging the extracted feature vectors (referred to as embedded support points) from the annotated dataset S_l belonging to class l.

$$c_l = \frac{1}{|S_l|} \sum_{(x_i, y_i) \in S_l} f_\theta(x_i) \tag{2}$$

The feature vectors extracted from the query set are classified using the generated prototypes, based on a distance function d, which could be methods such as Euclidean distance or cosine similarity. In ProtoNet [18], the distance between the query embedding and the prototype is measured by *Euclidean distance*, which shows excellent performance. Consequently, we also employed Euclidean distance as our 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:

$$p_\theta(y = l|x) = \frac{exp(-d(f_\theta(x), c_l))}{\sum_{l'} exp(-d(f_\theta(x), c_{l'}))} \tag{3}$$

Algorithm 1 Process of Meta Training. $k \leq K$ Is the Number of Classes per Episode, E Is the Selected k Classes for Episode, N_S Is the Number of Support samples per Class, N_Q Is the Number of Query samples 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 uniformS, N Denotes Uniform and Random Selection of N Values From the S Set. Signal LengthS, z Denotes the Adjustment of All x Lengths in Set S to z.

Input: Training set $S_{train} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$
Output: Trained base encoder f_θ
For l in {1,…,k} **do**
  $S_{support} \leftarrow$ Random uniform(S_{E_l}, N_S)
  $S_{query} \leftarrow$ Random uniform$(S_{E_l} \setminus S_{support}, N_Q)$
  $S_{support} \leftarrow$ Signal Length$(S_{support}, z_m)$
  $S_{query} \leftarrow$ Signal Length(S_{query}, z_m)
  $c_l \leftarrow \frac{1}{N_S} \sum_{(x_i, y_i) \in S_l} f_\theta(x_i)$
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

As each episode progresses, the parameters θ of f_θ are iteratively updated using the Adam optimizer [19] to minimize the negative log probability of the actual class k, as described in Equation 4.

$$L(\theta) = -log p_\theta(y = k|x) \tag{4}$$

The algorithm 1 illustrates the main encoder meta-training process for an episode. The sub encoder is also trained using the same approach.

C. META-TESTING

Meta-Testing module utilizes the encoders f_θ and $f_{\theta'}$ trained through the meta-training phase. The parameters θ, θ' remain fixed and are not updated during the meta-testing process. In the meta-testing phase, both the support set and the query set consist of unseen modulations, enabling us to evaluate the model's adaptation to a new domain and assess its generalization capability. The signal length L for both the support set and the query set follows these conditions: $z_s < L \leq z_m$ for input to the main encoder, and $z_{min} \leq L \leq z_s$ for input to the sub-encoder. Here, z_m is set to 1024, the signal length of RadioML2018.01A, and z_s is experimentally chosen as 128 to enhance the model's scalability for shorter signal lengths. Additionally, the minimum length z_{min} is set to 64, achieving over 50% performance, and anything below cannot be used for classification. As we will discuss in Section IV, we consider the application of our method to SDR platform scenarios. For example, in the case of operational SDR equipment, upgrades are necessary to enable recognition of new modulations not encompassed in the current model's training. For this purpose, the meta-testing module can include datasets for both the seen modulations used in the training phase and new unseen modulations. The results of these tests are presented in Section IV-D. A commonly adopted configuration for support sets in most FSL-based approaches is the 5-shot setting, where the support set comprises five data samples. Similar to the meta-training phase, the meta-testing module generates k' prototypes using the trained f_θ, $f_{\theta'}$, where k' denotes the number of target classes for meta-testing. The query set used for inference is classified based on the Euclidean 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 IV-D.

IV. PERFORMANCE EVALUATION

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

The training dataset ratio p_{train} is set to 0.8, and N_{epoch} is set to 50 for the main encoder and 100 for the sub encoder. The optimizer used is Adam [19], 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 widely utilized RadioML2018.01A dataset [5] 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. This diverse set of modulations includes high-order schemes like QAM256 and APSK256. Each frame consists of 1024 samples for the IQ components. The dataset consists 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. PATCH SIZE

The proposed Meta-Transformer utilizes an encoder based on ViT [14], and the input signal is divided and tokenized at the patch level for processing. In this case, the patch size s used has an impact on the receptive field, ultimately affecting the classification performance. In the original ViT, images are divided into patches of size $s \times s$, assuming a square-shaped image. However, for the AMC task, signals are provided in the form of IQ components, resulting in a 2D shape of $2 \times L$. When setting s to 2 and dividing the signal into 2×2 patch size, the amount of information becomes extremely low, resulting in significantly reduced classification performance. Therefore, we conducted experiments to find an appropriate value for s that is suitable for the AMC task. The proposed model comprises both the main and sub encoders, each trained for signal lengths of 1024 and 128. Hence, we conducted separate experiments to determine the appropriate s value for both encoders with these two input sizes.

Fig. 4 and Fig. 5 depict the experimental results for $s = \{8, 16, 32, 64\}$. In both figures, the best performance is observed when $s = 16$ in good SNR environments (0 dB or higher). It can be noted that performance decreases as the patch size increases or decreases from this optimal value. On the other hand, in poor SNR environments, there is a slight difference, but better performance is observed as s increases. This suggests that, in the presence of high noise, expanding the receptive field is necessary to capture a broader range of information. We selected $s = 16$ as the default patch size, ensuring robust performance in good SNR environments, and proceeded with the remaining experiments.

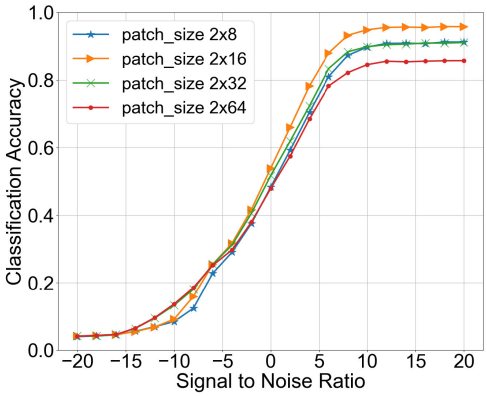


FIGURE 4. Performance test based on the patch size of the main encoder.

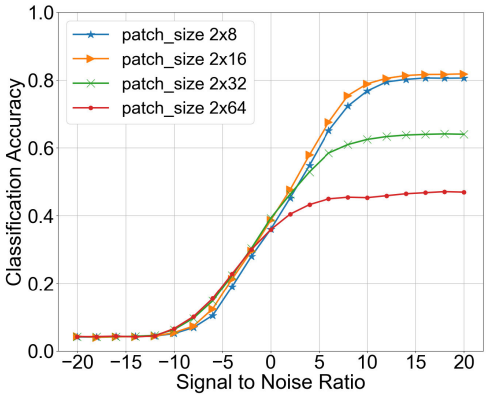


FIGURE 5. Performance test based on the patch size of the sub encoder.

C. COMPARING META-LEARNING AND SUPERVISED LEARNING

We first conducted an experiment to compare how well meta-learning models, including our proposed model, perform in accurately classifying the 24 different modulations, compared to supervised learning models and other transformer-based models. The supervised learning models used in the experiments include ResNet [5] based and CNN [10] based models, where we will denote them as ResNet and CNN, respectively. For the meta-learning models, we employed ProtoNet [18] and DAELSTM [20] along with our proposed model. Note that the original DAELSTM [20] is a supervised learning-based model, but we modified it into a meta-learning structure by leveraging our encoder-based framework. We then included both the modified DAELSTM and the original one in our comparison experiment. These five models were trained using a SNR range of [−10, 20] dB, demonstrating their optimal performance within this SNR range. Their performance was evaluated in terms of accuracy, with a step size of 2 dB, across the entire range of [-20 to 20] dB contained in the RadioML2018.01A dataset. This experiment was designed to compare the performance of the proposed meta-learning approach with models proposed using traditional supervised learning methods. Additionally,

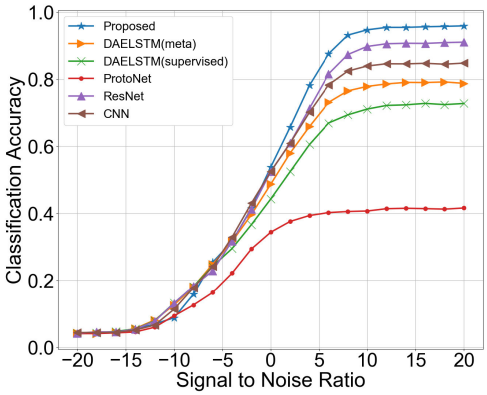


FIGURE 6. Performance comparison between meta-learning (our proposed method, DAELSTM [20] and ProtoNet [18]) and supervised learning (ResNet [5] and CNN [10]) models for all 24 modulations.

TABLE 4. Complexity comparison of different models.

Model	FLOPs	Memory	Speed	Params
ResNet [5]	0.026G	5.32MB	0.004s	0.17M
CNN [10]	0.038G	16.69MB	0.005s	0.04M
ProtoNet [18]	**0.014G**	7.06MB	**0.002s**	.02M
DAELSTM [20]	0.028G	**0.06MB**	0.012s	**0.01M**
Proposed	0.046G	15.98MB	0.005s	0.72M

it aims to verify the effectiveness of the transformer architecture for the AMC task by comparing it with commonly used CNN and LSTM-based models.

Fig. 6 shows the results of the evaluation. We observed that our proposed Meta-Transformer achieved the highest performance at 95.76% accuracy in the good SNR range, particularly at SNR 20 dB (Fig. 7). This demonstrates the effectiveness of our transformer architecture and meta-learning approach for the AMC task, outperforming CNN and ResNet models based on conventional supervised learning methods. Furthermore, we demonstrated that even existing models like DAELSTM, originally proposed using supervised learning, can be adapted to our proposed framework by utilizing the framework's encoder. ProtoNet, despite being a meta-learning approach, exhibited lower performance. This suggests that the model architecture was not specifically designed to address the AMC task, highlighting the importance of tailoring the model structure to suit the requirements of the task.

We also compared the complexity of the models in Table 4. Despite having a higher computational complexity compared to the other four models, the proposed model demonstrated the best performance across all 24 modulations.

D. UNSEEN MODULATION

Next, we evaluate the adaptation performance of our proposed method to new modulation types. As mentioned previously, one of the advantages of meta-learning is its ability to quickly adapt the model to new unseen classes. For instance, consider a scenario where an operational SDR

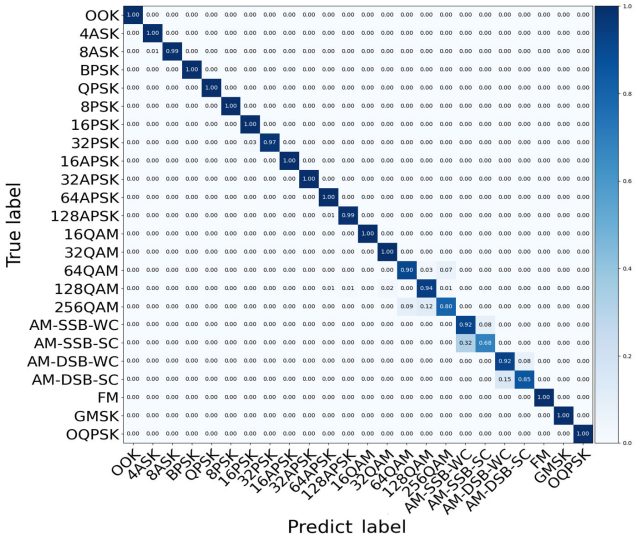


FIGURE 7. Confusion matrix for the proposed model inference results at SNR 20dB.

TABLE 5. 12 modulations used for training in three repeated test cases.

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'

equipment requires an upgrade to recognize new modulation types. We conducted experiments where the proposed model was trained on 12 randomly selected 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 5. For each test case, we carried out 100 test iterations, with each iteration involving the random selection of five *Unseen* modulations. We then calculated the average accuracy. The default value for "shot" was set to 5-shots. The reason is that many few-shot learning studies use 1-shot and 5-shot evaluations as benchmarks. The number of sample frames used for training was approximately 1.3 million, while for testing, around 0.5 million frames were used for 5 randomly selected modulations.

Fig. 8 depicts the accuracy results for the three test cases, illustrating an average accuracy of around 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 used during the training phase. More complex modulations tend to demonstrate better performance during the inference phase.

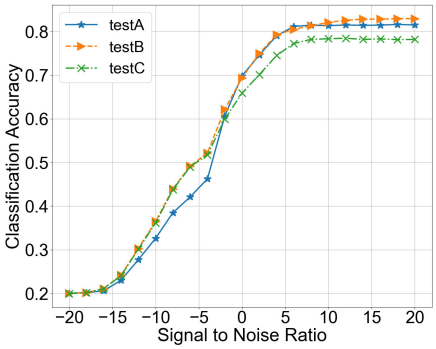


FIGURE 8. Performance comparisons for three test cases in Table 5.

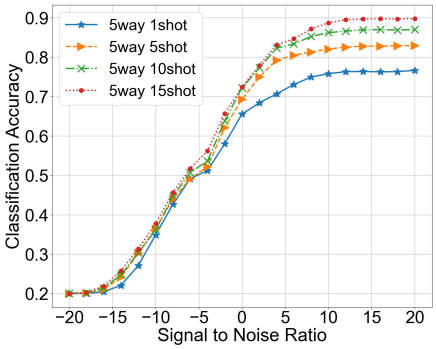


FIGURE 9. Impact of number of shots on classification performance for 5-way (five *Unseen* modulations/classes) with 1, 5, 10, and 15 different shots.

For the subsequent experiments, we used the Test B category in Table 5.

Fig. 9 presents the results of an experiment that investigated the influence 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 demonstrate that accuracy increases with a higher number of shots. 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 datasets.

Fig. 10 displays the classification performance results for three different numbers of *Unseen* modulations while keeping the shots fixed at 5. We conducted tests with *Unseen* modulations consisting of 3, 5, and 7 classes. The results indicate that as the number of *Unseen* modulations decreases, there is an improvement in differentiating prototypes within the embedding space, leading to higher performance. Notably, a substantial increase in classification accuracy is observed for lower SNRs as the number of *Unseen* modulations decreases.

Fig. 11 represents an experiment tailored for the SDR platform scenarios. The evaluation encompasses both the *Seen* modulations used in the training phase and the *Unseen* modulations. Despite the inherent challenges posed by the 12-way and 13-way configurations in the meta-learning

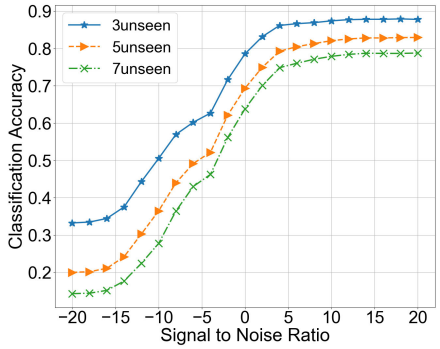


FIGURE 10. Performance evaluation for different numbers of ways, i.e., 3, 5, and 7 *Unseen* modulations, with a fixed 5-shot learning.

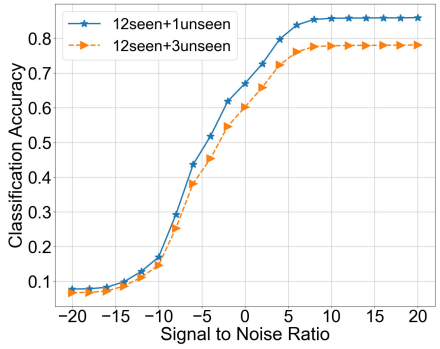


FIGURE 11. Performance evaluation using both the 12 *Seen* modulations used during training and additional *Unseen* modulations in the test phase.

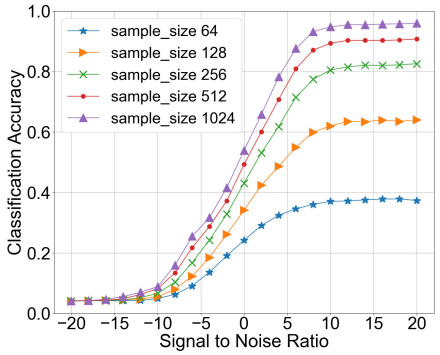


FIGURE 12. The impact of varying input frame lengths on the classification accuracy of the proposed model (only main encoder) in the RadioML2018.01A dataset [5].

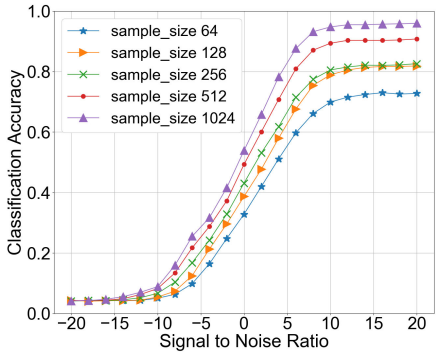


FIGURE 13. The impact of varying input frame lengths on the classification accuracy of the proposed model in the RadioML2018.01A dataset [5].

context, our method achieved a high performance level, surpassing 80% accuracy with 5-shot learning. We expect that performance can be further improved through an investigation of hyperparameters and by leveraging more powerful computing environments. We plan to explore these possibilities in greater detail in our future work.

E. 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 overlooked in both model design and evaluation. Fig. 12 clearly indicates that using the main model trained with a 2×1024 frame size results in performance degradation for smaller input frame sizes when actually utilized. Therefore, we conducted additional experiments to evaluate the scalability of our Meta-Transformer using two encoders for frame sizes smaller than the given 2×1024 frames. Specifically, we experimented with frame sizes of 2×64, 128, 256, 512 to evaluate the model's performance and generalizability.

Fig. 13 presents the results of evaluating the proposed model using smaller input frames while it was trained with 2×1024 frames. Thanks to the main-sub encoder structure, even for samples with smaller input frame sizes, each encoder effectively captures the interactions between sample patches within the same frame. Consequently, using smaller input frame sizes results in a relatively minor performance degradation compared to using the main encoder architecture alone, specifically for sizes below 2×128.

V. CONCLUSION

In this work, we introduced a Meta-Transformer, a scalable AMC scheme that provides flexibility for handling new modulations and adaptability to diverse input signal configurations. By utilizing a meta-learning framework based on FSL, we empowered the model to acquire general knowledge and effectively 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 employing two Transformer-based encoders, 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.

Despite achieving rapid adaptation to a new set of modulations and providing a high classification performance of over 90%, incorporating the modulations from the initial

training stage poses a challenging task. This challenge can be particularly critical in SDR platform scenarios demanding precise classification across a diverse range of modulation types. In our future endeavors, we plan to address these challenges.

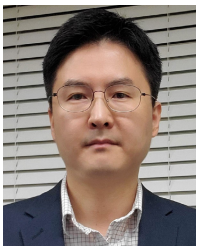
REFERENCES

[1] S. Peng, S. Sun, and Y.-D. Yao, "A survey of modulation classification using deep learning: Signal representation and data preprocessing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7020–7038, Dec. 2022.
[2] S. Zheng, P. Qi, S. Chen, and X. Yang, "Fusion methods for CNN-based automatic modulation classification," *IEEE Access*, vol. 7, pp. 66496–66504, 2019.
[3] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: Classical approaches and new trends," *IET Commun.*, vol. 1, no. 2, pp. 137–156, 2007.
[4] F. Hameed, O. A. Dobre, and D. C. Popescu, "On the likelihood-based approach to modulation classification," *IEEE Trans. Wireless Commun.*, vol. 8, no. 12, pp. 5884–5892, Dec. 2009.
[5] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.
[6] T. Huynh-The, C.-H. Hua, Q.-V. Pham, and D.-S. Kim, "MCNet: An efficient CNN architecture for robust automatic modulation classification," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 811–815, Apr. 2020.
[7] G. B. Tunze, T. Huynh-The, J.-M. Lee, and D.-S. Kim, "Sparsely connected CNN for efficient automatic modulation recognition," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15557–15568, Dec. 2020.
[8] P. Qi, X. Zhou, S. Zheng, and Z. Li, "Automatic modulation classification based on deep residual networks with multimodal information," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 1, pp. 21–33, Mar. 2021.
[9] R. Zhang, Z. Yin, Z. Wu, and S. Zhou, "A novel automatic modulation classification method using attention mechanism and hybrid parallel neural network," *Appl. Sci.*, vol. 11, no. 3, p. 1327, Feb. 2021.
[10] S.-H. Kim, J.-W. Kim, W.-P. Nwadiugwu, and D.-S. Kim, "Deep learning-based robust automatic modulation classification for cognitive radio networks," *IEEE Access*, vol. 9, pp. 92386–92393, 2021.
[11] T. Huynh-The, Q.-V. Pham, T.-V. Nguyen, T. T. Nguyen, D. B. D. Costa, and D.-S. Kim, "RanNet: Learning residual-attention structure in CNNs for automatic modulation classification," *IEEE Wireless Commun. Lett.*, vol. 11, no. 6, pp. 1243–1247, Jun. 2022.
[12] B. Jdid, K. Hassan, I. Dayoub, W. H. Lim, and M. Mokayef, "Machine learning based automatic modulation recognition for wireless communications: A comprehensive survey," *IEEE Access*, vol. 9, pp. 57851–57873, 2021.
[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[15] Q. Zhou, R. Zhang, J. Mu, H. Zhang, F. Zhang, and X. Jing, "AMCRN: Few-shot learning for automatic modulation classification," *IEEE Commun. Lett.*, vol. 26, no. 3, pp. 542–546, Mar. 2022.
[16] Z. Zhang, Y. Li, and M. Gao, "Few-shot learning of signal modulation recognition based on attention relation network," in *Proc. 28th Eur. Signal Process. Conf. (EUSIPCO)*, Jan. 2021, pp. 1372–1376.
[17] X. Hao, Z. Feng, S. Yang, M. Wang, and L. Jiao, "Automatic modulation classification via meta-learning," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12276–12292, Jul. 2023.
[18] J. Wang and Y. Zhai, "Prototypical Siamese networks for few-shot learning," in *Proc. IEEE 10th Int. Conf. Electron. Inf. Emergency Commun. (ICEIEC)*, Jul. 2020, pp. 178–181.
[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[20] Z. Ke and H. Vikalo, "Real-time radio modulation classification with an LSTM auto-encoder," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 4935–4939.

JUNGIK JANG (Student Member, IEEE) received the M.S. degree from the Department of Software, Gachon University, South Korea, in 2023. Since 2021, he has been a Researcher with the Intelligent Networking and Computing Laboratory (INC Lab.), Gachon University. His current research interests include deep learning-based wireless sensing, domain adaptation, and generative AI.

JISUNG PYO is currently pursuing the B.S. degree with the School of Computing, Gachon University, South Korea. Since 2022, he has been a Researcher with the Intelligent Networking and Computing Laboratory (INC Lab), Gachon University. His research interests include Wi-Fi sensing and meta-learning.

YOUNG-IL YOON received the M.S. degree from Chungnam National University, Daejeon, South Korea, in 2013. From 2012 to 2013, he was a Software Engineer with the Cloud Team, Naver Cloud, contributing to the virtual machine. Since 2013, he has been the Project Manager with the C4I Research Center, LIG Nex1, contributing to the tactical radio systems. His current research interests include software frameworks and embedded Linux kernels.



JAEHYUK CHOI (Member, IEEE) received the Ph.D. degree in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2008. From 2008 to 2011, he was a Postdoctoral Researcher with the Real-Time Computing Laboratory, University of Michigan, Ann Arbor, MI, USA. He is currently a Professor with the Department of Software, Gachon University, Seongnam, South Korea. His current research interests include wireless/mobile systems, the Internet of Things connectivity, and intelligent sensing systems.